Trần Thạch Tùng

Hiệu đính : **Trần Tiến Dũng**
Cố vấn khoa học : **Đoàn Thiện Ngân**



Bảo mật và Tối ưu trong Red Hat Linux

- Hướng dẫn chi tiết cài đặt Hệ điều hành và các ứng dụng chính của một máy Linux (6.x, 7.x).
- Đĩa mềm chứa đầy đủ các tập tin cấu hình chuẩn cho các ứng dụng chính trên Linux.

NHÀ XUẤT BẢN LAO ĐỘNG - XÃ HỘI

Trần Thạch Tùng
Hoàng Đức Hải
Hiệu đính : Trần Tiến Dũng
Cố vấn khoa học : Đoàn Thiện Ngân

Bảo mật và Tối ưu trong Red Hat Linux

NHÀ XUẤT BẢN LAO ĐỘNG - XÃ HỘI
NĂM 2002

GIỚI THIỆU

Giới thiệu về cuốn sách

Khi bắt đầu viết cuốn sách này, câu hỏi đầu tiên tôi đã tự hỏi mình là làm thế nào cài đặt Linux trên một server và đảm bảo rằng không một ai từ bên ngoài hoặc bên trong Linux có thể truy cập Linux mà không có sự cho phép. Kế đến tôi tự hỏi có bất kỳ phương pháp nào giống như trên windows dùng để cải thiện hiệu năng của máy tính. Sau đó tôi đã bắt đầu tìm kiếm trên Internet, đọc sách, thu thập thông tin về bảo mật, về cải thiện hiệu năng cho hệ thống của tôi. Sau nhiều năm nghiên cứu và học hỏi, cuối cùng tôi đã tìm được câu trả lời cho câu hỏi của tôi. Những câu trả lời đã được tìm thông qua tài liệu, bài báo, sách và Internet. Tôi đã viết tài liệu dựa trên những nghiên cứu của tôi và có thể giúp tôi trong công việc hằng ngày. Qua nhiều năm, tài liệu của tôi đã lớn dần và giống như một cuốn sách với nhiều ví dụ và những chú ý được phân bổ theo từng chương. Tôi đã quyết định đưa cuốn sách này đến tay bạn đọc với hy vọng nó sẽ ít nhiều giúp ích cho các bạn đang làm việc với Linux.

Với việc chia sẻ thông tin này, tôi cảm thấy rằng tôi đã làm một việc có ích cho mọi người là trả lời nhiều câu hỏi về tính hấp dẫn, đáng tin cậy, ổn định, mạnh mẽ, nhanh chóng và miễn phí của hệ điều hành Linux. Tôi rất mong nhận được sự phản hồi và những góp ý chân thành về quyển sách này, nó sẽ giúp ích cho việc hoàn thiện trong thời gian tới, tôi cũng mong được mọi người đón nhận nó và sẽ tìm nhiều tiện ích của Hệ điều hành Linux.

Với nhiều thời gian cùng nhiều nỗ lực bản thân khi viết sách này chắc chắn rằng kết quả chính xác và khả thi. Tuy nhiên, sẽ không tránh khỏi những sai sót, nếu bạn tìm được những lỗi cũng như những thiếu sót ở bất kỳ chương nào trong cuốn sách này, làm ơn cho phép tôi biết để sửa đổi cho lần tái bản tiếp theo.

Độc giả

Cuốn sách này dành cho bất cứ những ai đã, đang và sẽ sử dụng Linux, đặc biệt dành cho các kỹ thuật viên, những người quản trị hệ thống Linux và cũng là tài liệu tham khảo rất tốt cho các sinh viên đang học Linux. Cuốn sách này sẽ trình bày cho các bạn làm thế nào để cài đặt và thiết lập một Red Hat Linux server, cùng những vấn đề bảo mật cần thiết và sự tối ưu hoá một Linux server ở mức cao. Vì chúng ta nói đến cấu hình của tối ưu hoá và bảo mật, chúng ta sẽ sử dụng các chương trình nguồn thường được phân phối dưới dạng .tar.gz chẳng hạn, như những gói chương trình nguồn của Apache, BIND/DNS, Samba, Squid, OpenSSL....Với những gói tài nguyên này, chúng ta có thể nhanh chóng tiến hành việc nâng cấp, cập nhật bảo mật khi cần thiết, ngoài ra với những gói tài nguyên trên, chúng ta sẽ có một sự biên dịch theo ý muốn và tối ưu hoá tốt hơn, so với bình thường chúng ta không thể sử dụng với những gói dữ liệu dạng RPM.

Những yêu cầu chung

Chúng ta cần một ổ CD-ROM trên máy tính và một bộ CD chứa chương trình cài đặt Linux mà ta hay gọi là chương trình nguồn, tốt nhất là nên dùng bộ "Official Red Hat Linux" (bộ "chính thống"-?-) thay vì dùng bộ tải xuống miễn phí (free download) vì trong bộ "chính thống", các trình cài đặt đều đã được kiểm tra. Các hướng dẫn trong sách này đều đã được kiểm chứng với bộ "chính thống" phiên bản 6.1, 6.2, trong số này có những chương trình được kiểm tra trên Red Hat Linux 7.2.

Trước khi cài đặt Linux, bạn nên hiểu về các phần cứng có trong máy tính của bạn hoặc các thiết bị bạn sẽ cài thêm với Linux. Sau khi xem xét phần cứng máy tính, phần còn lại của tiến trình cài đặt sẽ được hướng dẫn trong cuốn sách này.

Những sản phẩm (chương trình) được đề cập trong cuốn sách

Có nhiều sản phẩm sẽ được đề cập trong cuốn sách này, chúng là những chương trình mang tính thương mại, nhưng cũng có nhiều sản phẩm không thương mại, tức là bạn được miễn phí khi sử dụng hoặc sửa đổi và phân phát cho người khác, do đó nhiều khi chúng ta cũng sẽ không xác định một cách chính xác nguồn gốc của những chương trình này. Tuy nhiên, chúng được sử dụng hằng ngày trong một số công ty kể cả những công ty lớn.

Tập tin cấu hình

Tất cả các phần mềm mà tôi mô tả trong cuốn sách này có một thư mục hoặc một thư mục con cụ thể trong tập tin được nén ở dạng lưu trữ tar có tên là "floppy.tgz" trong đĩa mềm kèm theo sách. Các thư mục đó chứa file cấu hình cho một phần mềm cụ thể nào đó. Ngoài việc bạn có thể có được các file cấu hình từ đĩa mềm, nếu bạn cần lấy một tập tin cấu hình nào đó, bạn có thể thực hiện việc cắt và dán chúng từ các phần cấu hình trong chương tương ứng trong sách. Dù bạn thực hiện bằng cách nào đi nữa, trách nhiệm của bạn là phải hiệu chỉnh chúng cho phù hợp với nhu cầu của bạn và đặt các tập tin cấu hình liên quan với các phần mềm tương ứng tới các nơi thích hợp trên máy server của bạn.

Lời cảm ơn:

Xin cảm ơn bố mẹ, người đã sinh ra và nuôi dưỡng con nên người, xin cảm ơn các bạn đồng nghiệp đặc biệt là anh Ngân và anh Dũng đã góp ý, bổ sung cho cuốn sách. Tôi cũng xin cảm ơn công ty Global Cybersoft Việt Nam và các đồng nghiệp trong phòng IT của công ty đã tạo điều kiện để tôi hoàn thành cuốn sách này.

Lời ngỏ

Kính thưa quý Bạn đọc gần xa, Ban xuất bản MK.PUB trước hết xin bày tỏ lòng biết ơn và niềm vinh hạnh trước nhiệt tình của đông đảo Bạn đọc đối với tủ sách MK.PUB trong thời gian qua.

Khẩu hiệu của chúng tôi là:

* Lao động khoa học nghiêm túc.

* Chất lượng và ngày càng chất lượng hơn.

* Tất cả vì Bạn đọc.

Rất nhiều Bạn đọc đã gửi mail cho chúng tôi đóng góp nhiều ý kiến quý báu cho tủ sách.

Ban xuất bản MK.PUB xin được kính mời quý Bạn đọc tham gia cùng nâng cao chất lượng tủ sách của chúng ta:

Trong quá trình đọc, xin các Bạn ghi chú lại các sai sót (dù nhỏ, lớn) của cuốn sách hoặc các nhận xét của riêng Bạn. Sau đó xin gửi về địa chỉ:

E-mail: mk.book@cinet.vnnews.com; mk.pub@cinet.vnnews.com

Hoặc gửi về: Nhà sách Minh Khai

249 Nguyễn Thị Minh Khai, Q.I, Tp. Hồ Chí Minh

Nếu Bạn ghi chú trực tiếp lên cuốn sách, rồi gửi cuốn sách đó cho chúng tôi thì chúng tôi sẽ xin hoàn lại cước phí bưu điện và gửi lại cho Bạn cuốn sách khác.

Chúng tôi xin gửi tặng một cuốn sách của tủ sách MK.PUB tùy chọn lựa của Bạn theo một danh mục thích hợp sẽ được gửi tới Bạn.

Với mục đích ngày càng nâng cao chất lượng của tủ sách MK.PUB, chúng tôi rất mong nhận được sự hợp tác của quý Bạn đọc gần xa.

" MK.PUB và Bạn đọc cùng làm !"

MK.PUB

Tổng quan

Giới thiệu

Phần 1 - Cài đặt Linux

Chương 1 - Giới thiệu về Linux

Chương 2 - Cài đặt server Linux của bạn

Phần 2 - Các vấn đề liên quan về bảo mật và tối ưu

Chương 3 - Tổng quan về bảo mật hệ thống

Chương 4 - Tổng quan về tối ưu hệ thống

Chương 5 - Cấu hình và xây dựng một kernel bảo mật và được tối ưu

Phần 3 - Liên kết mạng

Chương 6 - Quản lý hệ thống mạng Linux TCP/IP

Chương 7 - Liên kết Firewall

Chương 8 - Liên kết Firewall với hỗ trợ giả lập IP và chuyển tiếp

Phần 4 - Sự tham khảo các phần mềm liên quan

Chương 9 - Chức năng biên dịch

Chương 10 - Phần mềm bảo mật (Công cụ giám sát)

Chương 11 - Phần mềm bảo mật (Các dịch vụ mạng)

Chương 12 - Phần mềm bảo mật (Tính toàn vẹn hệ thống)

Chương 13 - Phần mềm bảo mật (Sự quản lý và sự giới hạn)

Chương 14 - Phần mềm server (Dịch vụ mạng BIND/DNS)

Chương 15 - Phần mềm server (Dịch vụ Mail)

Chương 16 - Phần mềm server (Dịch vụ mã hóa)

Chương 17 - Phần mềm server (Dịch vụ cơ sở dữ liệu)

Chương 18 - Phần mềm server (Dịch vụ Proxy)

Chương 19 - Phần mềm server (Dịch vụ về Web)

Chương 20 - Cài đặt Apache với các thành phần tùy chọn

Chương 21 - Phần mềm server (Dịch vụ chia sẻ tập tin)

Phần 5 - Các vấn đề liên quan về sao lưu dữ liệu

Chương 22 - Các thủ tục về sao lưu và khôi phục dữ liệu

Phần 6 - Cài đặt và cấu hình một số phần mềm trên Red Hat Linux 7.2

Chương 23 - Cài đặt WEB SERVER APACHE 1.3.20

Chương 24 - Cài đặt LINUX FREES/WAN

Chương 25 - Cài đặt Squid Proxy Server

Chương 26 - Cài đặt OpenSSH Client/Server

Chương 27 - Cài đặt và sử dụng Samba

Phần 7 - Các phụ lục

Phụ lục A - Những lắt léo, mẹo và công việc của người quản trị

Phụ lục B - Các RFC đang tồn tại

MỤC LỤC

GIỚI THIỆU **3**
Giới thiệu về cuốn sách 3
Độc giả 3
Những yêu cầu chung 4
Tập tin cấu hình 4
Lời ngỏ **5**
Tổng quan **7**

PHẦN 1 : HƯỚNG DẪN CÀI ĐẶT LINUX **17**
Chương 1: GIỚI THIỆU VỀ LINUX **19**
Linux là gì? 20
Một vài nguyên nhân dẫn đến việc sử dụng Linux 20
Xua tan đi sự lo ngại, sự không chắc chắn và sự nghi ngờ về Linux 21
Chương 2: CÀI ĐẶT SERVER LINUX **23**
Hướng dẫn cài đặt Linux 24
Sự hiểu biết về phần cứng trong server 24
Tạo đĩa boot và tiến trình boot 25
Các cách cài đặt và phương pháp của chúng 26
Cài đặt đĩa (Disk setup) 26
Sự chọn lựa những package (gói dữ liệu) riêng lẻ 32
Các gói chương trình không nên cài đặt do các nguyên nhân bảo mật 34
Làm thế nào sử dụng những lệnh RPM 38
Khởi động và dừng những dịch vụ daemon 39
Các phần mềm cần phải loại bỏ sau khi tiến trình cài đặt của server hoàn thành 40
Các phần mềm phải được cài đặt sau sự cài đặt của server 43
Những chương trình được cài đặt trên server của bạn 46
Định màu trên terminal của bạn 51
Cập nhật phần mềm mới nhất 52

PHẦN 2 : CÁC VẤN ĐỀ LIÊN QUAN VỀ BẢO MẬT VÀ TỐI ƯU **53**
Chương 3: TỔNG QUAN VỀ BẢO MẬT LINUX **55**

Chương 4: TỔNG QUAN VỀ SỰ TỐI ƯU LINUX 91

Chương 5: CẤU HÌNH VÀ XÂY DỰNG MỘT KERNEL 109

Linux Kernel 110

Làm một đĩa mềm boot cấp cứu 111

Bảo mật Kernel 114

Cấu hình Kernel 117

Cài đặt Kernel mới: 124

Xóa những chương trình, tập tin, dòng có liên quan tới các module 127

Tạo một đĩa cứu nguy mới (rescue floppy): 129

Tạo đĩa mềm boot cấp cứu (emergency boot floppy disk) 129

Cập nhật thư mục "/dev" 130

PHẦN 3 : LIÊN KẾT MẠNG 131

Chương 6: QUẢN LÝ HỆ THỐNG MẠNG LINUX TCP/IP 133

Cài đặt nhiều card mạng trên một máy 134

Các tập tin liên quan đến chức năng của mạng 135

Cấu hình mạng TCP/IP bằng tay với giao diện dòng lệnh 140

Chương 7: LIÊN KẾT FIREWALL 145

Linux IPCHAINS 146

Xây dựng một nhân của hệ thống (Kernel) với sự hỗ trợ IPCHAINS Firewall 150

Một vài sự giải thích về các nguyên lý được sử dụng trong các tập tin firewall script 150

Các tập tin firewall scripts 153

Cấu hình tập tin script "/etc/rc.d/init.d/firewall" cho Web Server 154

Cấu hình tập tin script "/etc/rc.d/init.d/firewall" cho Mail Server 167

Tóm tắt về IPCHAINS 178

Chương 8: LIÊN KẾT FIREWALL VỚI HỖ TRỢ TÍNH NĂNG GIẢ MẠO (MASQUERADING) IP VÀ CHUYỂN TIẾP (FORWARDING) 185

Giả mạo và chuyển tiếp IP trong Linux 186

Xây dựng một Kernel với Firewall hỗ trợ giả mạo IP và chuyển tiếp 186

Sự cấu hình tập tin script cho Gateway Server 189

Từ chối truy nhập vào hệ thống từ một vài địa chỉ IP 203

Công cụ quản trị IPCHAINS 204

PHẦN 4 : SỰ THAM KHẢO CÁC PHẦN MỀM LIÊN QUAN 205

Chương 9: CHỨC NĂNG BIÊN DỊCH.....207

Chức năng biên dịch trong Linux.....208

Các gói dữ liệu cần thiết.....208

Tại sao chúng ta chọn sử dụng tarballs?.....210

Biên dịch các phần mềm trên hệ thống của bạn.....210

Xây dựng và cài đặt phần mềm trên hệ thống của bạn.....212

Soạn thảo các tập tin với trình soạn thảo vi.....213

Một vài ghi chú sau cùng.....213

Chương 10: PHẦN MỀM BẢO MẬT (Công cụ giám sát).....215

Linux sXid.....216

Cấu hình.....217

Các công cụ quản trị sXid.....219

Linux Logcheck.....221

Giới thiệu.....221

Cấu hình.....223

Linux PortSentry.....225

Cấu hình.....227

Khởi động PortSentry.....233

Chương 11: PHẦN MỀM BẢO MẬT (Các dịch vụ mạng).....235

Linux OpenSSH Client/Server.....236

Cấu hình.....239

Cấu hình OpenSSH để dùng TCP-Wrappers inetd super server.....245

Cấu hình OpenSSH cho mỗi người sử dụng.....246

Các công cụ OpenSSH cho người sử dụng.....248

Linux SSH2 Client/Server.....251

Cấu hình.....252

Cấu hình Sshd2 để dùng TCP-Wrappers inetd super server.....260

Cấu hình SSH2 cho mỗi người sử dụng.....261

Các công cụ SSH2 cho người sử dụng.....262

Chương 12: PHẦN MỀM BẢO MẬT (Tính toàn vẹn hệ thống).....265

Linux Tripwire 2.2.1.....266

Cấu hình.....271

Bảo vệ Linux với Tripwire.....278

Các lệnh.....279

Linux Tripwire ASR 1.3.1 283
Cấu hình 285
Sự an toàn của Tripwire 288
Các lệnh 288

Chương 13: PHẦN MỀM BẢO MẬT (Sự quản lý và sự giới hạn) 291

Linux GnuPG 292
Các lệnh 293
Thiết đặt hạn ngạch (Quota) trên hệ thống Linux 299
Xây dựng một kernel hỗ trợ Quota 299
Thay đổi tập tin "/etc/fstab" 299
Tạo các tập tin "quota.user" và "quota.group" 300
Ấn định Quota cho các người dùng và các nhóm 302
Các lệnh 304

Chương 14: PHẦN MỀM SERVER (Dịch vụ mạng BIND/DNS) 307

Server Linux DNS và BIND 308
Cấu hình 311
Caching-Only Name Server 312
Primary master name server 314
Secondary slave name server 317
An toàn cho ISC BIND/DNS 321
Các công cụ quản trị DNS 328
Các công cụ sử dụng DNS 329

Chương 15: SERVER SOFTWARE (Dịch vụ Mail Network) 333

Linux sendmail server 334
Cấu hình 340
Bảo mật Sendmail 355
Các công cụ quản trị Sendmail 360
Các công cụ sử dụng Sendmail 361
Linux Imap & Pop Server 363
Cấu hình 366
Cho phép dịch vụ IMAP or POP thông qua tcp-wrappers inetd super server 367
Bảo mật cho IMAP/POP 368

Chương 16: PHẦN MỀM SERVER (Dịch vụ mã hóa) 371

Linux OpenSSL Server 372
Cấu hình 377

Các lệnh 384
Bảo vệ OpenSSL 388
LINUX FreeS/WAN VPN 391
Sự cấu hình 397
Cấu hình RSA private keys secrets 403
Yêu cầu cài đặt Network cho IPSec 410
Kiểm tra sự cài đặt 414

Chương 17: PHẦN MỀM SERVER (Dịch vụ về cơ sở dữ liệu) 421
Linux OpenLDAP Server 422
Sự cấu hình 426
Bảo mật OpenLDAP 431
Các công cụ bảo trì và khởi tạo OpenLDAP 432
Các công cụ cho các người sử dụng OpenLDAP 435
Đặc trưng Address Book client của Netscape cho LDAP 435
Linux PostgreSQL Database Server 441
Cài đặt cơ sở dữ liệu từ tài khoản super-user PostgreSQL của bạn 445
Công việc cấu hình 446
Các lệnh 449

Chương 18: PHẦN MỀM SERVER (Dịch vụ Proxy) 455
Linux Squid Proxy Server 456
Sử dụng thư viện GNU malloc để cải tiến khả năng thực hiện cache của Squid 458
Cấu hình 461
Bảo mật Squid 471
Tối ưu Squid 472
Trình tiện ích cachemgr.cgi của Squid 472
Cấu hình proxy cho Netscape để dùng với Squid 474

Chương 19: PHẦN MỀM SERVER (Dịch vụ Web) 477
Linux MM – Shared Memory Library for Apache 478
Linux Apache Web Server 481
Biên dịch và cài đặt Apache 482
Cấu hình 490
Ngôn ngữ script PHP4 server-side 500
Môđun Perl Devel::Symdump 501
Thư viện Perl CGI.pm 504
An toàn cho Apache 505

Chạy Apache trong một vùng riêng biệt chroot (jail chroot) 508
Tối ưu Apache 516

Chương 20: CÀI ĐẶT APACHE VỚI CÁC THÀNH PHẦN TÙY CHỌN...527

Linux Webalizer 528
Cấu hình 530
Báo cho Apache biết về thư mục output của Webalizer 532
Chạy Webalizer bằng tay trong lần đầu tiên 532
Chạy Webalizer một cách tự động với một cron job 533
Linux FAQ-O-Matic 535
Báo cho Apache biết vị trí của các tập tin Faq-O-Matic 537
Cấu hình phần mềm FAQ-O-Matic của bạn 538
Linux Webmail IMP 543
Cài đặt PHPLib, thành phần được yêu cầu bởi chương trình Horde của Webmail IMP 544
Cấu hình và tạo cơ sở dữ liệu SQL cho Webmail IMP 546
Cấu hình tập tin "php.ini" cho PHP4 của bạn 549
Cấu hình Apache để nhận biết Webmail IMP 550
Cấu hình Webmail IMP thông qua trình duyệt web của bạn 550

Chương 21: PHẦN MỀM SERVER (Dịch vụ chia sẻ tập tin) 553

Linux Samba Server 554
Cấu hình 555
Tạo một tập tin mật khẩu (password) Samba được mã hoá cho các máy client của bạn 565
Bảo mật Samba 569
Tối ưu Samba 569
Các công cụ quản trị Samba 571
Các công cụ của những người sử dụng Samba 572
Linux FTP Server 575
Thiết lập một tài khoản FTP cho mỗi user không có shells 578
Cài đặt một môi trường sử dụng chroot 580
Cấu hình 582
Cấu hình ftpd để dùng tcp-wrappers inetd super server 588
Các công cụ quản trị FTP (FTP Administrative Tools) 589
Bảo mật FTP 590

PHẦN 5 : CÁC VẤN ĐỀ LIÊN QUAN VỀ SAO LƯU DỮ LIỆU 593

Chương 22: CÁC THỦ TỤC VỀ SAO LƯU VÀ KHÔI PHỤC DỮ LIỆU ..595

Chương trình sao chép dự phòng tar 597
Tạo các sao chép dự phòng với tar 597
Các công việc sao lưu tự động được tạo với tar 600
Phục hồi các tập tin với tar 602
Chương trình sao chép dự phòng dump 603
Tạo các sao lưu với dump 605
Phục hồi lại các tập tin với dump 608
Sao chép dự phòng và phục hồi thông qua mạng 610

PHẦN 6 : CÀI ĐẶT VÀ CẤU HÌNH MỘT SỐ PHẦN MỀM TRÊN RED HAT LINUX 7.2 613

Chương 23: WEB SERVER APACHE 1.3.20 615

Biên dịch và cài đặt apache 617
Cài đặt Apache từ tập tin apache-1.3.20-16.i386.rpm 618
Cấu hình tập tin /etc/httpd/conf/httpd.conf 619

Chương 24: CÀI ĐẶT LINUX FreeS/WAN 635

Biên dịch và đưa FreeS/WAN vào Kernel 638
Định lại cấu hình và cài đặt Kernel để hỗ trợ FreeS/WAN 638
Biên dịch và cài đặt Kernel mới với FreeS/WAN 639
Sự cấu hình cho Freeswan 640
Yêu cầu cài đặt Network cho IPSec 650
Kiểm tra sự cài đặt 655

Chương 25: CÀI ĐẶT SQUID PROXY SERVER 659

Proxy Caching 660
Network Address Translation (NAT) 661
Khác biệt giữa một bộ lọc gói dữ liệu (Packet Filter) và một Proxy Server 662
Sự thực thi Squid Web Proxy Cache Server 663
Khởi động và kiểm tra Squid 668
Cấu hình các proxy client 668

Chương 26: CÀI ĐẶT OPENSSH CLIENT/SERVER 675

Giới thiệu về SSH protocol 676
Các tầng của bảo mật SSH 678

Cài đặt và cấu hình OpenSSH 679
Cấu hình một OpenSSH client 681
Cấu hình ssh-agent với GNOME 684
Cấu hình ssh-agent 685

Chương 27: CÀI ĐẶT VÀ SỬ DỤNG SAMBA 689

Cài đặt Samba 690
Thiết đặt security = domain trong Samba 2.x 691
Cấu hình Samba 2.2 như một Primary Domain Controller 693

PHẦN 7 : CÁC PHỤ LỤC 703

Phụ lục A: NHỮNG LẮT LÉO (TWEAKS), MẸO (TIPS) VÀ CÔNG VIỆC CỦA NGƯỜI QUẢN TRỊ 705

Phụ lục B: Các RFC đang tồn tại 709

PHẦN 1

HƯỚNG DẪN CÀI ĐẶT LINUX

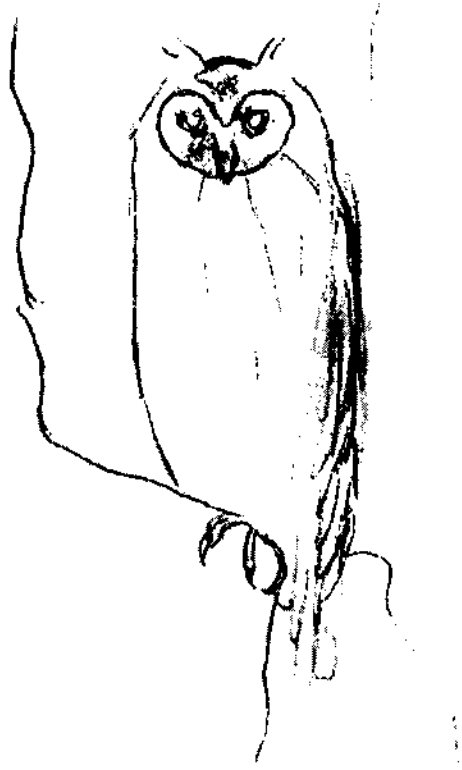
Trong phần này chúng ta sẽ đề cập đến những chương sau:

- **Giới thiệu về Linux**
- **Cài đặt Server Linux**

Chương 1: GIỚI THIỆU VỀ LINUX

Các phần chính trong chương này:

- Linux là gì?
- Một vài nguyên nhân khiến chúng ta sử dụng Linux.
- Gạt bỏ một vài sự lo ngại, không chắc chắn, nghi ngờ về Linux.



GIỚI THIỆU VỀ LINUX

Linux là gì?

Linux là một hệ điều hành được tạo ra tại Viện đại học Helsinki ở Phần Lan bởi một sinh viên trẻ tên là Linus Torvalds. Tại thời điểm này sinh viên đã làm việc trên những hệ thống Unix với những phần cứng rất đắt tiền. Bởi vì Linus Torvalds có ít tiền và cũng cần làm việc ở nhà nên Linus đã quyết định tạo một bản copy của hệ thống Unix để chạy trên những phần cứng rẻ tiền hơn chẳng hạn như một IBM PC. Linus đã bắt đầu công việc trong năm 1991 khi Linus đã đưa ra phiên bản 0.02 và đã làm việc tốt đến năm 1994 thì version 1.0 của nhân Linux (Linux kernel) đã được đưa ra. Tại thời điểm viết cuốn sách này version của kernel là 2.4.7 với Red Hat Linux 7.2 và đang tiếp tục được phát triển trong thời gian tới.

Hệ điều hành Linux được phát triển dưới dạng GNU General Public License (cũng được hiểu như GNU GPL) và mã nguồn (source code) được phân phát một cách miễn phí cho mọi người, bất kỳ ai cũng có thể download mã nguồn Linux thông qua các website trên Internet, CD-ROM của Linux được bán rộng rãi ở các cửa hàng và công ty máy tính với những giá cả có thể khác nhau do chất lượng của CD và phí hỗ trợ kỹ thuật. Linux có thể sử dụng một cách rộng rãi với nhiều mục đích khác nhau như liên kết mạng, phát triển phần mềm và là một công cụ hữu hiệu của người sử dụng đầu cuối. Linux cũng được nghiên cứu khá rộng rãi ở nhiều nơi, sự thay đổi về giá thấp so với nhiều hệ thống đắt tiền khác vì chúng ta có thể cài đặt trên nhiều máy tính mà không phải trả tiền bản quyền.

Một vài nguyên nhân dẫn đến việc sử dụng Linux

Không có tiền bản quyền tác giả và sử dụng Linux miễn phí, mã nguồn có thể thay đổi theo nhu cầu, việc thay đổi có thể tạo ra những sản phẩm tốt hơn và bạn có thể bán sản phẩm của mình cho người khác nhưng tác giả ban đầu (original author) vẫn giữ tác quyền và bạn sẽ phải cung cấp mã nguồn bạn đã thay đổi cho cộng đồng.

Mã nguồn kernel (nhân hệ điều hành) mở, khả chuyển, điều này có ý nghĩa rằng Linux có thể chạy trên nhiều loại CPU và các phần cứng khác nhau hơn bất kỳ hệ điều hành nào khác.

Xu hướng gần đây của công nghiệp phần cứng và phần mềm đã đẩy người tiêu dùng mua những máy tính có tốc độ nhanh hơn với nhiều bộ nhớ và dung lượng lưu trữ lớn của đĩa cứng. Linux không bị tác động bởi sự định hướng của nền công nghiệp này vì khả năng chạy trên bất kỳ loại máy tính nào của Linux, ngay cả những hệ máy tính trước đây như 486 với sự giới hạn về dung lượng bộ nhớ.

Giống như các hệ UNIX trước đó, Linux là một hệ điều hành đa nhiệm, chúng ta có thể làm cùng lúc nhiều công việc khác nhau. Linux sử dụng sự quản lý

bộ nhớ tinh vi và điều khiển tất cả các tiến trình, điều này có ý nghĩa rằng nếu một chương trình nào đó bị hỏng chúng ta có thể loại bỏ nó và tiếp tục làm những công việc khác.

Một tiện ích khác của Linux là gần như miễn dịch với sự tấn công của các loại virus, việc này đã xảy ra với các hệ thống khác. Hiện tại chỉ có hai loại virus có tác động đến Linux và cả hai loại này chưa xuất hiện ở Việt Nam chúng ta.

Xua tan đi sự lo ngại, sự không chắc chắn và sự nghi ngờ về Linux.

Linux là một hệ điều hành đang được ưa chuộng?

Với sự phát triển và thay đổi liên tục về công nghệ và giao diện, Linux đã làm cho nhiều công ty, chính phủ, các tổ chức và người sử dụng quan tâm đến. Hàng ngàn website trên thế giới đã sử dụng Linux như một webserver, nhiều công ty lớn như HP, Ericsson đã sử dụng Linux như một hệ điều hành chạy trên những sản phẩm của họ mà chúng ta đã biết thuật ngữ Linux Embedded.

Nhiều công ty đã và đang phát triển những sản phẩm hỗ trợ cho Linux, một số thì phát triển ứng dụng chạy trên Linux. Với việc hầu như miễn phí hoàn toàn và có nhiều ứng dụng sẵn có đáp ứng được những yêu cầu công việc hằng ngày đã có nhiều người sử dụng Linux trên PC của họ. Một điểm khá đặc biệt là gần đây chính phủ của nhiều quốc gia đã chọn Linux trong chiến lược phát triển công nghệ bảo mật (security) chống lại sự tấn công của các tin tặc (hacker).

Không ai hỗ trợ Linux?

Điều này không còn đúng nữa, có một điều khiến người dùng yên tâm là với trên 12.000 trang tài liệu hướng dẫn sử dụng, bảo trì khá chi tiết luôn đi kèm khi bạn tiến hành cài đặt Linux một cách đầy đủ (full) là điều kiện tốt và dễ dàng trong việc học Linux. Có nhiều phiên bản Linux đã phát triển thành những sản phẩm thương mại có độ ổn định cao, đáp ứng nhiều công việc như Red Hat Linux, Caldera, SuSE. Bên cạnh đó có hàng ngàn website và tài liệu hỗ trợ cho Linux.

Với những việc làm trên, ta thấy Linux sẽ là một hệ điều hành của tương lai đáp ứng được nhiều yêu cầu của người tiêu dùng trên khắp thế giới.

Chương 2: CÀI ĐẶT SERVER LINUX

Trong chương này chúng ta sẽ đề cập đến các vấn đề sau:

- Sự hiểu biết về phần cứng trong server.
- Tạo đĩa boot và tiến trình boot.
- Các cách cài đặt và phương pháp của chúng.
- Cài đặt đĩa (disk).
- Các thành phần cài đặt.
- Sự chọn lựa những package (gói dữ liệu) riêng lẻ.
- Khởi động và dừng các dịch vụ daemon.
- Các gói chương trình không nên cài đặt do các nguyên nhân bảo mật.
- Các phần mềm phải được cài đặt sau sự cài đặt của server.
- Những chương trình được cài đặt trên server.
- Định màu trên máy (terminal) của bạn.
- Cập nhật các phần mềm mới nhất.



Hướng dẫn cài đặt Linux

Chương này chủ yếu dành cho những ai chưa quen hoặc chưa biết với tiến trình cài đặt hệ thống Linux trên IBM PC, nếu bạn đã thông thạo với việc này thì hãy bỏ qua vì sẽ làm mất thời gian của các bạn cho một việc không bổ ích.

Chúng tôi chuẩn bị chương này theo một ngữ cảnh rằng việc cài đặt Red Hat Linux của bạn được tiến hành từ CD-ROM chứa chương trình cài đặt Red Hat Linux (Red Hat Linux installation). Từng phần trong chương này sẽ diễn giải và hướng dẫn cho bạn thực hiện, nhiều màn hình khác nhau của tiến trình cài đặt sẽ xuất hiện trong suốt quá trình cài đặt của bạn sau khi bạn đưa đĩa khởi động Red Hat (Red Hat boot diskette) vào máy tính, và bạn sẽ theo những bước được mô tả dưới đây.

Theo thời gian hệ điều hành Red Hat Linux luôn được cập nhật với những phiên bản mới và tiến hành thêm, thay đổi hoặc tháo bỏ một vài chương trình cũng như thay đổi một vài vị trí, nội dung hoặc đặc trưng của những tập tin được thực hiện trong sự sắp đặt của từng ứng dụng. Gần đây (6-2002) Red Hat Linux đã được cập nhật tới phiên bản 7.3 với nhiều thay đổi so với phiên bản 6.1. Tuy thế những thông tin được trình bày trong chương này không mất tính chính xác và khả thi, chúng tôi sẽ diễn giải dựa trên việc trình bày với Red Hat Linux 6.1 và Linux 6.2, một vài khía cạnh có thể áp dụng cho Linux 7.2 hay Linux 7.3 cho những ai sẽ cập nhật (update) hoặc cài đặt (install) nó. Tương ứng bất kỳ những phần trình bày nào trong chương này nói đến phiên bản 6.1 sẽ ám chỉ đến Red Hat Linux 6.1 (Cartman), và bất kỳ phần nào chúng tôi nói về phiên bản 6.2 sẽ được ám chỉ đến Red Hat Linux 6.2 (Zoot)

Những qui định sau đây sẽ đơn giản hoá việc giải thích của chương này.



Hình tượng này áp dụng cho Red Hat Linux phiên bản 6.1 và 6.2.



Hình tượng này chỉ áp dụng cho Red Hat Linux 6.1.

Hình tượng này chỉ áp dụng cho Red Hat Linux 6.2.

Chúng tôi biết rằng nhiều tổ chức và công ty sử dụng những phiên bản khác nhau của hệ điều hành Linux và có thể chạy nhiều ứng dụng khác nhau trên chúng. Thỉnh thoảng, có thể có những khó khăn trong việc cập nhật những phiên bản mới nhất khi những người sử dụng chạy những dịch vụ trên server Linux 24 giờ một ngày. Với những qui ước đơn giản trên, bất kỳ người nào duy trì và sử dụng phiên bản 6.1 của Red Hat Linux sẽ luôn tìm thấy chính xác những thông tin liên quan đến những việc họ cần để tiến hành cập nhật.

Sự hiểu biết về phần cứng trong server

Sự hiểu rõ phần cứng máy tính của bạn là yếu tố cần thiết cho sự cài đặt thành công của Linux, vì vậy trước khi cài đặt bạn nên tìm hiểu và làm quen với những phần cứng trên chính máy bạn và hãy trả lời những câu hỏi dưới đây:

1. Bạn có bao nhiêu đĩa cứng?

2. Dung lượng của mỗi đĩa cứng?.
3. Nếu bạn có nhiều hơn một đĩa cứng, ổ đĩa nào là chính (primary)?
4. Loại đĩa cứng nào bạn có? Ví dụ IDE, SCSI.
5. Bạn có bao nhiêu RAM?
6. Bạn có card SCSI không? Nếu có thì nhà sản xuất là ai?
7. Bạn có hệ thống RAID không? Nếu như vậy ai làm nó và mô hình nó là gì?
8. Loại mouse nào bạn có? (ví dụ PS/2, Microsoft, Logitech...).
9. Mouse của bạn có bao nhiêu nút, hai hay ba?
10. Nếu bạn có serial mouse (mouse nối qua cổng COM), cổng COM nào nó được kết nối? Ví dụ COM1.
11. Card màn hình của bạn là loại gì và hiệu gì? Card màn hình có bao nhiêu RAM (ví dụ 4MB)?
12. Bạn có ý định kết nối mạng không? Nếu có thì những thông số sau đây sẽ là gì?
    a. Địa chỉ IP?
    b. Netmask?
    c. Địa chỉ Gateway?
    d. Địa chỉ IP của server dịch vụ tên miền (DNS Server)?
    e. Tên miền?
    f. Tên máy tính của bạn?
    g. Kiểu của card mạng?

Tạo đĩa boot và tiến trình boot

All Version Điều đầu tiên cần làm là tạo một đĩa mềm cài đặt (installation diskette) cũng được hiểu như một đĩa khởi động. Nếu bạn mua CD-ROM Red Hat Linux chính thức bạn sẽ thấy có một đĩa mềm với nhãn là "Boot Diskette" trong hộp, bạn không cần tạo lại nó và có thể bắt đầu cài đặt Linux từ đĩa này. Tuy nhiên bạn có thể gặp trường hợp là bạn không thể cài đặt Linux từ đĩa mềm này (do đĩa mềm của bạn bị hư hoặc không đúng là Boot Diskette), nếu điều này xảy ra thì bạn cần có một đĩa mềm khác tương tự như cái vừa rồi và có thể cài đặt được Linux. Trong trường hợp này bạn có thể tải (download) đĩa này tại địa chỉ http://www.redhat.com/errata. Có một điều tôi muốn nói là bạn có thể cài đặt Linux bằng cách boot trực tiếp từ CD-ROM nếu như máy tính của bạn có hỗ trợ boot từ CD-ROM, khi đó bạn có thể đi thẳng qua bước hai. Nếu như máy tính của bạn không hỗ trợ boot từ CD-ROM, chỉ có thể boot từ đĩa mềm (floopy disk) hoặc đĩa cứng (HDD), bạn có thể tạo một đĩa mềm khởi động (boot disketle) bằng cách như sau.

Bước 1:

Trước khi tạo đĩa boot, đưa CD-ROM Red Hat Linux vào trong ổ CD trên máy tính của bạn đang chạy hệ điều hành Windows, bạn cũng có thể tạo đĩa này trên máy tính đang chạy MS-DOS nhưng yêu cầu là phải hiểu ổ đĩa CD của bạn. Tôi giả sử bạn đang ở trong môi trường MS Window và ổ đĩa CD được hiểu như ổ D. Khi chương trình hỏi cho tên file, gõ vào **boot.img** cho đĩa boot.

- Mở Command Prompt dưới Windows: Start | Programs | Command Prompt

```
C:\> d:
D:\> cd \dosutils
D:\dosutils> rawrite
Enter disk image source file name: ..\images\boot.img
Enter target diskette drive: a:
Please insert a formatted diskette into A drive: and press ENTER

D:\dosutils>
```

Chương trình rawrite.exe hỏi tên tập tin của disk image (ảnh đĩa): Gõ vào **boot.img** và đưa đĩa mềm vào ổ đĩa A. Sau đó nó sẽ hỏi đĩa nào sẽ được ghi vào, gõ vào a: bạn đã hoàn thành bước này và bạn có một đĩa mềm với tên là "**Red Hat boot disk**"

Bước 2

Đưa đĩa boot vừa tạo vào trong đĩa A trên máy tính bạn muốn cài đặt Red Hat Linux và khởi động lại máy. Ở dấu nhắc boot nhấn phím Enter để tiếp tục boot và theo ba bước căn bản dưới đây:

- ❖ Chọn ngôn ngữ.
- ❖ Chọn kiểu bàn phím.
- ❖ Chọn kiểu con chuột.

Các cách cài đặt và phương pháp của chúng

Red Hat Linux 6.1 và 6.2 có 4 lớp cài đặt khác nhau đó là:

- ➢ GNOME Workstation
- ➢ KDE Workstation
- ➢ Server
- ➢ Custom

Cả ba lớp cài đặt đầu tiên (GNOME Workstation, KDE Workstation và Server) cấp cho bạn sự lựa chọn đơn giản của tiến trình cài đặt, ở đó máy tính tự động làm hết mọi thứ và bạn mất đi đáng kể tính linh hoạt trong việc cấu hình mà chúng ta không nên bỏ qua và sẽ được đề cập chi tiết trong những bước tiếp theo.

Vì lý do trên chúng ta sẽ chọn cách cài đặt **custom.** Cách này cho phép bạn chọn những dịch vụ nào sẽ được thêm vào và làm thế nào để phân hoạch hệ thống.

Với ý tưởng là sử dụng tối thiểu các gói phần mềm trong khi đảm bảo tối đa hiệu năng cho hệ thống. Bớt đi những phần mềm hiện có trong Linux, khai thác những tiềm năng bảo mật và loại trừ những lỗ hổng có thể xuất hiện.

➢ Chọn "**Custom**" và click **Next**

Cài đặt đĩa (Disk setup)

All Version Chúng tôi giả sử rằng bạn đang cài đặt server Linux mới trên một ổ đĩa cứng mới không có một hệ điều hành nào được cài đặt trước đó. Một chiến lược phân hoạch tốt là tạo từng partion riêng rẽ cho mỗi hệ thống tập tin chính. Việc

này làm tăng khả năng bảo mật và ngăn chặn kiểu tấn công "từ chối dịch vụ" (denial of service) hoặc khai thác của những chương trình SUID.

Việc tạo nhiều partions sẽ cung cấp cho bạn có những thuận lợi sau:

- Bảo vệ chống lại các tấn công theo kiểu denial of service.
- Chống lại tấn công của những chương trình SUID.
- Khởi động nhanh hơn.
- Dễ dàng backup và quản lý việc nâng cấp các ứng dụng khi cần thiết
- Khả năng điều khiển tốt hơn của việc kết gắn các hệ thống tập tin (mount tập tin system).
- Giới hạn khả năng tăng về kích thước của các hệ thống tập tin (file system).

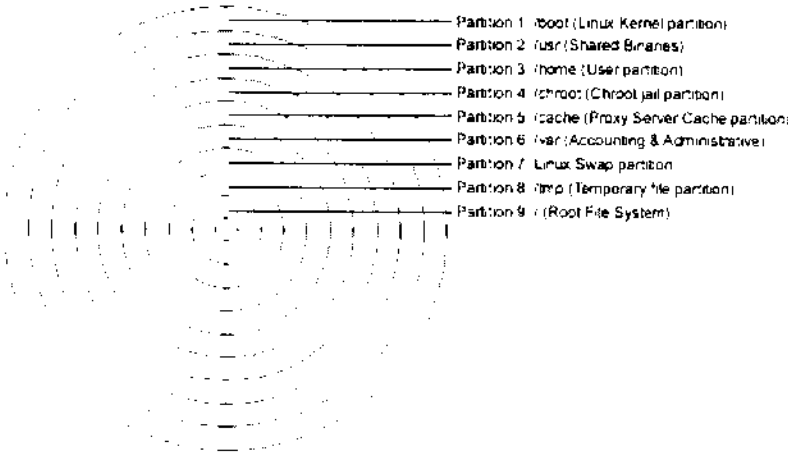
Cảnh báo : *Nếu hệ thống tập tin hoặc hệ điều hành trước đây có sẵn trên đĩa cứng và máy tính mà bạn muốn cài đặt hệ thống Linux, tôi đề nghị rằng bạn nên sao lưu (backup) hệ thống tập tin hiện tại trước khi tiến hành việc phân hoạch đĩa.*

Bước 1:

Để đạt hiệu quả cao, ổn định và an toàn bạn nên tạo các partition như những partition được liệt kê dưới đây trên máy tính của bạn. Chúng tôi cũng giả sử rằng thực tế bạn có ổ cứng (Hardisk) SCSI 3.2 GB để phân hoạch và dĩ nhiên bạn sẽ cần điều chỉnh kích thước partition tùy theo nhu cầu bạn cần và kích thước đĩa bạn đang có.

Những partition bạn phải tạo trên hệ thống của bạn:

/boot	5MB	Tất cả các kernel images thì được giữ ở đây.
/usr	512MB	Partition này cần phải lớn trước khi tất cả các chương trình ở dạng binary được cài đặt ở đây.
/home	1146MB	Cân đối số người sử dụng bạn có ý định tạo ra trên máy này. Ví dụ 10MB/người như vậy với 114 người cần 1140MB.
/chroot	256MB	Nếu bạn không muốn cài đặt những chương trình trong môi trường không tự do (jail environment) chẳng hạn như DNS tức là môi trường mà chỉ có root mới có quyền thực thi.
/cache	256MB	Đây là partition lưu trữ của proxy server (ví dụ Squid).
/var	256MB	Chứa đựng những tập tin thay đổi khi hệ thống chạy bình thường (ví dụ các tập tin log).
<Swap>	128MB	Đây là Swap partition được coi như bộ nhớ ảo của hệ thống, bạn nên chia kích thước của partition này lớn hơn hoặc bằng dung lượng RAM hiện có trên máy của bạn.
/tmp	256MB	Partition chứa những tập tin tạm thời.
/	256MB	Root partition của chúng ta.



All major file systems are on separate partitions

Tất cả các tập tin hệ thống chính trên các partition riêng lẻ

Chúng ta có thể tạo thêm hai partition đặc biệt là "/chroot" và "/cache", partition /chroot có thể được sử dụng cho DNS server, Apache Server và những chương trình tương lai khác theo dạng như DNS và Apache. Partition /cache có thể được sử dụng cho Squid proxy server. Nếu bạn không có ý định cài đặt Squid Proxy server, bạn không cần tạo partition /cache.

Đặt /tmp và /home trên các partition riêng biệt thì rất hay và có tính chất bắt buộc nếu người sử dụng có shell truy cập tới server (sự bảo vệ chống lại những chương trình SUID), ngăn chặn những chương trình này xâm nhập vào những partition riêng lẻ và cũng ngăn cản những người sử dụng làm suy yếu hoạt động của bất kỳ tập tin hệ thống nào (denial of service attack) trên server. Đặt /var và /usr trên các partition riêng rẽ cũng là một ý rất hay bởi vì cách ly partition /var sẽ làm cho partition root của bạn không bị tràn đầy (denial of service attack).

Trong cấu hình partition chúng ta sẽ dành riêng 256MB đĩa trống cho những chương trình chuyển đổi root (chrooted program) giống Apache, DNS và những chương trình khác. Việc này là cần thiết bởi vì những tập tin tài liệu gốc, những tập tin dạng nhị phân, những chương trình liên quan tới Apache sẽ được cài đặt trong partition này nếu bạn có ý định chạy Web server Apache trong vùng riêng biệt đó. Lưu ý cân đối kích thước thư mục Apache trên partition chroot và kích thước của những tài liệu gốc. Nếu bạn không có ý định cài đặt và sử dụng Apache trên server bạn, có thể giảm bớt kích thước của partition này xuống khoảng 10MB và chỉ sử dụng cho DNS là dịch vụ luôn cần trong môi trường chroot vì lý do bảo mật.

Kích thước tối thiểu của các partition

Đây là cấu hình một phân hoạch tối thiểu tính trên megabytes, để một hệ thống Linux hoạt động được. Kích thước của các partition được liệt kê dưới đây thì thật là nhỏ. Sự phân hoạch này có thể phù hợp với những loại đĩa cứng cũ khoảng

512MB, và thường thấy trong các máy tính cũ x486. Chúng tôi trình bày cho bạn thông tin về những yêu cầu tối thiểu của các partition và tất nhiên bạn có thể thay đổi cho phù hợp với đĩa cứng bạn có.

/	35MB
/boot	5MB
/chroot	10MB
/home	100MB
/tmp	30MB
/usr	232MB
/var	25MB
/swap	50MB

Disk Druid

Disk Druid là một chương trình được sử dụng để phân chia đĩa cứng cho bạn. Chọn "Add" để thêm một partition mới, "Edit" để hiệu chỉnh một partition, "Delete" để xoá một partition và "Reset" để xác lập những partition về lại trạng thái ban đầu. Khi bạn thêm một partition mới, một cửa sổ mới xuất hiện trên màn hình và công việc của bạn là chọn những thông số cho partition đó. Sự khác nhau của các thông số là:

Mount Point: Vị trí trong hệ thống tập tin bạn muốn mount partition mới của bạn tới.

Size (Megs): Kích thước của partition mới tính trên megabytes.

Partition Type: Có hai kiểu chính là Linux native dùng cho Linux filesystem và Swap dùng cho Linux Swap partition.

Nếu bạn có đĩa cứng loại SCSI thì tên thiết bị sẽ là /dev/sda và nếu bạn có đĩa cứng kiểu IDE thì tên sẽ là /dev/hda. Nếu bạn đang cần một hệ thống có hiệu năng và độ ổn định cao thì SCSI là sự chọn lựa tốt nhất.

Linux đặt tên các partition dựa vào việc sử dụng sự kết hợp của các mẫu tự và số. Linux sử dụng một kiểu đặt tên khá mềm dẻo và linh hoạt hơn so với những hệ thống khác. Đây là một bản tóm tắt:

- Hai ký tự đầu tiên: Hai ký tự đầu tiên của tên partition chỉ định kiểu của thiết bị partition tồn tại trên đó. Như bạn thường thấy "hd" ý nói cho IDE disk và "sd" ý nói cho SCSI disk
- Ký tự tiếp theo chỉ định partition được mở. Ví dụ "/dev/hda" ngụ ý là Harddisk thứ nhất, "/dev/hdb" ngụ ý là harddisk thứ hai

Những thông tin trên bạn nên ghi nhớ vì nó sẽ giúp bạn dễ dàng hơn trong việc phân chia các partition trên đĩa.

Partition Swap

Partition Swap được sử dụng để hỗ trợ bộ nhớ ảo. Nếu máy tính của bạn có 16MB RAM hoặc ít hơn bạn phải tạo một partition swap, ngay cả khi bạn có bộ

nhớ lớn bạn cũng nên tạo partition swap. Kích thước tối thiểu của partition swap nên bằng hoặc lớn hơn với dung lượng RAM có trên máy tính của bạn. Kích thước lớn nhất có thể sử dụng cho partition swap là 1GB (với kernel 2.2) cho nên nếu bạn tạo một partition swap lớn hơn 1GB thì phần còn lại sẽ trở nên vô ích. Lưu ý rằng bạn có thể tạo và sử dụng nhiều hơn một swap partition mặc dù việc này thì thường chỉ cần thiết khi cài đặt những server cực lớn.

Dưới đây là một ví dụ:

Tạo các partition được liệt kê dưới đây trên hệ thống của bạn (đây là partition chúng ta sẽ cần cho tiến trình cài đặt server của chúng ta). Những lệnh được thực hiện dưới Disk Druid:

Add
Mount Point: /boot <- thư mục /boot trên máy server của bạn.
Size (Megs): 5
Partition Type: Linux Native
Ok

Add
Mount Point: /usr <- thư mục /usr trên máy server của bạn.
Size (Megs): 512
Partition Type: Linux Native
Ok

Add
Mount Point: /home <- thư mục /home trên máy server của bạn.
Size (Megs): 1146
Partition Type: Linux Native
Ok

Add
Mount Point: /chroot <- thư mục /chroot trên máy server của bạn.
Size (Megs): 256
Partition Type: Linux Native
Ok

Add
Mount Point: /cache <- thư mục /cache trên máy server của bạn.
Size (Megs): 256
Partition Type: Linux Native
Ok

Add
Mount Point: /var <- thư mục /var trên máy server của bạn.
Size (Megs): 256

Partition Type: Linux Native
Ok

Add
Mount Point: <- partition swap trên máy server của bạn. (phần Mount Point để trống).
Size (Megs): 128
Partition Type: Linux Swap
Ok

Add
Mount Point: /tmp <- thư mục /tmp trên máy server của bạn.
Size (Megs): 256
Partition Type: Linux Native
Ok

Add
Mount Point: / <- thư mục / (gốc) trên máy server của bạn.
Size (Megs): 256
Partition Type: Linux Native
Ok

Sau khi tạo các partition trên hard disk hoàn thành, bạn sẽ thấy thông tin partition trên màn hình giống như bảng liệt kê dưới đây, các mount point sẽ trông giống như dưới đây.

Mount Point	Device	Requested	Actual	Type
/boot	sda1	5MB	5M	Linux Native
/usr	sda5	512MB	512MB	Linux Native
/home	sda6	1146MB	1146MB	Linux Native
/chroot	sda7	256M	256M	Linux Native
/cache	sda8	256M	256M	Linux Native
/var	sda9	256M	256M	Linux Native
<Swap>	sda10	128M	128M	Linux Native
/tmp	sda110	256M	256M	Linux Native
/	sda12	256M	256M	Linux Native

Drive	Geom[C/H/S]	Total (M)	Free (M)	Used (M)	Used (%)
Sda	[3079/64/32]	3079M	1M	3078M	99%

Chú ý: *Chúng ta đang sử dụng một đĩa cứng SCSI bởi vì hai ký tự đầu tiên của thiết bị là "sd"*

Bây giờ chúng ta đang phân chia và chọn mount point cho các thư mục của bạn, chọn "Next" để tiếp tục. Sau khi các partition được tạo, chương trình cài đặt sẽ hỏi bạn chọn partition để định dạng (format). Chọn partition bạn muốn format và chọn vào ô "Check for bad blocks during format) và nhấn "Next". Chương trình sẽ format các partition và làm chúng có hiệu lực khi Linux sử dụng chúng.

Trên màn hình kế tiếp bạn sẽ thấy sự cấu hình LILO, ở đó bạn chọn cài đặt LILO trên boot record:

- Master Boot Record (MBR)

  hoặc

- First Sector of Boot Partition

Trong trường hợp Linux là hệ điều hành (OS) duy nhất trên máy của bạn, bạn nên chọn "Master Boot Record". Kế đó bạn cần cấu hình mạng (network) và giờ (clock) trên máy của bạn. Sau khi hoàn thành việc cấu hình clock (chú ý timezone của chúng ta là: Asia/Saigon), bạn cần đặt một mật khẩu (password) cho root và cấu hình việc kiểm tra tính xác thực (authentication) trên server của bạn.

Khi cấu hình Authentication đừng quên chọn:

- Enable MD5 passwords
- Enable Shadow

Nếu bạn không sử dụng dịch vụ NIS thì bạn không cần chọn "Enable NIS".

Sự chọn lựa những package (gói dữ liệu) riêng lẻ

AllVersion Sau khi các partition đã định hình và được chọn để format, bạn chuẩn bị chọn những gói dữ liệu cho tiến trình cài đặt. Mặc định Linux là một hệ điều hành rất mạnh có khả năng thực thi nhiều dịch vụ hữu ích. Tuy nhiên có khá nhiều dịch vụ không cần thiết thì không nên đưa vào vì có thể tạo ra những nguy hiểm tiềm ẩn cho việc bảo mật.

Một cách lý tưởng là cần cài đặt từng dịch vụ mạng trên các máy phục vụ chuyên biệt. Theo mặc định, nhiều hệ điều hành Linux được cấu hình để cung ứng một bộ các dịch vụ và ứng dụng rộng hơn những yêu cầu cung cấp một dịch vụ mạng riêng biệt, do vậy bạn cần cấu hình server để loại bỏ những dịch vụ không cần thiết. Chỉ đưa ra những dịch vụ chủ yếu trên một máy chủ riêng biệt. Có thể tăng khả năng bảo mật trong server theo một vài phương pháp như sau:

- Những service khác không thể được sử dụng để tấn công máy chủ và làm hư hại hoặc loại bỏ (remove) những dịch vụ được mong muốn.
- Những người khác nhau có thể quản lý những service khác nhau. Bằng cách cô lập các service, mỗi máy chủ và service có thể có riêng lẻ một người quản trị (administrator), bạn sẽ giảm đến mức tối thiểu khả năng xung đột giữa các quản trị viên.
- Máy chủ có thể được cấu hình cho phù hợp hơn với yêu cầu của từng service riêng biệt. Những service khác nhau có thể yêu cầu sự cấu hình phần cứng và phần mềm khác nhau và những cấu hình đó có thể dẫn đến những tổn hại không cần thiết hoặc sự giới hạn service.
- Bằng cách giảm bớt những service, số tập tin log (logfile) và các mục ghi cũng được giảm, vì thế việc xoá bỏ những thông tin không cần thiết trở nên dễ dàng hơn.

Một tiến trình cài đặt chính xác của Linux server chính là bước đầu tiên cho việc ổn định, bảo vệ hệ thống của bạn. Trước hết bạn phải chọn những thành phần (component) hệ thống nào bạn muốn cài đặt. Chọn những component và sau đó bạn có thể tiếp tục chọn và không chọn mỗi gói dữ liệu riêng lẻ của mỗi thành phần bằng cách chọn option (Select individual packages) trên màn hình setup Red Hat.

Khi cấu hình một Linux server chúng ta không cần thiết phải cài đặt một chương trình giao tiếp đồ hoạ (Xfree86) trên máy tính. Việc giảm bớt giao tiếp đồ hoạ (graphical interface) có ý nghĩa lớn trong việc tăng các process, tăng khả năng xử lý của CPU, bộ nhớ, giảm sự nguy hiểm trong bảo mật và giảm một vài bất tiện khác. Giao tiếp đồ họa (Graphical interface) thường chỉ được sử dụng trên các trạm làm việc (workstation).

Chọn những gói dữ liệu dưới đây cho tiến trình cài đặt của bạn:

- Networked Workstation
- Network Management Workstation
- Utilities

Sau khi chọn những thành phần bạn muốn cài đặt bạn vẫn có thể chọn và không chọn các gói dữ liệu.

Chú ý: *Việc chọn tùy chọn (Select individual package) rất quan trọng trước khi tiếp tục khả năng chọn và không chọn các gói dữ liệu.*

Lựa chọn các gói dữ liệu riêng lẻ (Individual package selection)

Trong phần chỉ dẫn cài đặt dưới tôi đưa ra những nhóm gói dữ liệu đã có trong Linux, chọn một nhóm gói dữ liệu nào đó để xem xét.

Ngoài mục đích hướng dẫn cài đặt, trong chương này tôi cũng có ý đưa vài vấn đề bảo mật và tối ưu hoá Linux vào trong tiến trình cài đặt. Những thành phần được liệt kê dưới đây cần được loại bỏ từ cửa sổ chọn gói dữ liệu do vấn đề bảo mật, tối ưu hoá cũng như một vài nguyên nhân khác sẽ được diễn giải dưới đây.

Version 6.1

Applications/File:	git
Applications/Internet:	finger, ftp, fwhois, ncftp, rsh, rsync, talk, telnet
Applications/Publishing:	ghostscript, ghostscript-fonts, mpage, rhs-printfilters
Applications/System:	arpwatch, bind-utils, knfsd-clients, procinfo, rdate, rdist, screen, ucd-snmp-utils
Documentation:	indexhtml
System Environment/Base:	chkfontpath, yp-tools
System Environment/Daemons: rwho, tftp, ucd-snmp, ypbind	XFree86-xfs, lpr, pidentd, portmap, routed, rusers,
System Environment/Libraries:	XFree86-libs, libpng
User Interface/X:	XFree86-75dpi-fonts, urw-fonts

Version 6.2

Applications/File:	git
Applications/Internet:	finger, ftp, fwhois, ncftp, rsh, rsync, talk, telnet
Applications/Publishing: pnm2ppa, rhs-printfilters	ghostscript, ghostscript-fonts, groff-perl, mpage,
Applications/System:	arpwatch, bind-utils, rdate, rdist, screen, ucd-snmp-utils
Documentation:	indexhtml
System Environment/Base:	chkfontpath, yp-tools
System Environment/Daemons:	XFree86-xfs, finger-server, lpr, nfs-utils, pidentd, portmap, rsh-server, rusers, rusers-server, rwall-server, rwho, talk-server, telnet-server, tftp-server, ucd-snmp, ypbind, ypserv
System Environment/Libraries:	XFree86-libs, libpng
User Interface/X:	urw-fonts

Trước khi chúng tôi diễn giải về những chương trình chúng tôi không muốn cài đặt, một ai đó có thể hỏi tôi tại sao không cài đặt finger, ftp, fwhois và telnet trên server. Trước hết chúng ta biết rằng bản thân của những chương trình đó thì không an toàn tức là bản thân chúng không có khả năng tự bảo vệ, giả sử rằng có một tin tặc nào đó truy cập đến server linux của bạn và hắn có thể sử dụng những chương trình như finger, ftp, fwhois, telnet để truy cập đến những máy khác trong mạng nội bộ của bạn. Nếu những chương trình này không được cài đặt trên máy Linux server của bạn thì những tin tặc buộc phải sử dụng những chương trình này từ bên ngoài hoặc thứ cài đặt chúng trên server của bạn. Trong những trường hợp như vậy bạn có thể tìm ra dấu vết chúng nhờ những chương trình giống như Tripwire. Mặc dù trên thực tế với Red Hat Linux 6.2 thì những dịch vụ mạng giống như telnet, finger, talk, rsh, rusers, rwall và tftp đã cài đặt thành từng gói riêng biệt cho máy chủ và các máy client, nhưng chúng tôi luôn đề nghị không cài đặt chúng do vấn đề bảo mật tối đa.

Các gói chương trình không nên cài đặt do các nguyên nhân bảo mật

Dưới đây là danh sách của những chương trình và một sự diễn giải ngắn về việc sử dụng của chúng. Chúng ta phải tháo bỏ (uninstall) chúng để cho việc bảo mật được tốt hơn. Để biết thêm nhiều thông tin và sự diễn giải khả năng và cách dùng chúng ta hãy xem tài liệu hướng dẫn hoặc cài đặt chúng và gõ lệnh rpm –qi packagename để xem nhiều thông tin về chúng và sau đó tháo gỡ chúng.

Applications/File:

- All Version Gói dữ liệu GIT cung cấp một trình duyệt tập tin hệ thống có thể mở rộng, một trình xem (viewer) tập tin ở dạng ASCII/hexadecimal, một tiến trình viewer/killer và các tiện ích liên quan khác. Việc này thì không cần thiết cho server.

Applications/Internet:

- All Version Gói dữ liệu finger là một tiện ích client, nó cho phép người sử dụng xem thông tin về những users hệ thống (nguy hiểm trong bảo mật).
- All Version Gói dữ liệu ftp cung cấp dòng lệnh UNIX chuẩn FTP client (nguy hiểm trong bảo mật).
- All Version Chương trình fwhois client cho phép truy vấn những database nào (nguy hiểm trong bảo mật).
- All Version Gói dữ liệu Ncftp là một sự cải tiến của ftp client (nguy hiểm trong bảo mật, không cần thiết).
- All Version Gói dữ liệu rsh cung cấp những chương trình client cho phép những user chạy những lệnh trên các máy từ xa login vào tới những máy khác và sao chép những tập tin giữa các máy (rsh, rlogin, rcp) (nguy hiểm trong bảo mật).
- All Version Gói dữ liệu ntalk cung cấp cho client và những chương trình daemon cho internet talk protocol cho phép bạn chat với một người sử dụng khác trên những hệ thống UNIX khác (nguy hiểm trong bảo mật).
- All Version Telnet là một protocol phổ biến cho việc login vào những hệ thống từ xa trên network nhưng nó thì không có an toàn (truyền tải mật khẩu không được mã hoá) (nguy hiểm trong bảo mật).
- All Version Gói dữ liệu GhostScript là một bộ của những phần mềm cung cấp một trình phiên dịch PostScript(TM) và một trình phiên dịch cho những tập tin PDF (Portable Document Format) (không cần thiết).
- All Version Trình phiên dịch GhostScript sử dụng gói dữ liệu Ghostscript-fonts trong suốt tiến trình dịch (không cần thiết).
- Version 6.2 Gói dữ liệu groff-perl là một bộ của những lệnh về in được sử dụng trong môi trường in ấn (không cần thiết, vì không có máy in cài đặt trên server).
- All Version Gói dữ liệu mpage làm giảm kích thước văn bản của các tập tin plain text (thuần văn bản) hay tập tin Postscript, và có thể in chúng trên máy in Postsript thành nhiều trang trên một mặt giấy (không cần thiết, vì không có máy in cài đặt trên server).
- All Version Gói dữ liệu pnm2ppa là một driver màu cho những máy in PPA (không cần thiết, vì không có printer cài đặt trên server).
- All Version Gói dữ liệu rhs-printfilter chứa đựng một thiết lập của những bộ lọc printer, chủ yếu được sử dụng với Red Hat printtool (không cần thiết, vì không có printer cài đặt trên server).

Applications/System

- All Version Gói dữ liệu arpwatch bao gồm những tiện ích giám sát lưu thông network Ethenet or FDDI và xây dựng database của cặp địa chỉ Ethernet/IP (không cần thiết).
- All Version Gói dữ liệu bind-utils chứa đựng một bộ của những tiện ích tìm những thông tin về những host trên internet (chúng tôi sẽ viết về nó sau trong cuốn sách này).
- Version 6.1 Gói dữ liệu knfsd-clients chứa đựng những chương trình truy vấn mount daemon trên một máy ở xa cho thông tin về NFS server trên host đó (nguy hiểm trong bảo mật và dịch vụ NFS thì không được cài đặt trên server này).
- Version 6.1 Gói dữ liệu procinfo thu được thông tin về hệ thống của bạn từ kernel khi nó đang chạy (Không cần thiết vì sẵn có những phương pháp khác).
- All Version Gói dữ liệu rdate có thể lấy ngày giờ từ một máy khác trong mạng của bạn (nguy hiểm trong bảo mật).
- All Version Gói dữ liệu rdisk là một chương trình đảm bảo sự giống hệt nhau trong copy tập tin trên nhiều host (nguy hiểm trong bảo mật).
- All Version Gói dữ liệu screen là một tiện ích hữu ích cho những user telnet vào trong một máy nhưng muốn login nhiều lần (không cần thiết, nguy hiểm trong bảo mật).

Documentation:

- All Version Gói dữ liệu indexhtml chứa đựng trang HTML và những hình ảnh cho một trang welcome và được trình bày bởi Web browser (không cần thiết, chúng ta không sử dụng giao tiếp đồ hoạ).

System Environment/Base:

- All Version Gói dữ liệu chkfontpath là một chương trình đơn giản cho việc thêm, loại bỏ và liệt kê những thư mục có trong đường dẫn của X-font server (không cần thiết).
- All Version Network Information Service (NIS) là một hệ thống cung cấp và tập trung thông tin mạng (tên login, mật khẩu, home directory và nhóm thông tin) cho tất cả các máy trên mạng (nguy hiểm trong bảo mật, chúng ta không sử dụng chúng trên server).

System Environment/Daemons:

- All Version Gói dữ liệu XFree86-xfs là một font server cho Xfree86 mà có thể cung cấp font cho các X server từ xa (không cần thiết, chúng ta không sử dụng giao tiếp đồ hoạ).

- Version 6.2 Gói dữ liệu finger-server chứa đựng finger deamon chạy từ tập tin /etc/inetd.conf và cho phép người sử dụng xem thông tin về những user hệ thống trên server (nguy hiểm trong bảo mật).
- All Version Gói dữ liệu lpr cung cấp tiện ích hệ thống cơ bản cho việc quản lý những dịch vụ in ấn (không cần thiết vì printer không được cài đặt trên server).
- Version 6.2 Gói dữ liệu nfs-utils cung cấp những công cụ và daemon cho kernel NFS server. Gói dữ liệu này phải được cài đặt nếu bạn muốn cung cấp dịch vụ NFS trên server của bạn (nguy hiểm trong bảo mật và NFS service không nên cài đặt trên server này).
- All Version Gói dữ liệu pidentd chứa đựng identd, tiện ích này tìm kiếm những kết nối TCP/IP đặc biệt và trả về hoặc tên người sử dụng hoặc thông tin khác về tiến trình là sở hữu của sự kết nối (không cần thiết vì nhiều máy không có nó và nhiều người thường tắt dịch vụ này).
- All Version Gói dữ liệu portmapper quản lý những kết nối RPC mà nó được sử dụng bởi những protocol giống như NFS và NIS (không cần thiết, nguy hiểm trong bảo mật, những dịch vụ NIS/NFS không nên được cài đặt trên server này).
- Version 6.2 Gói dữ liệu rsh-server cung cấp cho server cần cho rsh, rlogin, rcp, tiện ích này cho phép những user chạy những lệnh truy cập trên những máy từ xa (nguy hiểm trong bảo mật).
- All Version Gói dữ liệu rusers cho phép các user tìm kiếm ai đã đăng nhập vào những máy khác nhau trên mạng nội bộ (nguy hiểm trong bảo mật)
- Version 6.2 Gói dữ liệu rusers-server giúp cho server trả lời với các yêu cầu của user và cho phép họ xem ai đã đăng nhập vào trong hệ thống của bạn (nguy hiểm trong bảo mật).
- Version 6.2 Gói dữ liệu rwall-server chứa đựng daemon cho phép nhận những messages của những user trong các host từ xa (nguy hiểm trong bảo mật).
- All Version Gói dữ liệu rwho cho biết ai đã đăng nhập trong tất cả các host trên local network đang chạy rwho daemon (nguy hiểm trong bảo mật).
- Version 6.2 Gói dữ liệu talk-server cung cấp chương trình daemon cho phép bạn chat qua terminal với những user khác trên những hệ thống unix từ xa. (nguy hiểm trong bảo mật).
- Version 6.1 Gói dữ liệu tftp hoặc Trivial File Transfer Protocol (TFTP) cho phép người sử dụng truyền tải tập tin từ một máy ở xa. Nó chỉ thường được sử dụng cho việc khởi động các workstation không đĩa (nguy hiểm trong bảo mật, không cần thiết).
- Version 6.2 Gói dữ liệu telnet-server cung cấp daemon cho phép người sử dụng login từ xa tới server của bạn (nguy hiểm trong bảo mật, thay thế bằng SSH).

- Version 6.2 Gói dữ liệu tftp-server cung cấp cho server dịch vụ TFTP cho phép người sử dụng truyền tải tập tin tới server từ một máy ở xa (nguy hiểm trong bảo mật, không cần thiết).
- All Version Gói dữ liệu ucd-snmp hoặc SNMP (Simple Network Information Management Protocol) là một giao thức được sử dụng cho quản lý mạng (không cần thiết, nguy hiểm trong bảo mật).
- Version 6.2 Gói dữ liệu ypserv là NIS (Network Information Service) server, gói dữ liệu này cung cấp thông tin mạng (NIS) cho tất cả các máy trong mạng (nguy hiểm trong bảo mật, chúng ta không sử dụng nó trên server của chúng ta).

System Environment/Libraries:

- All Version Gói dữ liệu Xfree86-libs chứa đựng những thư viện dùng chung mà hầu hết những chương trình X Window cần sử dụng để chạy (không cần thiết, chúng ta không sử dụng giao tiếp đồ hoạ).
- All Version Gói dữ liệu libpng chứa đựng một thư viện của những chức năng cho việc tạo và chế tác những tập tin hình ảnh dạng PNG. PNG là một dạng ảnh điểm (bit-mapped graphic format) tương tự như dạng GIF (không cần thiết).

User Interface/X

- Version 6.1 Gói dữ liệu Xfree86-75dpi-fonts chứa đựng những font 75dpi (font chuẩn) được sử dụng trên hầu hết những hệ thống X window (không cần thiết, chúng ta không sử dụng giao tiếp đồ hoạ).
- All Version Gói dữ liệu urw-font chứa đựng những font tự do của chuẩn 35, loại font PostScript (không cần thiết, chúng ta không dùng đồ hoạ).

Sau khi hoàn tất việc lựa chọn những gói dữ liệu thích hợp sẽ được cài đặt trên server chương trình cài đặt sẽ format mọi partition bạn đã chọn trong hệ thống của bạn. Việc này sẽ mất vài phút đến vài chục phút tùy thuộc vào tốc độ của máy tính bạn. Một khi tất cả các partition đã được format, chương trình cài đặt đã bắt đầu cài đặt những gói dữ liệu.

Xin nhắc lại mục đích của cuốn sách này là giới thiệu cho bạn các vấn đề liên quan đến bảo mật và tối ưu hoá một server Linux nên nhiều gói dữ liệu bị loại bỏ, tuy nhiên nếu bạn cài Linux với mục đích để nghiên cứu và sử dụng thì bạn nên chọn hết tất cả các gói dữ liệu để cài đặt.

Làm thế nào sử dụng những lệnh RPM

Phần này giới thiệu tổng quan về lệnh RPM, cách sử dụng lệnh RPM cho sự cài đặt, loại bỏ, nâng cấp, truy vấn và kiểm tra những gói dữ liệu trên hệ thống Linux của bạn. Bây giờ bạn phải làm quen với những lệnh RPM vì chúng ta sẽ thường sử dụng chúng trong cuốn sách này.

- Cài đặt (install) một gói dữ liệu RPM, sử dụng lệnh:

```
[root@deep /]# rpm -ivh foo-1.0-2.i386.rpm
```

Dòng lệnh trên có ý nghĩa cài đặt một gói dữ liệu rpm có tên là foo-1.0-2.i386.rpm với các thành phần như sau:

Tên gói dữ liệu: foo

Version: 1.0

Release: 2

Kiến trúc i386

- Loại bỏ (uninstall) một gói dữ liệu RPM, sử dụng lệnh:

```
[root@deep /]# rpm -e foo
```

Chú ý: *Chúng ta sử dụng tên gói dữ liệu "foo" không phải tên nguyên thủy của gói dữ liệu "foo-1.0-2.i386.rpm".*

- Nâng cấp (upgrade) một gói dữ liệu RPM, sử dụng lệnh:

```
[root@deep /]# rpm -Uvh foo-1.0-2.i386.rpm
```

Với lệnh này RPM gỡ bỏ (uninstall) một cách tự động version cũ và cài đặt một version mới. Chúng ta có thể sử dụng "rpm –Uvh" để cài đặt gói dữ liệu ngay cả khi không có những version trước của gói dữ liệu cài đặt.

- Truy vấn (query) một gói dữ liệu RPM:

```
[root@deep /]# rpm -q foo
```

Lệnh này sẽ in tên, version, và số release của gói dữ liệu foo đã được cài đặt. Dùng lệnh này có thể kiểm tra một gói dữ liệu có được cài đặt trên hệ thống hay không.

- Trình bày thông tin gói dữ liệu, sử dụng lệnh:

```
[root@deep /]# rpm -qi foo
```

Lệnh này trình bày thông tin gói dữ liệu bao gồm tên, version và sự diễn giải của chương trình được cài đặt. Dùng lệnh này để biết thông tin về gói dữ liệu được cài đặt.

- Liệt kê những tập tin trong gói dữ liệu sử dụng lệnh:

```
[root@deep /]# rpm -ql foo
```

Lệnh này sẽ liệt kê danh sách những tập tin trong một gói dữ liệu RPM đã được cài đặt, nó chỉ làm việc khi gói dữ liệu đã được cài sẵn trên hệ thống.

- Kiểm tra một RPM signature gói dữ liệu sử dụng lệnh:

```
[root@deep /]# rpm --checksig foo
```

Lệnh này kiểm tra chữ ký PGP của gói dữ liệu được chỉ định để đảm bảo tính toàn vẹn và nguyên gốc của nó. Luôn sử dụng lệnh này đầu tiên trước khi cài đặt gói dữ liệu RPM mới trên hệ thống của bạn. Để dùng lệnh này trước tiên phần mềm GnuPG hoặc Pgp phải được cài sẵn trên máy của bạn.

Khởi động và dừng những dịch vụ daemon (starting and stopping daemon services)

Chương trình init của Linux (cũng được hiểu như khởi tạo việc điều khiển tiến trình) phụ trách việc khởi động tất cả các tiến trình (process) bình thường hoặc được

ủy quyền cần chạy lúc khởi động hệ thống. Những tiến trình này có thể bao gồm APACHE, NETWORK daemon và bất kỳ những tiến trình khác yêu cầu phải chạy khi server bạn khởi động. Mỗi process này có một tập tin script trong thư mục "/etc/rc.d/init.d". Bạn có thể thi hành những script này với dòng lệnh như sau:

Ví dụ:

- Khởi động httpd Web server bằng tay dưới Linux:

```
[root@deep /]# /etc/rc.d/init.d/httpd start
Starting httpd: [ OK ]
```

- Dừng httpd Web server bằng tay dưới Linux:

```
[root@deep /]# /etc/rc.d/init.d/httpd stop
Shutting down http: [ OK ]
```

- Khởi động lại httpd bằng tay dưới Linux:

```
[root@deep /]# /etc/rc.d/init.d/httpd restart
Shutting down http: [ OK ]
Starting httpd: [ OK ]
```

Tương tự bạn có thể kiểm tra việc start, stop, restart với những service khác trong thư mục /etc/rc.d/init.d.

Các phần mềm cần phải loại bỏ sau khi tiến trình cài đặt của server hoàn thành

Mặc định có một số gói dữ liệu mà hệ thống Red Hat Linux không cho phép bạn chọn để tháo gỡ trong suốt tiến trình setup. Vì nguyên nhân này bạn phải loại bỏ (uninstall) chúng sau khi tiến trình cài đặt hoàn thành.

Version 6.1

Pump	kernel-pcmcia-cs	setserial	Red Hat-release
mt-st	linuxconf	kudzu	gd
eject	getty_ps	raidtools	pciutils
mailcap	setconsole	gnupg	rmt
apmd	isapnptools	Red Hat-logos	

Version 6.2

Pump	kernel-pcmcia-cs	kudzu	gd
mt-st	linuxconf	raidtools	pciutils
eject	getty_ps	gnupg	rmt
mailcap	isapnptools	Red Hat-logos	
apmd	setserial	Red Hat-release	

Sử dụng lệnh RPM như dưới đây để tháo gỡ chúng.

- Lệnh dùng để tháo gỡ phần mềm là:

```
[root@deep /]# rpm -e <softwarenames>
```

Ở đây <softwarename> là tên của phần mềm bạn muốn tháo gỡ, ví dụ như (foo)

Một số chương trình daemon như apmd, kudzu, sendmail đều chạy ở lúc khởi động máy, tốt nhất bạn nên dừng (stop) chúng trước khi tháo gỡ ra khỏi hệ thống của bạn.

- Dừng các process với những lệnh:

```
[root@deep /]# /etc/rc.d/init.d/apmd stop
[root@deep /]# /etc/rc.d/init.d/sendmail stop
[root@deep /]# /etc/rc.d/init.d/kudzu stop
```

Bây giờ bạn có thể tháo gỡ chúng cùng với các gói dữ liệu khác một cách an toàn với lệnh sau:

Bước 1:

Version 6.1

Xóa bỏ những gói dữ liệu được chỉ định cho Red Hat Linux6.1:

```
[root@deep /]# rpm -e --nodeps pump mt-st eject mailcap apmd kernel-
pcmcia-cs linuxconf getty_ps setconsole isapnptools setserial kudzu raidtools
gnupg Red Hat-logos Red Hat-release gd pciutils rmt.
```

Version 6.2

Xóa bỏ những gói dữ liệu được chỉ định cho Red Hat Linux6.2:

```
[root@deep /]# rpm -e --nodeps pump mt-st eject mailcap apmd kernel-
pcmcia-cs linuxconf getty_ps isapnptools setserial kudzu raidtools gnupg Red
Hat-logos Red Hat-release gd pciutils rmt
```

Bước 2

All Version

Xóa bỏ tập tin linux.conf-installed bằng tay:

```
[root@deep /]# rm -f /etc/conf.linuxconf-installed
```

Chú ý: *Đây là tập tin cấu hình có quan hệ với phần mềm linuxconf và phải được remove bằng tay.*

Chương trình hdparm cần cho các IDE hard disk nhưng không cần cho SCSI hard disk. Nếu bạn có một IDE hard disk bạn phải giữ lại chương trình này (hdparm), nhưng nếu bạn không có IDE hard disk bạn có thể xóa bỏ nó khỏi hệ thống.

- Xóa bỏ chương trình hdparm khỏi hệ thống của bạn với lệnh:

```
[root@deep /]# rpm -e hdparm
```

Những chương trình như kbdconfig, mouseconfig, timeconfig, authconfig, ntsysv và setuptool theo thứ tự thiết lập loại keyboard, mouse, timezone, NIS và shadow password, chúng ít khi thay đổi sau khi cài đặt, vì thế bạn có thể tháo gỡ chúng, nếu trong tương lai bạn cần thay đổi keyboard, mouse, timezone ...bạn có thể cài đặt những chương trình này từ các gói dữ liệu RPM trên CD-ROM Rehat Linux.

- Xóa bỏ những chương trình trên từ hệ thống với lệnh:

```
[root@deep /]# rpm -e kbdconfig mouseconfig timeconfig authconfig ntsysv
setuptool
```

Sự diễn giải những chương trình phải được tháo gỡ sau tiến trình cài đặt của server. tháo gỡ

Dưới đây là những chương trình và sự diễn giải ngắn về công dụng của chúng. Chúng ta phải gỡ bỏ chúng cho việc bảo mật được tốt hơn cũng như có thêm khoảng đĩa trống trên server. Để biết thêm thông tin về khả năng và cách dùng, bạn có thể đọc tài liệu hướng dẫn về chúng trong thư mục "/usr/doc".

- All Version Gói dữ liệu Pump DHCP cho phép các client trên mạng có thể nhận IP address từ các DHCP server trong mạng (không cần thiết).
- All Version Gói dữ liệu mt(magnetic tape drives) và st (SCSI tape devices) : chương trình quản lý những đĩa từ và những thiết bị băng từ SCSI, có thể điều khiển việc xoá, tua lại, bỏ qua các file ...(không cần thiết nếu chúng ta không có tape backup trên server).
- All Version Gói dữ liệu eject chứa đựng một chương trình mà cho phép người sử dụng dùng phần mềm điều khiển những thiết bị như CD-ROM, đĩa mềm, đĩa Iomega (chỉ cần thiết nếu có tape backup trên server).
- All Version Metamail là một chương trình định rõ làm thế nào nó sẽ trình bày những tài liệu ở dạng phi - văn bản hay multimedia.
- All Version Gói dữ liệu apmd và những tiện ích về sự quản lý nguồn (Power Management) có thể theo dõi năng lượng pin của các máy notebook và cảnh báo cho các user khi nào nguồn năng lượng này cạn kiệt (không cần thiết cho server).
- All Version Gói dữ liệu kernel-pcmcia-cs cho các máy laptop và một vài loại non-laptop hỗ trợ sự mở rộng cho card PCMCIA (không cần thiết cho server).
- All Version Gói dữ liệu linuxconfig là một công cụ cấu hình hệ thống (không cần thiết).
- All Version Gói dữ liệu getty_ps chứa đựng những chương trình được dùng để chấp nhận những login trên console (bàn giao tiếp người-máy) hoặc một terminal trên hệ thống của bạn.
- Version 6.1 Gói dữ liệu setconsole là một tiện ích hệ thống cơ bản cho việc thiết lập các tập tin /etc/inittab, /dev/systty và /dev/console để điều khiển console mới (không cần thiết).
- All Version Gói dữ liệu isapnptools chứa đựng những tiện ích cho việc cấu hình các card/board ISA Plug-and-Play(PnP) (không cần thiết).
- All Version Gói dữ liệu setserial là một tiện ích hệ thống cơ bản cho việc trình bày hoặc thiết lập thông tin cổng nối tiếp (COM port) (không cần thiết).

- All Version Gói dữ liệu kudzu là một công cụ phát hiện phần cứng chạy ở lúc boot máy nhằm chỉ rõ hardware nào được thêm vào hoặc gỡ bỏ khỏi hệ thống (không cần thiết).
- All Version Gói dữ liệu raidtools chứa đựng những công cụ bạn cần cài đặt và duy trì thiết bị RAID trên Linux. (tùy thuộc nếu bạn sử dụng RAID hay không).
- All Version Gói dữ liệu GnuPG là một công cụ cho việc bảo vệ sự trao đổi thông tin và lưu trữ dữ liệu. Nó là một sự thay thế cho phần mềm PGP, nó cũng được sử dụng để mã hoá dữ liệu và tạo chữ ký điện tử (digital signature) (chúng ta sẽ viết về nó sau đó trong sách này ở các chương sau).
- All Version Gói dữ liệu Red Hat-logos chứa đựng những tập tin của logo Red Hat và logo RPM (không cần thiết trên server).
- All Version Gói dữ liệu Red Hat-release chứa đựng những tập tin thông báo những thay đổi trong các bản phát hành của Red Hat Linux (không cần thiết).
- All Version Gói dữ liệu gd cho phép mã hoá những hình ảnh một cách nhanh chóng và ghi ra dạng tập tin .gif (không cần thiết).
- All Version Gói dữ liệu rmt cung cấp tiện ích truy cập mạng từ xa cho việc backup (nguy hiểm trong bảo mật).

Các phần mềm phải được cài đặt sau sự cài đặt của server

Để có thể biên dịch (compile) những chương trình trên server của bạn, bạn phải cài đặt những gói dữ liệu RPM sau. Phần cài đặt này rất quan trọng và yêu cầu bạn cài đặt tất cả những gói dữ liệu liên hệ được kể ra ở dưới đây. Những gói dữ liệu này có trong CD-ROM Red Hat Linux (phần 1) dưới thư mục RedHat/RPMS.

Bước 1:

Đầu tiên chúng ta mount ổ đĩa CD-ROM và chuyển tới thư mục RPMS trên CD-ROM.

- Mount CD-ROM drive và chuyển tới thư mục RPMS sử dụng những lệnh sau:

```
[root@deep /]# mount /dev/cdrom /mnt/cdrom/
[root@deep /]# cd /mnt/cdrom/Red Hat/RPMS/
```

Dưới đây là những gói dữ liệu mà bạn cần biên dịch và cài đặt trên hệ thống Linux. Nên nhớ rằng đây là số gói dữ liệu tối thiểu cho phép bạn biên dịch hầu hết những tarball sẵn có cho Linux. Những trình biên dịch khác tồn tại trên CD-ROM Red Hat do vậy cần kiểm tra tập tin README đi cùng với chương trình tarballs bạn muốn cài đặt, nếu bạn nhận một thông báo lỗi trong quá trình biên dịch của phần mềm cụ thể.

Version 6.1	Version 6.2
autoconf-2.13-5.noarch.rpm	autoconf-2.13-5.noarch.rpm

m4-1.4-12.i386.rpm	m4-1.4-12.i386.rpm
automake-1.4-5.noarch.rpm	automake-1.4-6.noarch.rpm
dev86-0.14.9-1.i386.rpm	dev86-0.15.0-2.i386.rpm
bison-1.28-1.i386.rpm	bison-1.28-2.i386.rpm
byacc-1.9-11.i386.rpm	byacc-1.9-12.i386.rpm
cdecl-2.5-9.i386.rpm	cdecl-2.5-10.i386.rpm
cpp-1.1.2-24.i386.rpm	cpp-1.1.2-30.i386.rpm
cproto-4.6-2.i386.rpm	cproto-4.6-3.i386.rpm
ctags-3.2-1.i386.rpm	ctags-3.4-1.i386.rpm
egcs-1.1.2-24.i386.rpm	egcs-1.1.2-30.i386.rpm
ElectricFence-2.1-1.i386.rpm	ElectricFence-2.1-3.i386.rpm
flex-2.5.4a-7.i386.rpm	flex-2.5.4a-9.i386.rpm
gdb-4.18-4.i386.rpm	gdb-4.18-11.i386.rpm
kernel-headers-2.2.12-20.i386.rpm	kernel-headers-2.2.14-5.0.i386.rpm
glibc-devel-2.1.2-11.i386.rpm	glibc-devel-2.1.3-15.i386.rpm
make-3.77-6.i386.rpm	make-3.78.1-4.i386.rpm
patch-2.5-9.i386.rpm	patch-2.5-10.i386.rpm

Chú ý: *Cách tốt nhất là đồng thời cài đặt các phần mềm mô tả ở trên, nếu bạn không muốn nhận những thông báo lỗi do sự phụ thuộc lẫn nhau giữa các phần mềm trong lúc cài đặt bằng RPM.*

Bước 2

Version 6.1

Cài đặt những phần mềm cần thiết ở trên với một lệnh RPM.

- Lệnh RPM để cài đặt các phần mềm với nhau là:

```
[root@deep RPMS]# rpm -Uvh autoconf-2.13-5.noarch.rpm m4-1.4-12.i386.rpm automake-1.4-5.noarch.rpm dev86-0.14.9-1.i386.rpm bison-1.28-1.i386.rpm byacc-1.9-11.i386.rpm cdecl-2.5-9.i386.rpm cpp-1.1.2-24.i386.rpm cproto-4.6-2.i386.rpm ctags-3.2-1.i386.rpm egcs-1.1.2-24.i386.rpm ElectricFence-2.1-1.i386.rpm flex-2.5.4a-7.i386.rpm gdb-4.18-4.i386.rpm kernel-headers-2.2.12-20.i386.rpm glibc-devel-2.1.2-11.i386.rpm make-3.77-6.i386.rpm patch-2.5-9.i386.rpm
autoconf ########################################################
m4       ########################################################
automake ########################################################
dev86    ########################################################
bison    ########################################################
byacc    ########################################################
cdecl    ########################################################
cpp      ########################################################
cproto   ########################################################
ctags    ########################################################
```

```
egcs            ##########################################
ElectricFence   ##########################################
flex            ##########################################
gdb             ##########################################
kernel-headers  ##########################################
glibc-devel     ##########################################
make            ##########################################
patch           ##########################################
```

Version 6.2

Cài đặt tất cả các phần mềm cần thiết ở trên với một lệnh RPM.

- Lệnh RPM để cài đặt tất cả cácphần mềm với nhau là:

```
[root@deep RPMS]# rpm -Uvh autoconf-2.13-5.noarch.rpm m4-1.4-12.i386.rpm automake-1.4-6.noarch.rpm dev86-0.15.0-2.i386.rpm bison-1.28-2.i386.rpm byacc-1.9-12.i386.rpm cdecl-2.5-10.i386.rpm cpp-1.1.2-30.i386.rpm cproto-4.6-3.i386.rpm ctags-3.4-1.i386.rpm egcs-1.1.2-30.i386.rpm ElectricFence-2.1-3.i386.rpm flex-2.5.4a-9.i386.rpm gdb-4.18-11.i386.rpm kernel-headers-2.2.14-5.0.i386.rpm glibc-devel-2.1.3-15.i386.rpm make-3.78.1-4.i386.rpm patch-2.5-10.i386.rpm
autoconf        ##########################################
m4              ##########################################
automake        ##########################################
dev86           ##########################################
bison           ##########################################
byacc           ##########################################
cdecl           ##########################################
cpp             ##########################################
cproto          ##########################################
ctags           ##########################################
egcs            ##########################################
ElectricFence   ##########################################
flex            ##########################################
gdb             ##########################################
kernel-headers  ##########################################
glibc-devel     ##########################################
make            ##########################################
patch           ##########################################
```

Bước 3

Bạn phải thoát ra khỏi console và login trở lại để tất cả các thay đổi có hiệu lực.

- Thoát khỏi console với lệnh:

```
[root@deep /]# exit
```

Sau khi đã cài đặt và biên dịch tất cả các chương trình bạn cần trên server của bạn sẽ là một ý hay nếu bạn xóa bỏ các tập tin object được tạo ra do biên dịch, các trình biên dịch,..., những tập tin mà bạn không còn cần nữa trong hệ thống của

bạn. Một trong những lý do là nếu một tên tin tặc xâm nhập server của bạn hắn không thể biên dịch hoặc thay đổi những chương trình nhị phân. Hơn nữa việc này sẽ giải phóng nhiều khoảng trống và sẽ giúp đỡ việc cải tiến kiểm tra tính toàn vẹn của những tập tin trên server.

Khi bạn chạy một server bạn sẽ truyền cho nó một công việc đặc biệt để thực hiện. Bạn sẽ không bao giờ đặt tất cả các service bạn muốn cung cấp trên một máy hoặc bạn sẽ làm chậm tốc độ (tài nguyên có sẵn được chia bởi một số tiến trình đang chạy trên server) và làm suy yếu khả năng bảo mật của bạn (với nhiều service chạy trên cùng một máy, nếu một tin tặc xâm nhập vào server này hắn có thể tấn công trực tiếp những gì sẵn có trên đó).

Có nhiều server khác nhau làm những công việc khác nhau sẽ đơn giản hoá sự trông coi, quản lý (bạn biết công việc gì mỗi server sẽ làm, những service nào có hiệu lực, port nào thì được mở cho những client truy cập và port nào thì đóng, bạn cũng sẽ biết những gì bạn cần thấy trong các log file....) và đặt cho bạn sự điều khiển và tính linh hoạt trên mỗi server (server chuyên dành cho mail, web, database, backup ....).

Những chương trình được cài đặt trên server của bạn

Bước 1

Do chúng ta chọn tối ưu hoá việc cài đặt hệ thống Linux của chúng ta, đây là danh sách của tất cả các chương trình được cài đặt mà bạn sẽ có sau khi hoàn tất việc cài đặt Linux. Danh sách này phải so khớp một cách chính xác với nội dung tập tin install.log trong thư mục /tmp. Đừng quên cài đặt tất cả các chương trình được liệt kê trong mục "Các phần mềm phải được cài đặt sau tiến trình cài đặt của server" để có thể biên dịch đúng cách trên server của bạn.

Version 6.1

Installing setup.	Installing gzip.	Installing sendmail.
Installing filesystem.	Installing hdparm.	Installing setconsole.
Installing basesystem.	Installing initscripts.	Installing setserial.
Installing ldconfig.	Installing ipchains.	Installing setuptool.
Installing glibc.	Installing isapnptools.	Installing shapecfg.
Installing shadow-utils.	Installing kbdconfig.	Installing slang.
Installing mktemp.	Installing kernel.	Installing slocate.
Installing termcap.	Installing kernel-pcmcia-cs.	Installing stat.
Installing libtermcap.	Installing kudzu.	Installing sysklogd.
Installing bash.	Installing ld.so.	Installing tar.
Installing MAKEDEV.	Installing less.	Installing tcp_wrappers.
Installing SysVinit.	Installing libc.	Installing tcpdump.
Installing XFree86-Mach64.	Installing libstdc++.	Installing tcsh.
Installing chkconfig.	Installing lilo.	Installing time.
Installing apmd.	Installing pwdb.	Installing timeconfig.

Installing ncurses.
Installing info.
Installing fileutils.
Installing grep.
Installing ash.
Installing at.
Installing authconfig.
Installing bc.
Installing bdflush.
Installing binutils.
Installing bzip2.
Installing sed.
Installing console-tools.
Installing e2fsprogs.
Installing rmt.
Installing cpio.
Installing cracklib.
Installing cracklib-dicts.
Installing crontabs.
Installing textutils.
Installing dev.
Installing diffutils.
Installing dump.
Installing ed.
Installing eject.
Installing etcskel.
Installing file.
Installing findutils.
Installing gawk.
Installing gd.
Installing gdbm.
Installing getty_ps.
Installing glib.
Installing gmp.
Installing gnupg.
Installing gpm.
Installing groff.
Installing pam.
Installing sh-utils.
Installing Red Hat-release.
Installing linuxconf.
Installing logrotate.
Installing losetup.
Installing lsof.
Installing mailcap.
Installing mailx.
Installing man.
Installing mingetty.
Installing mkbootdisk.
Installing mkinitrd.
Installing modutils.
Installing mount.
Installing mouseconfig.
Installing mt-st.
Installing ncompress.
Installing net-tools.
Installing netkit-base.
Installing newt.
Installing ntsysv.
Installing passwd.
Installing pciutils.
Installing perl.
Installing procmail.
Installing procps.
Installing psmisc.
Installing pump.
Installing python.
Installing quota.
Installing raidtools.
Installing readline.
Installing Red Hat-logos.
Installing rootfiles.
Installing rpm.
Installing sash.
Installing timed.
Installing tmpwatch.
Installing traceroute.
Installing utempter.
Installing util-linux.
Installing vim-common.
Installing vim-minimal.
Installing vixie-cron.
Installing which.
Installing zlib.

Version 6.2

Installing setup.
Installing filesystem.
Installing basesystem.
Installing gnupg.
Installing gpm.
Installing groff.
Installing quota.
Installing raidtools.
Installing readline.

Installing ldconfig.
Installing glibc.
Installing shadow-utils.
Installing mktemp.
Installing termcap.
Installing libtermcap.
Installing bash.
Installing MAKEDEV.
Installing SysVinit.
Installing XFree86-Mach64.
Installing anacron.
Installing chkconfig.
Installing apmd.
Installing ncurses.
Installing info.
Installing fileutils.
Installing grep.
Installing ash.
Installing at.
Installing authconfig.
Installing bc.
Installing bdflush.
Installing binutils.
Installing bzip2.
Installing sed.
Installing console-tools.
Installing e2fsprogs.
Installing rmt.
Installing cpio.
Installing cracklib.
Installing cracklib-dicts.
Installing crontabs.
Installing textutils.
Installing dev.
Installing diffutils.
Installing dump.
Installing ed.
Installing eject.
Installing etcskel.
Installing file.
Installing findutils.
Installing gzip.
Installing hdparm.
Installing inetd.
Installing initscripts.
Installing ipchains.
Installing iputils.
Installing isapnptools.
Installing kbdconfig.
Installing kernel.
Installing kernel-pcmcia-cs
Installing kernel-utils.
Installing kudzu.
Installing ld.so.
Installing less.
Installing libc.
Installing libstdc++.
Installing lilo.
Installing pwdb.
Installing pam.
Installing sh-utils.
Installing Red Hat-release.
Installing linuxconf.
Installing logrotate.
Installing losetup.
Installing lsof.
Installing mailcap.
Installing mailx.
Installing man.
Installing mingetty.
Installing mkbootdisk.
Installing mkinitrd.
Installing modutils.
Installing mount.
Installing mouseconfig.
Installing mt-st.
Installing ncompress.
Installing net-tools.
Installing newt.
Installing ntsysv.
Installing passwd.
Installing pciutils.
Installing Red Hat-logos.
Installing rootfiles.
Installing rpm.
Installing sash.
Installing sendmail.
Installing setserial.
Installing setuptool.
Installing shapecfg.
Installing slang.
Installing slocate.
Installing stat.
Installing sysklogd.
Installing tar.
Installing tcp_wrappers.
Installing tcpdump.
Installing tcsh.
Installing time.
Installing timeconfig.
Installing tmpwatch.
Installing traceroute.
Installing utempter.
Installing util-linux.
Installing vim-common.
Installing vim-minimal.
Installing vixie-cron.
Installing which.
Installing zlib.

Installing gawk.	Installing perl.
Installing gd.	Installing popt.
Installing gdbm.	Installing procmail.
Installing getty_ps.	Installing procps.
Installing glib.	Installing psmisc.
Installing gmp.	Installing pump.

Bước 2:

Sau khi chúng ta gỡ bỏ tất cả các phần mềm cần phải tháo gỡ sau tiến trình cài đặt của server (xem phần "các phần mềm phải tháo gỡ sau tiến trình cài đặt của server") và sau khi thêm những gói dữ liệu RPM cần thiết để có thể biên dịch những chương trình trên server của chúng ta (xem phần "Các phần mềm phải được cài đặt sau tiến trình cài đặt của server"). Chúng ta phải kiểm tra lại danh sách của tất cả các chương trình RPM đã được cài đặt với lệnh sau.

- Kiểm tra danh sách tất cả các gói dữ liệu được cài đặt trên server sử dụng lệnh:

```
[root@deep /]# rpm -qa > installed_rpm
```

Tham số "-qa" sẽ truy vấn tất cả các gói dữ liệu RPM được cài đặt trên hệ thống và ký tự đặc biệt ">" sẽ ghi lại tất cả những gì xuất ra trên màn hình vào tập tin installed_rpm.

Version 6.1

Nội dung của tập tin installed_rpm phải giống như dưới đây:

setup-2.0.5-1	findutils-4.1-32	netkit-base-0.10-37
filesystem-1.3.5-1	gawk-3.0.4-1	newt-0.50-13
basesystem-6.0-4	cdecl-2.5-9	passwd-0.63-1
ldconfig-1.9.5-15	gdbm-1.8.0-2	perl-5.00503-6
glibc-2.1.2-11	autoconf-2.13-5	flex-2.5.4a-7
shadow-utils-19990827-2	glib-1.2.5-1	procps-2.0.4-2
mktemp-1.5-1	gmp-2.0.2-10	psmisc-18-3
termcap-9.12.6-15	cpp-1.1.2-24	python-1.5.2-7
libtermcap-2.0.8-18	gpm-1.17.9-3	quota-1.66-8
bash-1.14.7-16	groff-1.11a-9	gdb-4.18-4
MAKEDEV-2.5-2	gzip-1.2.4-14	readline-2.2.1-5
SysVinit-2.77-2	initscripts-4.48-1	glibc-devel-2.1.2-11
chkconfig-1.0.7-2	ipchains-1.3.9-3	rootfiles-5.2-5
ncurses-4.2-25	cproto-4.6-2	rpm-3.0.3-2
info-3.12h-2	ElectricFence-2.1-1	sash-3.3-1
fileutils-4.0-8	kernel-2.2.12-20	make-3.77-6
grep-2.3-2	patch-2.5-9	shapecfg-2.2.12-2
ash-0.2-18	ld.so-1.9.5-11	slang-1.2.2-4

at-3.1.7-11
m4-1.4-12
bdflush-1.5-10
binutils-2.9.1.0.23-6
bzip2-0.9.5c-1
sed-3.02-4
console-tools-19990302-17
e2fsprogs-1.15-3
byacc-1.9-11
cpio-2.4.2-13
cracklib-2.7-5
cracklib-dicts-2.7-5
crontabs-1.7-7
textutils-2.0-2
dev-2.7.10-2
diffutils-2.7-16
dump-0.4b4-11
ed-0.2-12
bison-1.28-1
etcskel-2.0-1
file-3.27-3
less-340-1
libc-5.3.12-31
libstdc++-2.9.0-24
lilo-0.21-10
pwdb-0.60-1
pam-0.68-7
sh-utils-2.0-1
automake-1.4-5
logrotate-3.3-1
losetup-2.9u-4
lsof-4.45-1
mailx-8.1.1-9
man-1.5g-6
mingetty-0.9.4-10
mkbootdisk-1.2.2-1
mkinitrd-2.3-1
modutils-2.1.121-14
mount-2.9u-4
ctags-3.2-1
ncompress-4.2.4-14
net-tools-1.53-1
slocate-2.0-3
stat-1.5-11
sysklogd-1.3.31-12
tar-1.13.11-1
tcp_wrappers-7.6-9
tcpdump-3.4-16
tcsh-6.08.00-6
time-1.7-9
timed-0.10-23
tmpwatch-2.0-1
traceroute-1.4a5-16
utempter-0.5.1-2
util-linux-2.9w-24
vim-common-5.4-2
vim-minimal-5.4-2
vixie-cron-3.0.1-39
which-2.8-1
zlib-1.1.3-5
dev86-0.14.9-1
egcs-1.1.2-24
kernel-headers-2.2.12-20

Version 6.2

Nội dung của tập tin installed_rpm phải giống như dưới đây:

setup-2.1.8-1
filesystem-1.3.5-1
basesystem-6.0-4
ldconfig-1.9.5-16
glibc-2.1.3-15
shadow-utils-19990827-10
mktemp-1.5-2
termcap-10.2.7-9
libtermcap-2.0.8-20
bash-1.14.7-22
MAKEDEV-2.5.2-1
SysVinit-2.78-5
anacron-2.1-6
chkconfig-1.1.2-1
m4-1.4-12
ncurses-5.0-11
info-4.0-5
findutils-4.1-34
gawk-3.0.4-2
patch-2.5-10
gdbm-1.8.0-3
bison-1.28-2
glib-1.2.6-3
gmp-2.0.2-13
autoconf-2.13-5
gpm-1.18.1-7
groff-1.15-8
gzip-1.2.4a-2
inetd-0.16-4
initscripts-5.00-1
ipchains-1.3.9-5
iputils-20000121-2
cpp-1.1.2-30
cproto-4.6-3
flex-2.5.4a-9
ncompress-4.2.4-15
net-tools-1.54-4
newt-0.50.8-2
passwd-0.64.1-1
perl-5.00503-10
popt-1.5-0.48
procmail-3.14-2
procps-2.0.6-5
psmisc-19-2
quota-2.00pre3-2
gdb-4.18-11
readline-2.2.1-6
make-3.78.1-4
rootfiles-5.2-5
rpm-3.0.4-0.48
sash-3.4-2

fileutils-4.0-21
grep-2.4-3
ash-0.2-20
at-3.1.7-14
byacc-1.9-12
bc-1.05a-5
bdflush-1.5-11
binutils-2.9.5.0.22-6
bzip2-0.9.5d-2
sed-3.02-6
console-tools-19990829-10
e2fsprogs-1.18-5
cpio-2.4.2-16
cracklib-2.7-5
cracklib-dicts-2.7-5
crontabs-1.7-7
textutils-2.0a-2
dev-2.7.18-3
diffutils-2.7-17
dump-0.4b15-1
ed-0.2-13
cdecl-2.5-10
etcskel-2.3-1
file-3.28-2
kernel-2.2.14-5.0
ctags-3.4-1
kernel-utils-2.2.14-5.0
ElectricFence-2.1-3
ld.so-1.9.5-13
less-346-2
libc-5.3.12-31
libstdc++-2.9.0-30
lilo-0.21-15
pwdb-0.61-0
pam-0.72-6
sh-utils-2.0-5
automake-1.4-6
logrotate-3.3.2-1
losetup-2.10f-1
lsof-4.47-2
mailx-8.1.1-10
man-1.5h1-1
mingetty-0.9.4-11
mkbootdisk-1.2.5-3
mkinitrd-2.4.1-2
modutils-2.3.9-6
mount-2.10f-1
shapecfg-2.2.12-2
slang-1.2.2-5
slocate-2.1-2
stat-1.5-12
sysklogd-1.3.31-16
tar-1.13.17-3
tcp_wrappers-7.6-10
tcpdump-3.4-19
tcsh-6.09-4
time-1.7-9
tmpwatch-2.2-1
traceroute-1.4a5-18
utempter-0.5.2-2
util-linux-2.10f-7
vim-common-5.6-11
vim-minimal-5.6-11
vixie-cron-3.0.1-40
which-2.9-2
zlib-1.1.3-6
dev86-0.15.0-2
egcs-1.1.2-30
kernel-headers-2.2.14-5.0
glibc-devel-2.1.3-15

Bước thứ 2 này yêu cầu chúng ta chắc chắn không quên loại bỏ những gói dữ liệu RPM không cần thiết và thêm một vài gói dữ liệu quan trọng, những gói này cho phép bạn biên dịch những chương trình trên hệ thống. Nếu kết quả trông giống như tập tin installed_rpm ở trên, chúng ta đã yên tâm với server Linux mới này.

Định màu trên terminal của bạn

Version 6.1 Đặt một vài màu trên terminal của bạn có thể giúp cho bạn phân biệt các thư mục, file, thiết bị, các liên kết và các tập tin thực thi (executable file). Quan điểm của tôi là những màu sẽ giúp giảm bớt những lỗi và sự định hướng nhanh trong hệ thống. Đây là một vấn đề quan trọng và cần thiết chỉ cho Red Hat Linux 6.1 và những version cũ hơn, kể từ Red Hat Linux version 6.2 đặc trưng này luôn có bởi mặc định.

Hiệu chỉnh tập tin /etc/profile và thêm vào những dòng sau:

```
# Enable Colour ls
eval `dircolors /etc/DIR_COLORS -b`
export LS_OPTIONS='-s -F -T 0 --color=yes'
```

Hiệu chỉnh tập tin /etc/bashrc và thêm dòng:

```
alias ls='ls --color=auto'
```

Sau đó logout và login lại. Đến lúc này, biến môi trường COLORS mới được thiết đặt và hệ thống sẽ chấm nhận điều này.

Xin nhắc lại đặc trưng này chỉ cần cho Red Hat Linux6.1 và cũ hơn.

Cập nhật phần mềm mới nhất

Chúng ta nên giữ và cập nhật tất cả các phần mềm (đặc biệt là phần mềm mạng) với những version mới nhất. Chúng ta nên kiểm tra những trang đính chính ở http://www.RedHat.com/corp/support/errata/index.html. Những trang này có lẽ là tài nguyên tốt nhất vì đã sửa gần 90% những vấn đề chung với Red Hat. Thêm nữa các giải pháp về sửa chữa các lỗ hổng bảo mật cũng sẽ được đưa lên sau 24 giờ Red Hat được thông báo, bạn nên luôn kiểm tra web site này.

Dưới đây là những phần mềm chúng ta nên cập nhật, tuy nhiên tùy theo thời điểm cập nhật của bạn chắn chắn sẽ có những version mới hơn nên chúng ta phải đọc thêm sự chỉ dẫn và yêu cầu của chúng.

Version 6.1

groff-1_15-1_i386.rpm
sysklogd-1_3_31-14_i386.rpm
initscripts-4_70-1_i386.rpm
e2fsprogs-1.17-1.i386.rpm
pam-0_68-10_i386.rpm
gpm-1.19.1-1.i386.rpm
Linux kernel 2.2.14 (linux-2_2_14_tar.gz)

Version 6.2

gpm-1.19.1-1.i386.rpm

Chú ý: *Linux kernel là quan trọng nhất và luôn luôn phải cập nhật. Phần tiếp theo sẽ cung cấp nhiều thông tin cho việc xây dựng và tối ưu kernel cho một hệ thống cụ thể.*

- Bạn có thể kiểm tra các phần mềm RPM ở trên có được cài đặt trên hệ thống hay không trước khi cập nhật chúng, kiểm tra với lệnh:

```
[root@deep /]# rpm -q <softwarename>
```

Ở đây <softwarename> là tên của phần mềm bạn muốn kiểm tra chẳng hạn như groff, sysklogd .....

PHẦN 2

CÁC VẤN ĐỀ LIÊN QUAN VỀ BẢO MẬT VÀ TỐI ƯU

Trong phần này chúng ta sẽ đề cập đến những chương sau:

- **Tổng quan về bảo mật Linux.**
- **Tổng quan về sự tối ưu Linux.**
- **Cấu hình và xây dựng một Kernel bảo mật và được tối ưu.**

Chương 3: TỔNG QUAN VỀ BẢO MẬT LINUX (Linux General Security)

Trong chương này chúng ta sẽ đề cập về

-Tổng quan về bảo mật Linux



Tổng quan về bảo mật Linux

Giới thiệu tổng quan

Việc bảo mật an toàn cho máy Linux server phụ thuộc vào việc người quản trị hệ thống đã tạo nó như thế nào. Một khi chúng ta đã loại trừ các rủi ro an toàn tiềm ẩn bằng cách xóa các dịch vụ RPM không cần thiết, chúng ta có thể khởi động để bảo mật an toàn các dịch vụ hiện có trong hệ thống và phần mềm trên máy server. Trong chương này chúng ta sẽ thảo luận về một số vấn đề tổng quan hơn, các kỹ thuật cơ bản thường dùng để bảo mật an toàn cho hệ thống của bạn. Sau đây là một danh sách các đặc trưng mà có thể được dùng để ngăn chặn các cuộc tấn công từ bên ngoài và các nguồn bên trong.

Bảo mật BIOS – Đặt mật khẩu khởi động

Có một lời khuyên rằng bạn nên vô hiệu hoá việc khởi động (booting) từ ổ đĩa mềm đồng thời đặt các mật khẩu (password) trên các đặc tính của BIOS. Bạn có thể xem xét tài liệu hướng dẫn về BIOS của bạn hay xem nó khởi động ở lần kế tiếp như thế nào. Việc vô hiệu hoá khả năng khởi động từ các ổ đĩa mềm và có cài đặt một mật khẩu để truy cập các tính năng BIOS sẽ cải tiến tính an toàn bảo mật hệ thống của bạn. Điều này sẽ làm cho một số người không mong đợi cố gắng thay đổi tính năng BIOS để cho phép khởi động từ ổ đĩa mềm hay khởi động từ máy server luôn bị nhắc nhở mật khẩu.

Chính sách bảo mật

Một điều quan trọng cần phải chỉ ra là bạn không thể thực hiện việc an toàn và bảo mật hệ thống nếu bạn chưa xác định được là cần phải bảo vệ những gì, và cho ai. Do vậy bạn cần phải có một chính sách bảo mật – đó là danh sách những gì mà bạn cho phép quan tâm đến và cũng như những gì mà bạn không cho phép quan tâm đến và dựa trên cơ sở đó bạn có thể có bất kỳ quyết định nào về việc bảo mật một cách chính xác. Chính sách cũng nên xác định rõ việc đáp lại của bạn các đối với vi phạm về việc bảo mật và an toàn. Bạn cũng nên để ý rằng những gì được đề ra khi xây dựng một chính sách bảo mật sẽ phụ thuộc vào việc bạn đưa ra các định nghĩa về bảo mật. Sau đây là các câu hỏi sẽ cung cấp một số các chỉ dẫn tổng thể liên quan:

- Làm thế nào bạn phân biệt được các thông tin là nhạy cảm và bí mật?
- Hệ thống có nên chứa đựng các thông tin nhạy cảm và bí mật không?
- Chính xác là bạn cần phải đề phòng ai?
- Những người sử dụng từ xa có thật sự cần thiết truy cập tới hệ thống của bạn không?
- Mật khẩu và giải pháp mã hoá đã đủ bảo vệ hệ thống chưa?
- Bạn có cần phải truy cập Internet không?

- Bạn muốn cho phép hệ thống của bạn có thể được truy cập từ Internet bao nhiêu lần ?
- Bạn sẽ làm gì khi phát hiện ra một sự vi phạm trong việc bảo mật của bạn?

Danh sách này chỉ liệt kê một ít, và chính sách của bạn có thể sẽ bao gồm nhiều hơn trước khi nó được hoàn thành. Bất kỳ chính sách bảo mật nào cũng phải dựa trên một số mức độ còn nghi ngờ và xác định mức độ tin tưởng của bạn với mọi người như thế nào trong cả hai môi trường bên ngoài và bên trong cơ quan của bạn. Tuy nhiên, chính sách phải có một sự cân đối giữa việc cho phép những người sử dụng hệ thống một cách hợp lệ để truy cập thông tin mà họ cần để thực hiện các công việc của họ và hoàn toàn không được phép truy cập thông tin khác không liên quan đến họ. Điểm này là điều đáng lưu ý để xác định được chính sách bảo mật của bạn.

Chọn một mật khẩu (password) đúng

Điểm bắt đầu cho một chuyến tham quan về Linux General Security là mật khẩu. Nhiều người lưu giữ các thông tin và các tập tin đáng giá của họ trên một máy tính, và chỉ có một việc là ngăn chặn những người khác khi muốn xem các thông tin hay tập tin đó thì yêu cầu gõ vào một chuỗi dài tám ký tự hoặc nhiều hơn thường được gọi là một mật khẩu. Không như mọi người đã nghĩ, không thể có một mật khẩu nào không giải được. Với thời gian và công sức, tất cả mật khẩu đều có thể được đoán ra, kể cả việc dùng các mưu mẹo trong cuộc sống hay áp lực.

Dùng những mưu mẹo trong cuộc sống để có được các mật khẩu và các phương pháp truy cập khác vẫn là cách phổ biến nhất và dễ nhất để giành được quyền truy cập tới các tài khoản và các máy server. Thông thường, mọi thứ trở nên đơn giản hơn khi đóng vai như một người cấp trên hay có quyền hành trong một công ty nhắc nhở đúng người vào đúng thời điểm trong ngày và thu lại kết quả rất lớn.

Chạy một chương trình bẻ khóa mật khẩu mỗi tuần một lần dựa trên hệ thống của bạn là một ý tưởng hay. Việc này giúp bạn tìm và thay thế các mật khẩu mà nó dễ dàng được đoán ra hay có hiệu lực kém. Cũng thế, một cơ chế kiểm tra mật khẩu là nên có để loại bỏ những mật khẩu kém hiệu lực khi lần đầu tiên chọn một mật khẩu hay thay đổi một mật khẩu cũ. Chúng ta không nên chấp nhận một mật khẩu mới với chuỗi ký tự là các từ dễ hiểu, hoặc là tất cả các ký tự đều có định dạng in hay thường giống nhau, hoặc không chứa đựng các số hay các ký tự đặc biệt.

Chúng ta nên tuân theo các nguyên tắc sau để tạo mật khẩu một cách hiệu quả:

- Có chiều dài ít nhất là sáu ký tự, được ưa chuộng nhất là tám ký tự trong đó có ít nhất một ký tự số hay ký tự đặc biệt.
- Chúng phải không tầm thường; một mật khẩu tầm thường là một mật khẩu dễ dàng đoán ra và nó thường dựa trên tên, họ, nghề nghiệp hoặc một số đặc điểm cá nhân khác của người sử dụng.

- Chúng nên có hiệu lực trong một giai đoạn, yêu cầu thay đổi mật khẩu mới sau một thời gian đã định.
- Chúng nên được hủy bỏ hay tạo lại sau khi có một số lần thử không thành công.

Chiều dài mật khẩu

Chiều dài của mật khẩu tối thiểu có thể chấp nhận được mặc định khi bạn cài đặt hệ thống Linux là 6. Điều này nói lên rằng khi một người sử dụng mới thì được phép truy cập trên máy server, chiều dài mật khẩu của người sử dụng đó sẽ có tối thiểu là 6 ký tự và nó bao gồm các ký tự, ký số, ký tự đặc biệt v.v..Như thế vẫn chưa đủ mà nó phải là 8. Để ngăn chặn những người có tư tưởng không chú trọng đến bảo mật và bắt buộc họ phải dùng chiều dài password tối thiểu 8 ký tự, người quản trị hệ thống cũng có thể hiệu chỉnh tập tin quan trọng "/etc/login.defs" và thay đổi dòng sau trở thành:

```
PASS_MIN_LEN 6
```

trở thành :

```
PASS_MIN_LEN 8
```

"login.defs" là tập tin cấu hình cho chương trình login. Bạn nên xem lại hay thay đổi nội dung tập tin này cho phù hợp với hệ thống riêng biệt của bạn. Đây là nơi mà bạn đặt các cài đặt chính sách bảo mật khác (giống như sự mặc định hết hạn của mật khẩu hay chiều dài tối thiểu mật khẩu có thể chấp nhận).

Tài khoản root

Tài khoản root là được gán quyền nhiều nhất trên một hệ thống Unix. Tài khoản "root" không có một hạn chế bảo mật nào. Điều này có nghĩa là hệ thống cho rằng bạn biết những gì bạn đang làm, và sẽ thực hiện chính xác những gì bạn yêu cầu – không có câu hỏi nào được đặt ra về những điều đó. Chính vì vậy, nó rất dễ dàng với một câu lệnh nhập sai có thể sẽ xoá sạch tất cả các tập tin hệ thống. Vì vậy khi sử dụng tài khoản này bạn phải tuyệt đối cẩn thận tránh những lỗi lầm đáng tiếc. Để phục vụ cho các lý do bảo mật không cần phải log-in vào máy server của bạn như là "root" trừ khi nó tuyệt đối cần thực hiện một công việc nào đó cần phải có quyền truy cập "root". Hơn nữa, nếu bạn không có làm việc trên máy server của bạn, thì đừng nên vào và ra khỏi hệ thống như là "root"- điều này là rất dở.

Đặt thời gian time-out login cho tài khoản root

Mặc dù được lưu ý là không nên vào hệ thống với account là "root" và logout khi không làm việc trên máy server, tuy nhiên vẫn có những người quản trị hệ thống vẫn còn lưu lại màn hình login như "root" hay quên ra khỏi hệ thống (logout) sau khi kết thúc công việc của họ hoặc rời khỏi các máy trạm không cùng với tài khoản nào. Để giải quyết vấn đề này ta tạo một bash shell tự động logout sau khi không được dùng trong một khoảng thời gian. Để thực hiện điều đó, bạn phải đặt biến đặc biệt của Linux được đặt tên là "TMOUT" tới thời gian tính bằng giây không có thao tác nào được nhập vào trước khi logout.

Chỉnh sửa tập tin **profile** của bạn "vi /etc/profile" và thêm dòng sau vào một nơi nào đó bên dưới dòng chứa "HISTFILESIZE=" trong tập tin này:

TMOUT=7200

Giá trị mà chúng ta nhập vào cho biến "TMOUT=" là được tính bằng giây và biểu thị 2 giờ (60*60=3600*2=7200 giây). Điều quan trọng là hãy ghi chú rằng nếu bạn quyết định đặt dòng trên vào trong tập tin "/etc/profile", khi đó việc tự động logout sau 2 giờ nếu không có các hành động nào sẽ được áp dụng cho tất cả mọi người sử dụng trên hệ thống. Nếu bạn thích điều khiển việc tự động logout chỉ cho một user cụ thể nào đó bạn có thể đặt biến này trong tập tin riêng ".bashrc" của họ.

Sau khi tham số này đã được đặt lại trên hệ thống, bạn phải logout và login lại lần nữa (như root) cho việc thay đổi có tác dụng.

Tập tin "/etc/exports"

Nếu bạn đang xuất ra các hệ thống tập tin dùng dịch vụ NFS, thì bảo đảm cấu hình tập tin "/etc/exports" với hạn chế truy cập tối đa. Điều này có nghĩa là không dùng các ký tự đại diện (wild-cards), không cho phép root truy xuất ghi, và chỉ mount dưới dạng chỉ đọc khi có thể.

Soạn thảo tập tin exports (vi /etc/exports) và thêm vào:

Lấy một ví dụ:

```
/dir/to/export host1.mydomain.com(ro,root_squash)
/dir/to/export host2.mydomain.com(ro,root_squash)
```

Trong đó **"/dir/to/export"** là thư mục mà bạn muốn xuất ra (export), **host#.mydomain.com** là máy được phép login vào thư mục, tùy chọn **<ro>** nghĩa là mount chỉ đọc và tùy chọn **<root_squash>** là không cho phép root truy xuất ghi trong thư mục này.

Để việc thay đổi này có tác dụng bạn sẽ cần chạy lệnh sau trên máy trạm của bạn:

```
[root@deep]# /usr/sbin/exportfs -a
```

Ghi chú: *Bạn nên nhớ rằng, mặc định có một dịch vụ NFS có sẵn trên hệ thống của bạn có thể có một rủi ro về bảo mật. Theo ý kiến cá nhân tôi khuyên bạn không nên sử dụng nó.*

Vô hiệu hoá việc truy cập chương trình console

Trong một môi trường an toàn nơi mà chúng ta chắc rằng console được bảo mật bởi vì các mật khẩu cho BIOS và LILO đã được thiết lập và tất cả các công tắc nguồn và reset của hệ thống đã bị vô hiệu hoá, nó có thể có lợi để vô hiệu hoá hoàn toàn tất cả các console-equipvalent truy xuất tới các chương trình như tắt hẳn hệ thống (shutdown), khởi động lại hệ thống (reboot), và treo hệ thống (halt) cho những người sử dụng hợp lệ trên máy server của bạn.

Để làm việc này, chạy dòng lệnh sau:

```
[root@deep /]# rm -f /etc/security/console.apps/<servicename>
```

Trong đó <servicename> là tên của chương trình mà bạn muốn vô hiệu hóa truy cập console-equivalent. Tuy nhiên, trừ khi bạn dùng xdm, hãy cẩn thận không xóa tập tin xserver hoặc là không ai ngoài root có thể khởi động hệ thống xserver. (Nếu bạn luôn dùng xdm để khởi động xserver, root là người sử dụng duy nhất cần để khởi động X, trong trường hợp này, bạn thật sự muốn xóa tập tin xserver.)

Như ví dụ sau:

```
[root@deep /]# rm -f /etc/security/console.apps/halt
[root@deep /]# rm -f /etc/security/console.apps/poweroff
[root@deep /]# rm -f /etc/security/console.apps/reboot
[root@deep /]# rm -f /etc/security/console.apps/shutdown
```

[root@deep /]# rm -f /etc/security/console.apps/xserver (nếu xóa bỏ, root sẽ là người duy nhất có thể khởi động X).

Việc này sẽ vô hiệu hóa console-equivalent truy cập tới các chương trình halt, poweroff, reboot và shutdown. Một lần nữa, chương trình xserver chỉ áp dụng khi mà bạn đã cài đặt giao diện Xwindow trên hệ thống của bạn.

Ghi chú: *Nếu bạn cài đặt theo các bước của chương trình cài đặt của chúng ta, thì giao diện Xwindow không được cài đặt trên máy server của bạn và tất cả các tập tin được mô tả sẽ không xuất hiện trong thư mục "/etc/security". do vậy đừng chú ý các bước trên.*

Vô hiệu hóa tất cả truy cập console

Thư viện Linux-PAM được cài đặt mặc định trên hệ thống của bạn cho phép các người quản trị hệ thống chọn cách để các ứng dụng xác nhận những người sử dụng như thế nào, ví dụ như truy cập console, truy cập tập tin và chương trình. Để vô hiệu hóa tất cả các truy cập này cho những người sử dụng, bạn phải đánh dấu # và đầu tất cả các dòng có tham chiếu tới **pam_console.so** trong thư mục "/etc/pam.d" (nghĩa là bạn chuyển các dòng đó thành dòng ghi chú (comment line)). Bước này là sự tiếp tục cho bước "vô hiệu hóa việc truy cập chương trình console".

Đoạn script sau sẽ thực hiện mánh lới một cách tự động cho bạn. Đăng nhập như "root", tạo tập tin script **disabling.sh** (gõ disabling.sh) và thêm các dòng sau vào bên trong:

```
# !/bin/sh
cd /etc/pam.d
for i in * ; do
sed '/[^#].*pam_console.so/s/^/#/' < $i > foo && mv foo $i
done
```

Làm script này có thể thực thi với lệnh sau và thi hành nó:

```
[root@deep /]# chmod 700 disabling.sh
[root@deep /]# ./disabling.sh
```

Việc này sẽ đánh dấu là dòng ghi chú (comment line) lên tất cả các dòng có tham chiếu đến "pam_console.so" trong tất cả các tập tin nằm trong thư mục /etc/pamd. Sau khi script được thi hành, bạn có thể xóa nó khỏi hệ thống của bạn.

Tập tin "/etc/inetd.conf"

Inetd cũng được gọi "super server", sẽ nạp một chương trình mạng dựa trên một yêu cầu từ một hệ thống mạng. Tập tin "inetd.conf" báo cho inetd biết cổng (port) nào dùng để lắng nghe và server nào khởi động cho mỗi port. Việc đầu tiên cần xem xét là khi mà bạn đặt hệ thống Linux của bạn lên bất kỳ mạng nào thì các dịch vụ (service) nào bạn cần đưa ra.

Các dịch vụ mà bạn không cần đưa ra nên vô hiệu hóa và gỡ bỏ, như vậy bạn sẽ có rất ít điều lo ngại về nó, và những kẻ tấn công sẽ có ít cơ hội để tìm kiếm lỗ hổng trên hệ thống của bạn. Nhìn vào tập tin "/etc/inetd.conf" để xem những dịch vụ nào mà nó được cung cấp bởi chương trình inetd của bạn. Vô hiệu hóa những gì mà bạn không cần bằng cách thêm một dấu # ở ngay đầu dòng, và sau đó gửi đến tiến trình inetd của bạn một lệnh SIGHUP để cập nhật nó tới tập tin "inetd.conf" hiện hành.

Bước 1

Thay đổi các quyền hạn trên tập tin này là 600.

```
[root@deep /]# chmod 600 /etc/inetd.conf
```

Bước 2

Bảo đảm rằng người sở hữu là root.

```
[root@deep /]# stat /etc/inetd.conf

File: "/etc/inetd.conf"
Size: 2869 Filetype: Regular File
Mode: (0600/-rw-------) Uid: ( 0/ root) Gid: ( 0/ root)
Device: 8,6 Inode: 18219 Links: 1
Access: Wed Sep 22 16:24:16 1999(00000.00:10:44)
Modify: Mon Sep 20 10:22:44 1999(00002.06:12:16)
Change: Mon Sep 20 10:22:44 1999(00002.06:12:16)
```

Bước 3

Chỉnh sửa tập tin inetd.conf (vi /etc/inetd.conf) và vô hiệu hóa các dịch vụ như sau:

ftp, telnet, shell, login, exec, talk, imap, pop-2, finger, auth, v.v.. trừ khi bạn có kế hoạch dùng nó. Nếu nó bị tắt thì nó sẽ bớt rủi ro.

```
# To re-read this file after changes, just do a 'killall -HUP inetd'
#
#echo stream tcp nowait root internal
#echo dgram udp wait root internal
#discard stream tcp nowait root internal
#discard dgram udp wait root internal
#daytime stream tcp nowait root internal
#daytime dgram udp wait root internal
#chargen stream tcp nowait root internal
#chargen dgram udp wait root internal
#time stream tcp nowait root internal
#time dgram udp wait root internal
#
# These are standard services.
#
#ftp stream tcp nowait root /usr/sbin/tcpd in.ftpd -l -a
#telnet stream tcp nowait root /usr/sbin/tcpd in.telnetd
#
# Shell, login, exec, comsat and talk are BSD protocols.
#
#shell stream tcp nowait root /usr/sbin/tcpd in.rshd
#login stream tcp nowait root /usr/sbin/tcpd in.rlogind
#exec stream tcp nowait root /usr/sbin/tcpd in.rexecd
#comsat dgram udp wait root /usr/sbin/tcpd in.comsat
#talk dgram udp wait root /usr/sbin/tcpd in.talkd
#ntalk dgram udp wait root /usr/sbin/tcpd in.ntalkd
#dtalk stream tcp wait nobody /usr/sbin/tcpd in.dtalkd
#
# Pop and imap mail services et al
#
#pop-2 stream tcp nowait root /usr/sbin/tcpd ipop2d
#pop-3 stream tcp nowait root /usr/sbin/tcpd ipop3d
#imap stream tcp nowait root /usr/sbin/tcpd imapd
#
```

```
# The Internet UUCP service.
#
#uucp stream tcp nowait uucp /usr/sbin/tcpd /usr/lib/uucp/uucico -l
#
# Tftp service is provided primarily for booting. Most sites
# run this only on machines acting as "boot servers." Do not uncomment
# this unless you *need* it.
#
#tftp dgram udp wait root /usr/sbin/tcpd in.tftpd
#bootps dgram udp wait root /usr/sbin/tcpd bootpd
#
# Finger, systat and netstat give out user information which may be
# valuable to potential "system crackers." Many sites choose to disable
# some or all of these services to improve security.
#
#finger stream tcp nowait root /usr/sbin/tcpd in.fingerd
#cfinger stream tcp nowait root /usr/sbin/tcpd in.cfingerd
#systat stream tcp nowait guest /usr/sbin/tcpd /bin/ps -auwwx
#netstat stream tcp nowait guest /usr/sbin/tcpd /bin/netstat -f inet
#
# Authentication
#
#auth stream tcp nowait nobody /usr/sbin/in.identd in.identd -l -e -o
#
# End of inetd.conf
```

Ghi chú: *Đừng quên gửi tới tiến trình inetd của bạn một tín hiệu SIGHUP (killall -HUP inetd) sau khi thực hiện các thay đổi trên tập tin inetd.conf của bạn. Các dịch vụ mà bạn kích hoạt trên một host được chọn phụ thuộc vào các chức năng mà bạn muốn host cung cấp. Các chức năng có thể hỗ trợ dịch vụ mạng được chọn, các dịch vụ khác được host trên máy tính này, hay việc phát triển và bảo trì của hệ điều hành và các ứng dụng.*

```
[root@deep /]# killall -HUP inetd
```

Bước 4

Một biện pháp bảo mật nữa mà bạn có thể áp dụng để bảo mật tập tin "**inetd.conf**" là đặt nó không thay đổi, dùng lệnh **chattr**.

- Để đặt tập tin không thay đổi gì, thi hành lệnh sau:

```
[root@deep /]# chattr +i /etc/inetd.conf
```

Việc này sẽ ngăn chặn bất kỳ sự thay đổi (xảy ra bất ngờ hay trong các trường hợp khác) với tập tin "inetd.conf". Một tập tin với thuộc tính không thay đổi "i" sẽ làm cho tập tin không thể hiệu chỉnh, xóa hay đổi tên, không có mối liên kết nào có thể được tạo cho tập tin này và không dữ liệu nào có thể được ghi đến nó. Chỉ có một người có thể đặt hay xóa thuộc tính này là super-user root.

Nếu bạn muốn hiệu chỉnh tập tin inetd.conf bạn sẽ cần phải gỡ bỏ cờ hiệu không được thay đổi:

- Để gỡ cờ hiệu không được thay đổi, đơn giản thi hành lệnh sau:

```
[root@deep /]# chattr -i /etc/inetd.conf
```

TCP_WARPPERS

Mặc nhiên Red Hat Linux cho phép tất cả các yêu cầu dịch vụ. Dùng TCP_WRAPPERS thực hiện bảo mật các máy server của bạn để chống sự xâm phạm từ bên ngoài thì đơn giản và dễ dàng hơn rất nhiều những gì bạn suy nghĩ. Từ chối tất cả các host bằng cách đặt "ALL:ALL@ALL, PARANOID" trong tập tin "/etc/hosts.deny" và đồng thời liệt kê tường minh danh sách các host đáng tin cậy đã được phép truy cập tới máy của bạn vào trong tập tin "/etc/hosts.allow". Đây là cách cấu hình an toàn tốt nhất.

TCP_WRAPPERS được điều khiển từ hai tập tin và việc tìm kiếm dừng lại ở lần thấy đầu tiên.

```
/etc/hosts.allow
/etc/hosts.deny
```

- Quyền truy cập được gán khi một cặp (daemon, client) xứng hợp với một mục trong tập tin /etc/hosts.allow.
- Bằng không, quyền truy cập bị từ chối khi một cặp (daemon, client) xứng hợp với một mục trong tập tin /etc/hosts.deny.
- Còn lại, quyền truy cập sẽ được gán.

Bước 1

Soạn thảo tập tin hosts.deny (vi /etc/hosts.deny) và thêm vào các dòng sau :

```
Access is denied by default.
# Deny access to everyone.
ALL: ALL@ALL, PARANOID # Matches any host whose name does not
                       # match its address, see bellow.
```

Điều này có ý nghĩa cho tất cả các dịch vụ, tất cả các vị trí, do vậy bất kỳ dịch vụ nào đều bị ngăn chặn lại, trừ khi chúng được phép truy cập một cách tường minh bởi các mục trong tập tin allow.

Ghi chú: *Về tham số "**PARANOID**"; Nếu bạn có ý định muốn chạy các dịch vụ telnet hoặc ftp trên máy server, đừng quên thêm vào tên máy client và địa chỉ IP trong tập tin "/etc/hosts" trên server hay bạn có thể chờ đợi trong nhiều phút cho việc tìm kiếm DNS tới thời gian time out, trước khi bạn có được một dấu nhắc login.*

Bước 2:

Soạn thảo tập tin **hosts.allow** (vi /etc/hosts.allow) và giả sử thêm vào dòng sau:

Host được ủy quyền rõ ràng thì được liệt kê trong tập tin allow.

Như ví dụ sau:

sshd: 208.164.186.1 gate.openna.com

Đối với máy client: 208.164.186.1 là địa chỉ IP và gate.openna.com là tên host của một trong client của bạn được phép dùng dịch vụ sshd.

Bước 3

Chương trình tcpdchk là trình kiểm tra cấu hình tcpd wrapper. Nó thăm dò cấu hình tcp wrapper và báo về tất cả các vấn đề hiện có hay vấn đề tiềm ẩn mà nó tìm ra được.

- Sau khi việc cấu hình hoàn tất, chạy chương trình **tcpdchk.**

```
[root@deep /]# tcpdchk
```

Ghi chú: *Các thông điệp lỗi có thể trông giống như thế này: warning: /etc/hosts.allow, line 6: can't verify hostname: gethostbyname(win.openna.com) failed. Nếu bạn nhận kiểu thông điệp lỗi này, hãy kiểm tra trong tập tin cấu hình DNS để biết sự tồn tại của hostname.*

Đừng để hiển thị các tập tin thông tin hệ thống.

Nếu bạn không muốn hiển thị các tập tin thông tin hệ thống khi mọi người log-in từ xa, bạn có thể thay đổi tùy chọn telnet trong tập tin "/etc/inetd.conf" trông giống như thế:

```
telnet stream tcp nowait root /usr/sbin/tcpd in.telnetd -h
```

Thêm vào cờ hiệu "-h" ở cuối dòng lệnh sẽ gây cho daemon không hiển thị bất kỳ thông tin hệ thống nào và chỉ hiện dấu nhắc login cho người sử dụng. Mẹo vặt này chỉ cần thiết nếu bạn dùng một telnet daemon trên máy server của bạn (thay vì thế bạn được khuyên dùng SSH).

Tập tin "/etc/host.conf"

Linux sử dụng một thư viện phân giải để lấy một địa chỉ IP tương ứng với một host name. Tập tin "/etc/host.conf" chỉ rõ các tên được phân giải như thế nào.

Các mục trong tập tin "etc/host.conf" bảo các thư viện phân giải các dịch vụ nào cần dùng, và sắp theo thứ tự nào để phân giải các tên.

Soạn thảo tập tin **host.conf** (ví /etc/host.conf) và thêm vào các dòng sau:

```
# Lookup names via DNS first then fall back to /etc/hosts.
order bind,hosts
# We have machines with multiple IP addresses.
multi on
# Check for IP address spoofing.
nospoof on
```

Tùy chọn **order** chỉ rõ thứ tự của các dịch vụ. Ví dụ mẫu này chỉ rõ rằng thư viện phân giải đầu tiên nên tham khảo các name server để phân giải một tên (name) và sau đó kiểm tra tập tin "/etc/hosts". Để đạt hiệu quả cao và bảo mật tốt hơn bạn nên đặt thư viện phân giải theo thứ tự đầu tiên là kiểm tra name server (bind) sau đó tập tin hosts (hosts) cho việc thực thi và bảo mật được tốt hơn trên tất cả các máy server của bạn. Dĩ nhiên bạn phải có phần mềm DNS/BIND được cài đặt sẵn hay việc cấu hình sẽ không làm việc.

Tùy chọn **multi** xác định một host trong tập tin "/etc/hosts" có thể có nhiều địa chỉ IP (multiple interface ethN). Các host mà có nhiều hơn một địa chỉ IP được biết đến là multiomed, bởi vì sự hiện diện của nhiều địa chỉ IP bao hàm rằng host có nhiều giao tiếp mạng. Lấy một ví dụ, một Gateway Server sẽ luôn có nhiều địa chỉ IP và phải có tùy chọn này được đặt là ON.

Tùy chọn **nospoof** chỉ rõ hãy cẩn thận không được phép ngụy trang trên máy này. IP-Spoofing là một thành tích bảo mật chói lọi, nó làm việc bởi các mẹo vặt máy tính trong một mối quan hệ đáng tin cậy mà bạn là một người nào đó mà thật sự không phải vậy. Trong loại hình tấn công này, một máy được cài đặt như một server hợp pháp và khi đó sinh ra các kết nối và các loại hoạt động mạng khác tới các hệ thống đầu cuối hợp pháp, các server khác hay các hệ thống lưu trữ dữ liệu lớn. Tùy chọn này phải được đặt ON cho tất cả các loại server.

Tập tin "/etc/services"

Các số cổng (port number) được dùng cho các dịch vụ chuẩn được đưa ra trong RFC 1700 "Assigned Numbers". Tập tin "/etc/services" cho phép các chương trình server và client để chuyển đổi tên dịch vụ tới các số port này. Danh sách này thì được lưu giữ trong mỗi host trong tập tin "/etc/services". Chỉ người sử dụng "root" là được phép thay đổi nội dung tập tin này và rất hiếm khi phải chỉnh sửa tập tin "/etc/services" để tạo sự thay đổi kể từ khi nó đã chứa đựng nhiều tên dịch vụ tương ứng tới các số port. Để cải tiến việc bảo mật, chúng ta có thể làm tập tin này trở nên không bị chỉnh sửa (immmunize) để ngăn chặn việc xóa hay thêm vào các dịch vụ mà không được ủy quyền.

- Để định cho tập tin "/etc/services" không thay đổi, dùng lệnh :

```
[root@deep /]# chattr +i /etc/services
```

Tập tin "/etc/securetty"

Tập tin "/etc/securetty" cho phép bạn chỉ rõ các thiết bị **TTY** nào mà người sử dụng "root" được phép login vào. Tập tin "/etc/securetty" được đọc bởi chương trình login (thường là "/bin/login"). Định dạng của nó là một danh sách các tên thiết bị tty được phép, và đối với tất cả các thứ khác thì được chú giải hoặc không xuất hiện trong tập tin này, đăng nhập với root thì không được phép.

Vô hiệu hóa bất kỳ tty mà bạn không cần đến bằng cách đặt dấu chú giải (#) cho chúng (dấu # ở đầu dòng).

Chỉnh sửa tập tin **securetty** (vi /etc/securetty) và đặt dấu chú giải lên các dòng sau :

```
tty1
#tty2
#tty3
#tty4
#tty5
#tty6
#tty7
#tty8
```

Có nghĩa là root chỉ được phép login vào tty1. Đây là lời khuyên của chúng tôi, cho phép "root" chỉ login vào một thiết bị tty và dùng lệnh "su" để chuyển sang "root" nếu bạn cần nhiều quyền hơn. Các thiết bị phải log-in như "root"

Các tài khoản đặc biệt

Có một điều quan trọng là cần **vô hiệu hóa tất cả các tài khoản mặc định của nhà sản xuất mà** bạn không dùng trong hệ thống của bạn (một số tài khoản tồn tại mặc nhiên ngay cả khi bạn không cài đặt các dịch vụ liên quan trên máy server của bạn). Việc này nên được kiểm tra sau mỗi lần nâng cấp hay cài đặt phần mềm mới. Linux cung cấp các tài khoản này cho các hoạt động hệ thống khác nhau, bạn có thể không cần nếu các dịch vụ không được cài đặt trên hệ thống server của bạn. Nếu bạn không cần các tài khoản nào, thì xóa chúng. Bạn có càng nhiều tài khoản thì việc truy cập hệ thống của bạn sẽ dễ dàng hơn.

Chúng tôi cho rằng bạn đang dùng bộ mật khẩu Shadow trên hệ thống Linux. Nếu bạn không có dùng, thì bạn nên quan tâm đến việc này, và nó sẽ giúp làm chặt chẽ việc bảo mật hơn. Việc này đã sẵn có nếu bạn đi theo sự cài đặt Linux của chúng ta như ở phần trên và chọn dưới tùy chọn (option) "Authentication Configuration" sang chế độ "Enable Shadow Passwords" (Xem chương 2 Cài đặt Linux Server để biết thêm thông tin chi tiết).

- Để xoá người sử dụng trên hệ thống của bạn, dùng lệnh:

```
[root@deep /]# userdel username
```

- Để xoá nhóm người sử dụng trên hệ thống của bạn, dùng lệnh:

```
[root@deep /]# groupdel groupname
```

Bước 1

Gõ các lệnh sau trên máy terminal của bạn để xóa các người sử dụng được liệt kê bên dưới :

[root@deep /]# **userdel adm**

[root@deep /]# **userdel lp**

[root@deep /]# **userdel sync**

[root@deep /]# **userdel shutdown**

[root@deep /]# **userdel halt**

[root@deep /]# **userdel news**

[root@deep /]# **userdel uucp**

[root@deep /]# **userdel operator**

[root@deep /]# **userdel games** (*Xóa user này nếu bạn không dùng X Window*).

[root@deep /]# **userdel gopher**

[root@deep /]# **userdel ftp** (*Xóa user này nếu bạn không dùng ftp anonymous server*).

Ghi chú: *Mặc định, lệnh "userdel" sẽ không xoá một thư mục home của người sử dụng. Nếu bạn muốn thư mục home của các tài khoản cũng được xóa, thì phải thêm tùy chọn "-r" vào lệnh userdel.*

Bước 2

Gõ các lệnh sau trên máy terminal của bạn để xóa nhóm các người sử dụng được liệt kê bên dưới :

[root@deep /]# **groupdel adm**

[root@deep /]# **groupdel lp**

[root@deep /]# **groupdel news**

[root@deep /]# **groupdel uucp**

[root@deep /]# **groupdel games** (*Xóa group này nếu bạn không dùng X Window*).

[root@deep /]# **groupdel dip**

[root@deep /]# **groupdel pppusers**

[root@deep /]# **groupdel popusers** (*Xóa group này nếu bạn không dùng pop server cho email*).

[root@deep /]# **groupdel slipusers**

Bước 3

Thêm các người sử dụng cần thiết vào hệ thống:

- Để thêm một người sử dụng mới trên hệ thống của bạn, dùng lệnh sau:

  [root@deep /]# **useradd username**

- Để thêm hay thay đổi mật khẩu của người sử dụng trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# passwd username
```

Lấy ví dụ:

```
[root@deep /]# useradd admin
[root@deep /]# passwd admin
```

Các dòng kết quả được in ra như thế này:

```
Changing password for user admin
New UNIX password: somepasswd
Re-type.....
passwd: all authentication tokens updated successfully
```

Bước 4

Bit không thay đổi có thể được dùng để ngăn chặn việc xoá ngẫu nhiên hay ghi chồng lên một tập tin cần được bảo vệ. Nó cũng ngăn chặn một ai đó tạo một biểu tượng liên kết tới tập tin này, mà nó là nguồn tấn công xoay quanh việc xoá "/etc/passwd", "/etc/shadow", "/etc/group" hay "/etc/gshadow".

- Để đặt bit không thay đổi trên các mật khẩu và các nhóm tập tin, dùng lệnh sau:

```
[root@deep /]# chattr +i /etc/passwd
[root@deep /]# chattr +i /etc/shadow
[root@deep /]# chattr +i /etc/group
[root@deep /]# chattr +i /etc/gshadow
```

Ghi chú: *Trong tương lai, nếu bạn có ý định thêm hay xoá các người sử dụng, tập tin passwd, các nhóm người sử dụng, hay các tập tin group, thì bạn phải gỡ bỏ bit không làm thay đổi trên tất cả các tập tin đó, nếu không bạn sẽ không thể thay đổi được gì. Nếu như bạn cũng có dự định cài đặt một chương trình RPM sẽ tự động thêm một người sử dụng mới vào các tập tin passwd hay group đã đặt bit không thay đổi, bạn sẽ nhận được một thông điệp lỗi suốt quá trình cài đặt, khi bạn không gỡ bỏ bit không làm thay đổi từ các tập tin đó.*

Ngăn chặn bất kỳ người sử dụng nào chuyển thành root bằng lệnh "su".

Lệnh su (Substitute User) cho phép bạn trở thành những người sử dụng tồn tại trên hệ thống. Ví dụ bạn có thể trở thành "root" tạm thời và thi hành các lệnh như một super-user "root'. Nếu bạn không muốn bất kỳ ai chuyển thành root hay hạn chế lệnh "su" với một số người sử dụng nào đó, bạn hãy thêm vao hai dòng như sau vào đầu tập tin cấu hình "su" của bạn trong thư mục "/etc/pam.d". Chúng tôi rất muốn khuyên bạn rằng bạn nên giới hạn những người được phép chạy lệnh "su" để thành tài khoản root.

Bước 1

Soạn thảo tập tin **su** (vi /etc/pam.d/su) và thêm vào hai dòng sau vào phía trên đầu tập tin:

```
auth sufficient /lib/security/pam_rootok.so debug
auth required /lib/security/pam_wheel.so group=wheel
```

Sau khi thêm vào các dòng trên, tập tin "/etc/pam.d/su" sẽ trông giống như thế này:

```
#%PAM-1.0
auth        sufficient  /lib/security/pam_rootok.so debug
auth        required    /lib/security/pam_wheel.so group=wheel
auth        required    /lib/security/pam_pwdb.so shadow nullok
account     required    /lib/security/pam_pwdb.so
password    required    /lib/security/pam_cracklib.so
password    required    /lib/security/pam_pwdb.so shadow use_authtok nullok
session     required    /lib/security/pam_pwdb.so
session     optional    /lib/security/pam_xauth.so
```

Có nghĩa là chỉ có những người sử dụng là thành viên của nhóm "**wheel**" có thể dùng lệnh su chuyển thành root; nó cũng bao gồm logging. Chú ý rằng nhóm "wheel" là một tài khoản đặc biệt trên hệ thống của bạn mà cũng có thể được dùng cho mục đích này. Bạn không thể dùng bất kỳ tên nhóm mà bạn muốn để thực hiện mánh khóe này. Mánh khóe này được kết hợp với việc chỉ rõ các thiết bị TTY nào mà root được phép login vào sẽ cải tiến rất nhiều việc bảo mật trên hệ thống.

Bước 2

Bây giờ chúng ta phải định nghĩa nhóm "wheel" trong việc cấu hình tập tin "/etc/pam.d/su", đã đến lúc phải thêm vào một số người sử dụng được phép dùng "su" chuyển thành tài khoản "root". Ví dụ như bạn muốn tạo người sử dụng "admin" một thành viên của nhóm "wheel", và chính vì vậy để có thể "su" thành root, dùng lệnh sau:

```
[root@deep /]# usermod -G10 admin
```

Trong đó "G" có nghĩa là một danh sách các nhóm phụ, ở đó người sử dụng cũng là một thành viên. "10" là giá trị số của ID của người sử dụng "wheel", và "admin" là người sử dụng mà chúng ta muốn thêm vào nhóm "wheel". Sử dụng cùng lệnh trên cho tất cả các người sử dụng trên hệ thống của bạn mà bạn muốn họ có thể "su" thành tài khoản root.

***Ghi chú**: Nếu bạn không thể su trong một máy terminal GNOME, đó là vì bạn đã dùng terminal không đúng.(Do vậy đừng nghĩ rằng lời nhắc này là không đúng, đơn giản bởi vì vấn đề là ở một máy terminal).*

Các giới hạn tài nguyên

Tập tin limits.conf được đặt trong thư mục "/etc/security" có thể được dùng để điều khiển và giới hạn các tài nguyên cho những người sử dụng trên hệ thống

của bạn. Điều quan trọng là đặt giới hạn tài nguyên trên tất cả các người sử dụng của bạn để họ không thể thực hiện các cuộc tấn công "từ chối dịch vụ" (số lượng các tiến trình (process), số lượng tiền, v.v..). Các giới hạn này sẽ phải đặt cho người sử dụng khi người này log-in. Lấy ví dụ, giới hạn cho tất cả những người sử dụng trên hệ thống của bạn có thể tựa như thế này.

Bước 1

Soạn thảo tập tin limits.conf (vi /etc/security/limits.conf) và thêm vào hoặc thay đổi các dòng trở thành:

```
*    hard core    0
*    hard rss     5000
*    hard nproc   20
```

Điều này nói lên rằng ngăn cấm việc tạo các tập tin cốt yếu "core 0", hạn chế số lượng các tiến trình (process) là 20 "nproc 20", và hạn chế sử dụng bộ nhớ là 5M "rss 5000" cho mọi người ngoại trừ super user "root". Tất cả những dòng trên chỉ liên quan đến các người sử dụng đã vào trong hệ thống thông qua dấu nhắc lệnh login. Với loại chỉ tiêu này, bạn có nhiều điều khiển hơn trên các tiến trình, các tập tin cốt yếu (core), và cách dùng bộ nhớ mà những người sử dụng có thể có trên hệ thống của bạn. Dấu hoa thị "*" nghĩa là: tất cả người sử dụng đang log-in vào server.

Bước 2

Bạn cũng phải soạn thảo tập tin "/etc/pam.d/login" và thêm vào dòng sau ở cuối tập tin:

```
session    required    /lib/security/pam_limits.so
```

Sau khi thêm vào dòng trên, tập tin "/etc/pam.d/login" trông giống như thế này:

```
#%PAM-1.0
auth       required   /lib/security/pam_securetty.so
auth       required   /lib/security/pam_pwdb.so shadow nullok
auth       required   /lib/security/pam_nologin.so
account    required   /lib/security/pam_pwdb.so
password   required   /lib/security/pam_cracklib.so
password   required   /lib/security/pam_pwdb.so nullok use_authtok md5 shadow
session    required   /lib/security/pam_pwdb.so
session    required   /lib/security/pam_limits.so
#session   optional   /lib/security/pam_console.so
```

Nhiều điều khiển hơn khi lắp (mounting) một hệ thống tập tin

Bạn có thể điều khiển nhiều hơn trong việc mount của một hệ thống tập tin như các phần ngăn (partition) "/home" và "/tmp" với các tùy chọn tiện lợi như noexec, nodev, và nosuid. Việc này có thể cài đặt trong tập tin văn bản "/etc/fstab". Tập tin fstab chứa đựng các thông tin mô tả về các tùy chọn khi mount hệ thống tập tin khác nhau; mỗi dòng chỉ rõ một hệ thống tập tin.

Thông tin được liên quan tới các tùy chọn bảo mật trong tập tin văn bản fstab là:

defaults	Cho phép mọi thứ (quota,read-write và suid) trên partition này.
noquota	Không cho phép thiết lập quotas trên partition này.
nosuid	Không cho phép thiết lập việc truy cập bằng SUID/SGID trên partition này.
nodev	Không cho phép thiết lập ký tự hoặc các thiết bị truy cập đặc biệt trên partition này.
noexec	Không cho phép thực thi bất kỳ tập tin nhị phân trên partition này.
quota	Cho phép dùng quotas trên partition này.
ro	Cho phép chỉ đọc (read-only) trên partition này.
rw	Cho phép đọc-ghi (read-write) trên partition này.
suid	Cho phép SUID/SGID truy cập trên partition này.

Ghi chú: *Để biết thêm thông tin về các tùy chọn mà bạn có thể đặt trong tập tin này (fstab), xem các trang hướng dẫn (man pages) về mount(8).*

Soạn thảo tập tin fstab (vi /etc/fstab) và thay đổi nó phụ thuộc vào các yêu cầu của bạn. Lấy ví dụ :

```
/dev/sda11      /tmp     ext2 defaults 1 2
/dev/sda6       /home    ext2 defaults 1 2
```

trở thành:

```
/dev/sda11      /tmp     ext2 defaults,rw,nosuid,nodev,noexec 1 2
/dev/sda6       /home    ext2 defaults,rw,nosuid,nodev 1 2
```

Có nghĩa là <nosuid> không cho phép bit SUID và SGID có tác dụng, <nodev> không thông dịch ký tự hay khối các thiết bị đặc biệt trên partition tập tin hệ thống của bạn, và <noexec> không cho phép thi hành bất kỳ dạng nhị phân nào trên partition của tập tin hệ thống. Hãy chú ý rằng chúng ta thêm vào tùy chọn "rw" để hiệu chỉnh các dòng trên. Đây là các tùy chọn mặc nhiên cho các dòng trên. Mà nó có nghĩa là đặt lại chỉ tiêu, đọc–ghi, và suid, do vậy chúng ta phải thêm vào tùy chọn "rw" để tiếp tục đọc-ghi trên các hệ thống tập tin vừa được hiệu chỉnh.

Ghi chú: *Lấy ví dụ trên của chúng ta, thì "/dev/sda11" được gắn thành thư mục "/tmp" trên hệ thống, và "/dev/sda6" là partition của thư mục "/home". Dĩ nhiên, việc này sẽ không giống nhau mà nó phụ thuộc vào việc phân chia đĩa cứng của bạn như thế nào và loại đĩa nào được cài đặt trên hệ thống, IDE (had, hdb, v..v..) hay SCSI (sda, sdb, v.v...).*

Di chuyển dạng nhị phân RPM vào trong một nơi an toàn hoặc thay đổi quyền mặc nhiên cho nó.

Một khi bạn đã cài đặt tất cả phần mềm cần thiết trên hệ thống Linux server với lệnh RPM, đó là một ý tưởng hay cho việc bảo mật tốt hơn khi di chuyển nó đến một nơi an toàn như đĩa mềm hay một nơi an toàn nào khác mà bạn chọn. Với phương pháp này nếu ai đó truy cập tới server của bạn và có ý định cài đặt phần mềm có hại với lệnh RPM, anh ta sẽ không thể thực hiện được. Dĩ

nhiên, nếu trong tương lai bạn muốn cài đặt hay nâng cấp phần mềm mới thông qua RPM, việc bạn phải làm là di chuyển RPM trở lại thư mục gốc của nó nữa.

- Để di chuyển dạng nhị phân RPM trên đĩa mềm, dùng lệnh:

```
[root@deep /]# mount /dev/fd0 /mnt/floppy/
[root@deep /]# mv /bin/rpm /mnt/floppy/
root@deep /]# umount /mnt/floppy
```

Ghi chú: *Đừng bao giờ gỡ bỏ chương trình RPM một cách hoàn toàn khỏi hệ thống của bạn, hay bạn sẽ không thể cài đặt lại nó nữa ở lần sau. Vì rằng để cài đặt RPM hay phần mềm khác bạn cần phải có sẵn các lệnh RPM.*

Hơn nữa bạn có thể làm thay đổi quyền hạn mặc định của lệnh "rpm" từ 755 tới 700. Với sự hiệu chỉnh này, những người dùng không phải là root (non-root users) không thể dùng chương trình "rpm" để truy vấn, cài đặt v.v...; cả trong trường hợp bạn quên di chuyển nó tới một nơi an toàn sau khi cài đặt xong các chương trình mới.

- Để thay đổi quyền hạn mặc định của "/bin/rpm", dùng lệnh:

```
[root@deep /]# chmod 700 /bin/rpm
```

Shell logging

Để tạo dễ dàng cho bạn là lặp lại các câu lệnh dài, bash shell lưu trữ lên đến 500 câu lệnh cũ trong tập tin "~/.bash_history" (trong đó "~/ " là thư mục home của bạn). Mỗi người sử dụng có một tài khoản trên hệ thống sẽ có tập tin ".bash_history" trong thư mục home của họ. Việc giảm số lượng lệnh cũ mà các tập tin ".bash_history" có thể lưu giữ sẽ ngăn ngừa việc lưu giữ quá lâu mật khẩu của các người dùng trong tập tin ".bash_history" khi họ đã nhập nhầm mật khẩu của mình trong quá trình làm việc trên máy server.

Bước 1

Các dòng HISTFILESIZE và HISTSIZE trong tập tin "/etc/profile" xác định kích thước của các lệnh cũ có thể lưu giữ trong tập tin ".bash_history" cho tất cả các người sử dụng trên hệ thống của bạn. Đối với tất cả các tài khoản chúng tôi khuyên bạn là đặt các thông số HISTFILESIZE và HISTSIZE trong tập tin "/etc/profile" tới một giá trị thấp chẳng hạn như 20.

Soạn thảo tập tin profile (vi /etc/profile) và thay đổi các dòng sang:

```
HISTFILESIZE=20
HISTSIZE=20
```

Có nghĩa là, tập tin ".bash_history" trong mỗi thư mục home của các người sử dụng có thể lưu trữ 20 câu lệnh cũ và không hơn. Bây giờ, nếu kẻ bẻ khoá (cracker) cố gắng tìm xem tập tin "~/.bash_history" của những người sử dụng trên máy server của bạn để tìm một số mật khẩu được gõ sai bằng text, anh hoặc cô ta có ít cơ hội để tìm ra nó.

Bước 2

Người quản trị hệ thống cũng nên thêm vào tập tin "/etc/skel/.bash_logout" dòng "rm –f $HOME/.bash_history", để mỗi khi một người dùng logout, tập tin ".bash_history" của họ sẽ bị xóa và kẻ bẻ khoá sẽ không thể dùng tập tin ".bash_history" của những người sử dụng mà hiện tại không đăng nhập (login) vào hệ thống.

Soạn thảo tập tin **.bash_logout** (vi /etc/skel/.bash_logout) và thêm vào dòng sau:

```
rm -f $HOME/.bash_history
```

Ghi chú: *Mẹo vặt ở trên sẽ chỉ làm việc cho những người sử dụng tương lai mà bạn sẽ thêm vào trong server. Nếu bạn đã có sẵn các người sử dụng trong tập tin "/etc/passwd", bạn phải soạn thảo và thêm dòng trên vào trong những tập tin ".bash_logout" của họ bằng tay.*

Tập tin "/etc/lilo.conf"

LILO là cách chung nhất đã sử dụng trình nạp khởi động (boot loader) cho Linux. Nó quản lý quá trình khởi động và có thể khởi động các Linux kernel image từ đĩa mềm, đĩa cứng hoặc có thể thậm chí hoạt động như một "boot manager" cho các hệ điều hành khác. LILO rất quan trọng trong hệ điều hành Linux và để phục vụ cho lý do này, chúng ta phải bảo vệ nó tốt nhất nếu chúng ta có thể. Tập tin cấu hình quan trọng nhất của LILO là tập tin "lilo.conf", và nó ở trong thư mục "/etc". Với tập tin này chúng ta có thể cấu hình và cải tiến việc bảo mật chương trình LILO của chúng ta và hệ điều hành Linux. Sau đây là ba tùy chọn quan trọng sẽ cải tiến việc bảo mật chương trình LILO của chúng ta.

- Thêm : timeout=00

Tùy chọn này điều khiển chờ LILO cho việc nhập vào từ người sử dụng trước khi khởi động được chọn lựa mặc định là bao lâu (tính bằng giây). Một trong các yêu cầu của bảo mật C2 là đó là interval được đặt là 0 trừ khi hệ thống khởi động gấp đôi cái gì đó khác.

- Thêm : restricted

Tùy chọn này yêu cầu cho một mật khẩu, nếu các tham số không được chỉ rõ trên đòng lệnh. Tùy chọn restricted chỉ có thể được dùng với tùy chọn "password". Bảo đảm rằng bạn dùng điều này trên mỗi ảnh (image).

- Thêm : password=<password>

Tùy chọn này yêu cầu cho một mật khẩu khi cố gắng nạp hệ thống Linux trong chế độ "single mode". Các mật khẩu thì luôn phân biệt chữ thường và chữ in hoa, và cũng bảo đảm rằng tập tin "/etc/lilo.conf" là mọi người không thể đọc được nữa, hoặc bất kỳ người dùng nào sẽ có thể đọc mật mã. Đây là một ví dụ LILO được bảo vệ với tập tin "lilo.conf".

Bước 1

Soạn thảo tập tin lilo.conf (vi /etc/lilo.conf) và thêm hay thay đổi các tùy chọn trên như được trình bày bên dưới:

```
boot=/dev/sda
map=/boot/map
install=/boot/boot.b
prompt
timeout=00 ← Giá trị timeout được thay đổi với 00.
Default=linux
restricted ← Thêm dòng này
password=<password> ← Thêm dòng này và đặt vào mật khẩu.
image=/boot/vmlinuz-2.2.12-20
label=linux
initrd=/boot/initrd-2.2.12-10.img
root=/dev/sda6
read-only
```

Bước 2

Bởi vì tập tin cấu hình "/etc/lilo.conf" bây giờ chứa đựng các mật khẩu chưa được mã hoá, nó chỉ nên được đọc bởi super-user "root".

```
[root@deep /]# chmod 600 /etc/lilo.conf
```

Bước 3

Bây giờ chúng ta phải cập nhật tập tin cấu hình của chúng ta "/etc/lilo.conf" cho việc thay đổi có tác dụng.

```
[root@deep /]# /sbin/lilo -v (to update the lilo.conf file).
```

Bước 4

Một tiêu chuẩn bảo mật tốt hơn là bạn có thể bảo mật tập tin "lilo.conf" bằng cách đặt nó không thể thay đổi, dùng lệnh chattr.

- Để đặt tập tin không thể thay đổi đơn giản là dùng lệnh:

```
[root@deep /]# chattr +i /etc/lilo.conf
```

Và việc này sẽ ngăn chặn bất kỳ những thay đổi (bất ngờ hay khác) tới tập tin "lilo.conf". Nếu bạn muốn hiệu chỉnh tập tin "lilo.conf" thì bạn sẽ cần không đặt cờ hiệu không thay đổi:

- Để không đặt cờ hiệu không thay đổi , dùng lệnh:

```
[root@deep /]# chattr -i /etc/lilo.conf
```

Vô hiệu hoá lệnh shutdown bởi tổ hợp phím Control-Alt-Delete.

Chú thích một dòng (với một dấu "#") được liệt kê bên dưới trong tập tin "/etc/inittab" sẽ vô hiệu hóa việc sử dụng tổ hợp phím Ctrl-Alt-Del –lệnh shutdown

máy tính của bạn. Việc này rất quan trọng nếu bạn không có cách bảo mật vật lý tốt nhất trên Linux của bạn.

Để thực hiện điều này, soạn thảo tập tin inittab (vi /etc/inittab) và thay đổi dòng:

```
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

Trở thành:

```
#ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

Bây giờ, để thay đổi có tác dụng gõ vào dấu nhắc lệnh như sau:

```
[root@deep /]# /sbin/init q
```

Sao chép an toàn các logfile quan trọng

Một trong những vấn đề bảo mật đáng quan trọng nhất là tính toàn vẹn của các tập tin log khác nhau trong thư mục "/var/log" trên server của bạn. Mặc dù mỗi chức năng bảo mật đặt trong một nơi trên server của bạn, một kẻ bẻ khoá có thể lấy được quyền truy cập đến nó, sự che chở cuối cùng của chúng ta chính là hệ thống tập tin log, do vậy điều quan trọng là nghiên cứu một phương pháp bảo đảm tính nguyên vẹn của các tập tin log.

Nếu bạn đã cài đặt một máy in trên server của bạn, hoặc trên một máy trên mạng của bạn, một ý kiến hay là phải sao chép tất cả các log quan trọng. Việc này có thể dễ dàng hoàn thành bằng cách dùng một máy in liên tiến (continuous feed printer) và có chương trình syslog gửi tất cả các log mà bạn thấy quan trọng ra "/dev/lp0" (thiết bị máy in). Người bẻ khóa có thể thay đổi các tập tin, các chương trình, v.v..., trên server của bạn, nhưng không thể làm được gì khi bạn có một máy in đã in ra giấy tất cả các log quan trọng.

Ví dụ như:

Đối với logfile của tất cả telnet, mail, các thông điệp khởi động và các kết nối ssh từ server của bạn tới máy in được gắn vào với server này, bạn sẽ cần thêm dòng sau vào tập tin "/etc/syslog.conf":

Soạn thảo tập tin syslog.conf (vi /etc/syslog.conf) và thêm vào cuối tập tin này dòng sau:

```
authpriv.*;mail.*;local7.*;auth.*;daemon.info /dev/lp0
```

Bây giờ khởi động lại syslog daemon cho việc thay đổi có tác dụng:

```
[root@deep /]# /etc/rc.d/init.d/syslog restart
```

Ví dụ như:

Đối với log của tất cả telnet, mail, các thông điệp khởi động và các kết nối ssh từ server của bạn tới máy in được gắn vào với server từ xa trong mạng cục bộ của bạn, khi đó bạn sẽ cần thêm dòng sau vào tập tin "/etc/syslog.conf" trên máy server từ xa.

(Nếu bạn không có một máy in trong mạng của bạn, bạn có thể cũng sao chép tất cả các tập tin log tới máy khác; đơn giản bỏ qua bước đầu tiên bên dưới của việc thêm "/dev/lp0" tới tập tin "syslog.conf" của bạn trên máy từ xa và đi thẳng tới bước tùy chọn "-r" từ xa. Sử dụng đặc trưng của việc sao chép tất cả các tập tin log tới máy khác sẽ cung cấp cho bạn có thể để điều khiển tất cả các thông điệp syslog trên một host và sẽ xuất ra các yêu cầu của việc quản trị).

Soạn thảo tập tin syslog.conf (vi /etc/syslog.conf) trên máy server từ xa (lấy ví dụ: mail.openna.com) và thêm vào cuối tập tin này dòng như sau:

```
authpriv.*;mail.*;local7.*;auth.*;daemon.info /dev/lp0
```

Từ việc cấu hình mặc nhiên của syslog daemon là không nhận bất kỳ thông điệp nào từ mạng, chúng ta phải cho phép trên máy server từ xa dễ dàng nhận các thông điệp từ mạng. Để cho phép dễ dàng nhận các thông điệp từ mạng trên máy server từ xa, thêm tùy chọn "-r" sau vào tập tin script syslog daemon của bạn (chỉ trên host từ xa):

- Soạn thảo syslog daemon (vi +24 /etc/rc.d/init.d/syslog) và thay đổi:

  ```
  daemon syslogd -m 0
  ```

  Trở thành:

  ```
  daemon syslogd -r -m 0
  ```

- Bây giờ khởi động lại syslog daemon của bạn trên host từ xa cho việc thay đổi có tác dụng:

  ```
  [root@mail /]# /etc/rc.d/init.d/syslog restart
  ```

Bây giờ, nếu chúng ta có một firewall trên một server từ xa (bạn được hỗ trợ để có nó), chúng ta phải thêm hay kiểm tra sự tồn tại của các dòng sau:

```
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-s $SYSLOG_CLIENT \
-d $IPADDR 514 -j ACCEPT
```

Trong đó EXTERNAL_INTERFACE ="eth0" trong tập tin firewall.

Trong đó IPADDR ="208.164.186.2" trong tập tin firewall.

Trong đó SYSLOG_CLIENT ="208.164.168.0/24" trong tập tin firewall.

- Bây giờ khởi động lại syslog daemon của bạn trên host từ xa cho việc thay đổi có tác dụng:

  ```
  [root@mail /]# /etc/rc.d/init.d/firewall restart
  ```

Nguyên tắc firewall sẽ cho phép các packet UDP đến trên port 514 (syslog port) trên máy server từ xa mà đến từ máy trạm trong mạng nội bộ thì được chấp nhận. Để biết thêm thông tin về firewall xem chương 7 "Networking firewall".

Cuối cùng, soạn thảo tập tin syslog.conf (vi /etc/syslog.conf) trên server cục bộ, và thêm vào cuối tập tin dòng sau:

```
authpriv.*;mail.*;local7.*;auth.*;daemon.info @mail
```

Trong đó "mail" là hostname của server từ xa.

- Bây giờ khởi động lại syslog daemon của bạn để việc thay đổi có tác dụng:

```
[root@mail /]# /etc/rc.d/init.d/firewall restart
```

giống như trên host từ xa, chúng ta phải thêm hoặc kiểm tra sự tồn tại của các dòng sau trong tập tin script trên host cục bộ:

```
ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
-s $IPADDR 514 \
-d $SYSLOG_SERVER 514 -j ACCEPT
```

Trong đó EXTERNAL_INTERFACE ="eth0" trong tập tin firewall.

Trong đó IPADDR ="208.164.186.1 trong tập tin firewall.

Trong đó SYSLOG_CLIENT ="mail.openna.com" trong tập tin firewall.

- Bây giờ khởi động lại firewall của bạn để việc thay đổi có tác dụng:

```
[root@deep /]# /etc/rc.d/init.d/firewall restart
```

Nguyên tắc firewall sẽ cho phép các packet UDP ra khỏi mạng trên port 514 (syslog port) trên máy server cục bộ được dành riêng cho syslog server từ xa thì được chấp nhận. Để biết thêm thông tin về firewall xem chương 7 "Networking firewall".

Ghi chú: *Đừng bao giờ dùng Gateway Server như một host để điều khiển tất cả các thông điệp syslog; đấy là một ý tưởng rất dở. Có nhiều tùy chọn và chiến lược hơn tồn tại với chương trình sysklogd, xem các trang man về sysklogd(8), syslog(2), và syslog.conf(5) để biết thêm thông tin chi tiết.*

Đặt các quyền sử dụng hệ thống trên thư mục "/etc/rc.d/init.d" cho các tập tin dạng kịch bản.

Đặt các quyền sử dụng hệ thống của các tập tin dạng kịch bản (script) mà nó có nhiệm vụ khởi động (start) và dừng (stop) tất cả các tiến trình bình thường của bạn mà cần để chạy lúc khởi động.

```
[root@deep /]# chmod -R 700 /etc/rc.d/init.d/*
```

Điều này có nghĩa là chỉ có root là được phép đọc (Read), ghi (Write), và thi hành (Execute) các tập tin script trong thư mục này. Chúng tôi không nghĩ rằng những người sử dụng hợp lệ thì cần biết các tập tin script đó chứa nội dung gì.

Ghi chú: *Nếu bạn cài đặt một chương trình mới hay cập nhật một chương trình mà nó dùng khởi tạo script hệ thống V định vị trong thư mục "/etc/rc.d/init.d", đừng quên thay đổi hay kiểm tra quyền hạn của tập tin script lần nữa.*

Tập tin "/etc/rc.d/rc.local"

Mặc nhiên, khi bạn login vào Linux, nó sẽ báo cho bạn biết tên phân phối Linux, phiên bản, phiên bản hạt nhân của hệ điều hành (kernel version), và tên của máy server. Đây là việc cung cấp rất nhiều thông tin. Chúng tôi chỉ muốn nhắc những người sử dụng với dấu nhắc lệnh "Login:".

Bước 1

Để làm việc này, soạn thảo tập tin "/etc/rc.d/rc.local" và đặt "#" ở phía trước các dòng được trình bày sau:

```
--
# This will overwrite /etc/issue at every boot. So, make any changes you
# want to make to /etc/issue here or you will lose them when you reboot.
#echo "" > /etc/issue
#echo "$R" >> /etc/issue
#echo "Kernel $(uname -r) on $a $(uname -m)" >> /etc/issue
#
#cp -f /etc/issue /etc/issue.net
#echo >> /etc/issue
--
```

Bước 2

Sau đó, xoá các tập tin sau: "issue.net" và "issue" trong thư mục "/etc":

```
[root@deep /]# rm -f /etc/issue
[root@deep /]# rm -f /etc/issue.net
```

Ghi chú: *Tập tin "/etc/issue.net" là biểu ngữ login mà các người sử dụng sẽ thấy khi họ tạo một liên kết hệ thống mạng (nghĩa là telnet, SSH) với máy của bạn. Bạn sẽ tìm nó trong thư mục "/etc", theo sau với tập tin tương tự được gọi là "issue", mà nó là biểu ngữ login được hiển thị tới các người sử dụng cục bộ. Nó đơn giản chỉ là một tập tin văn bản và có thể được tạo theo ý thích của riêng bạn, nhưng cũng nhận biết các ghi chú bên trên, nếu bạn thay đổi nó hay xóa nó như chúng làm, thì bạn cũng sẽ cần phải hiệu chỉnh tập tin shell script "/etc/rc.d/rc.local", điều cần làm là tạo lại cả hai tập tin "issue" và "issue.net" mỗi lần hệ thống khởi động.*

Các Bits từ các chương trình được sở hữu riêng của root.

Một người dùng hợp lệ sẽ có thể chạy một chương trình như root nếu được đặt là SUID root. Tất cả các chương trình và các tập tin trên máy tính của bạn với bit 's' xuất hiện trong mode của nó, có SUID (-rwsr-xr-x) hay SGID(-r-xr-sr-x) bit được bật. Bởi vì các chương trình này cấp các quyền đặc biệt cho người sử dụng là người sẽ thi hành chúng, Việc quan trọng là xóa bit 's' từ các chương trình được sở hữu riêng của root mà sẽ không yêu cầu tuyệt đối quyền hạn nào đó. Điều này có thể hoàn thành bằng cách thi hành lệnh **'chmod a-s'** với tên của các tập tin SUID/SGID như các thông số của nó.

Rất nhiều các chương trình sau, nhưng không được giới hạn với:

- Các chương trình mà bạn không bao giờ dùng.

- Các chương trình mà bạn không cần bất kỳ người sử dụng không là root để chạy.
- Các chương trình mà bạn dùng thỉnh thoảng, và không phải chú ý dùng lệnh su để trở thành root khi chạy.

Chúng ta đặt một dấu sao (*) tiếp theo mỗi chương trình mà theo ý riêng, chúng ta phải vô hiệu hóa vì chúng không thật cần thiết cho công việc của server chúng ta. Nên nhớ rằng hệ thống của bạn cần một số chương trình suid root để làm việc đúng đắn, vì vậy nên cẩn thận.

- Để tìm tất cả các tập tin với bit 's' từ các chương trình được sở hữu riêng của root, dùng lệnh:

```
[root@deep]# find / -type f \( -perm -04000 -o -perm -02000 \) \-exec ls –lg {} \;
*-rwsr-xr-x    1 root     root        35168  Sep 22 23:35  /usr/bin/chage
*-rwsr-xr-x    1 root     root        36756  Sep 22 23:35  /usr/bin/gpasswd
*-r-xr-sr-x    1 root     tty          6788  Sep 6 18:17   /usr/bin/wall
-rwsr-xr-x 1   root       root        33152  Aug 16 16:35   /usr/bin/at
-rwxr-sr-x 1   root       man         34656  Sep 13 20:26   /usr/bin/man
-r-s--x--x 1   root       root        22312  Sep 25 11:52   /usr/bin/passwd
-rws--x--x 2   root       root       518140  Aug 30 23:12   /usr/bin/suidperl
-rws--x--x 2   root       root       518140  Aug 30 23:12   /usr/bin/sperl5.00503
-rwxr-sr-x 1   root       slocate     24744  Sep 20 10:29   /usr/bin/slocate
*-rws--x--x    1 root     root        14024  Sep 9 01:01    /usr/bin/chfn
*-rws--x--x 1  root       root        13768  Sep 9 01:01    /usr/bin/chsh
*-rws--x--x 1  root       root         5576  Sep 9 01:01    /usr/bin/newgrp
*-rwxr-sr-x 1  root       tty          8328  Sep 9 01:01    /usr/bin/write
-rwsr-xr-x 1   root       root        21816  Sep 10 16:03   /usr/bin/crontab
*-rwsr-xr-x    1 root     root         5896  Nov 23 21:59   /usr/sbin/usernetctl
*-rwsr-xr-x    1 root     bin         16488  Jul 2 10:21    /usr/sbin/traceroute
-rwxr-sr-x 1   root       utmp         6096  Sep 13 20:11   /usr/sbin/utempter
-rwsr-xr-x 1   root       root        14124  Aug 17 22:31   /bin/su
*-rwsr-xr-x    1 root     root        53620  Sep 13 20:26   /bin/mount
*-rwsr-xr-x    1 root     root        26700  Sep 13 20:26   /bin/umount
*-rwsr-xr-x    1 root     root        18228  Sep 10 16:04   /bin/ping
*-rwxr-sr-x    1 root     root         3860  Nov 23 21:59   /sbin/netreport
-r-sr-xr-x     1 root     root        26309  Oct 11 20:48   /sbin/pwdb_chkpwd
```

- Để vô hiệu hóa các bits suid trên các chương trình được chọn ở trên, gõ các lệnh sau:

```
[root@deep /]# chmod a-s /usr/bin/chage
[root@deep /]# chmod a-s /usr/bin/gpasswd
[root@deep /]# chmod a-s /usr/bin/wall
[root@deep /]# chmod a-s /usr/bin/chfn
[root@deep /]# chmod a-s /usr/bin/chsh
```

```
[root@deep /]# chmod a-s /usr/bin/newgrp
[root@deep /]# chmod a-s /usr/bin/write
[root@deep /]# chmod a-s /usr/sbin/usernetctl
[root@deep /]# chmod a-s /usr/sbin/traceroute
[root@deep /]# chmod a-s /bin/mount
[root@deep /]# chmod a-s /bin/umount
[root@deep /]# chmod a-s /bin/ping
[root@deep /]# chmod a-s /sbin/netreport
```

Nếu bạn muốn biết những chương trình đó làm gì, thì gõ "man program-name" và đọc trang hướng dẫn.

Ví dụ như:

```
[root@deep /]# man netreport
```

Các thông số kernel tinh chỉnh được.

Với phiên bản Red Hat Linux 6.2 tất cả các thông số kernel là có sẵn trong thư mục con "/proc/sys"của Linux có thể được cấu hình lúc thi hành. Bây giờ bạn có thể dùng tập tin mới "/etc/sysctl.conf" trong Red Hat Linux 6.2 để hiệu chỉnh và đặt các thông số kernel lúc thi hành. Tập tin "sysctl.conf" được đọc và nạp mỗi lần khi hệ thống khởi động. Tất cả các thông số cài đặt giờ đây được lưu trữ trong tập tin "/etc/sysctl.conf". Tất cả các hiệu chỉnh tới "/proc/sys" nên làm thông qua "/etc/sysctl.conf", bởi vì chúng là tốt hơn cho việc điều khiển, và được thi hành trước "rc.local' hay bất kỳ script của người sử dụng nào khác. Dưới đây, chúng tôi trình bày cho bạn về các tùy chọn bảo mật mạng mà bạn phải cấu hình trên máy server của bạn trên Red Hat Linux 6.1 và 6.2.

Ngăn chặn hệ thống của bạn từ việc đáp lại yêu cầu ping.

Ngăn chặn hệ thống của bạn từ việc đáp lại yêu cầu ping có thể là một cải tiến lớn trong việc bảo mật hệ thống mạng từ khi không có ai có thể ping trên máy server của bạn và nhận được một sự trả lời. Bộ giao thức TCP/IP có một số điểm yếu mà nó cho phép một kẻ tấn công sử dụng kỹ thuật hướng những gói dữ liệu gian lận xuyên qua cùng rất nhiều gói dữ liệu hợp lệ khác. Việc ngăn chặn server của bạn trả lời các yêu cầu ping có thể giúp để giảm bớt các lỗi này.

Trên Red Hat Linux 6.1

Một lệnh ...

```
[root@deep /]# echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_all
```

... nên thực hiện công việc này ví dụ như hệ thống của bạn không đáp lại lệnh ping trên bất kỳ giao tiếp nào. Bạn có thể thêm vào dòng này trong tập tin "/etc/rc.d/rc.local" của bạn như vậy lệnh sẽ được tự động thiết lập nếu hệ thống của bạn khởi động lại. Không đáp lại lệnh ping sẽ ít nhất giữ không cho "những kẻ tấn công" vào bởi vì không bao giờ biết nó ở đó.

Để quay trở về nó, đơn giản là:

```
[root@deep /]# echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_all
```

Bạn có thể thêm vào dòng này trong tập tin "/etc/rc.d/rc/local" do vậy lệnh sẽ được đặt một cách tự động khi hệ thống của bạn khởi động lại.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Enable ignoring ping request
net.ipv4.icmp_echo_ignore_all = 1
```

Bạn phải khởi động lại hệ thống mạng của bạn cho việc thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                 [ OK ]
Bringing up interface lo                   [ OK ]
Bringing up interface eth0                 [ OK ]
Bringing up interface eth1                 [ OK ]
```

Ngăn chặn hệ thống của bạn đáp lại yêu cầu broadcasts

Cũng giống như yêu cầu ping, nó cũng quan trọng để vô hiệu hóa yêu cầu broadcast. Khi mà một gói tin (packet) được gửi tới một địa chỉ broadcast IP (nghĩa là 192.168.1.255) từ một máy trên mạng cục bộ, mà packet được chuyển giao tới tất cả các máy trên mạng đó. Khi đó tất cả các máy trên mạng đáp lại yêu cầu ICMP này và kết quả có thể phục vụ tắc nghẽn hay ngừng lại của mạng (sự từ chối dịch vụ của những kẻ tấn công). Xem RFC 2644 để biết thêm thông tin.

Trên Red Hat Linux 6.1

```
[root@deep /]# echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
```

Bạn có thể thêm dòng này vào trong tập tin "/etc/rc.d/rc.local" vì vậy lệnh sẽ được đặt tự động khi hệ thống của bạn khởi động lại.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Enable ignoring ping request
net.ipv4.icmp_echo_ignore_all = 1
```

Bạn phải khởi động lại hệ thống mạng của bạn cho việc thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                 [ OK ]
```

```
Bringing up interface lo                [ OK ]
Bringing up interface eth0              [ OK ]
Bringing up interface eth1              [ OK ]
```

Routing Protocols

Routing hay routing protocols có thể tạo ra một vài vấn đề. Routing IP nguồn, nơi mà một IP packet chứa đựng chi tiết của đường dẫn tới một đích đến dự định của nó, là nguy hiểm bởi vì thông qua RFC 1122 host đích phải đáp lại với một đường dẫn tương tự. Nếu một kẻ tấn công có thể gửi một routed packet nguồn vào trong mạng của bạn, thì khi đó anh ta sẽ có thể chặn đứng việc trả lời và đánh lừa host của bạn nghĩ rằng nó đang kết nối với một host thật sự. Điều khuyên là bạn nên vô hiệu hóa các địa chỉ routing IP nguồn để bảo vệ máy server của bạn khỏi lỗ hổng này.

Để vô hiệu hoá IP routing nguồn trên server của bạn, gõ lệnh sau trong máy trạm của bạn (terminal):

Trên Red Hat Linux 6.1

```
[root@deep /]# for f in /proc/sys/net/ipv4/conf/*/accept_source_route; do
> echo 0 > $f
> done
[root@deep /]#
```

Thêm vào các lệnh trên vào tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó nữa ở lần kế tiếp bạn khởi động lại hệ thống

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Disables IP source routing
net.ipv4.conf.all.accept_source_route = 0
```

Bạn phải khởi động lại hệ thống mạng của bạn cho việc thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters              [ OK ]
Bringing up interface lo                [ OK ]
Bringing up interface eth0              [ OK ]
Bringing up interface eth1              [ OK ]
```

Ghi chú: *Có một ghi chú là lệnh bên trên cho Red Hat Linux 6.1 hoặc 6.2 sẽ vô hiệu hóa Source Routed Packets trên tất cả các giao tiếp của bạn (lo, ethN, pppN , v.v...).*

Hiệu lực hóa chế độ bảo vệ TCP Syncookie

Một "SYN Attack" là một kiểu tấn công từ chối dịch vụ (DoS = Denial of Service) làm tiêu tốn tất cả các tài nguyên trên máy của bạn, buộc bạn khởi động lại. Tấn công từ chối các dịch vụ (kiểu tấn công mà nó làm mất hết khả năng một server do bởi số lượng lưu thông quá cao hay cái gì đó mà tài nguyên của hệ thống server không thể đáp lại rất nhiều yêu cầu kết nối hợp lệ từ một hệ thống từ xa) thì dễ dàng được thực hiện từ các kết nối nội bộ bên trong hay bên ngoài thông qua extranet và internet. Để cho chế độ bảo vệ có hiệu lực (enable), bạn phải làm:

Trên Red Hat Linux 6.1

```
[root@deep /]# echo 1 > /proc/sys/net/ipv4/tcp_syncookies
```

Thêm vào các lệnh trên vào tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó nữa ở lần kế tiếp bạn khởi động lại hệ thống

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
net.ipv4.tcp_syncookies = 1
```

Bạn phải khởi động lại hệ thống mạng của bạn cho việc thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                [ OK ]
Bringing up interface lo                  [ OK ]
Bringing up interface eth0                [ OK ]
Bringing up interface eth1                [ OK ]
```

Ghi chú: *Nếu bạn nhận một thông điệp lỗi trong khi thi hành lệnh trên, kiểm tra rằng bạn enable tùy chọn TCP syncookie trong cấu hình kernel của bạn: IP: TCP syncookie support (chưa được enable mặc nhiên) (CONFIG_SYN_COOKIES) [Y/n/?].*

Làm mất tác dụng sự chấp nhận chuyển hướng ICMP (Diable ICMP Redirect Acceptance)

Khi các host sử dụng chưa tối ưu hay lộ trình không thể nào để tới một đích đến riêng biệt, một ICMP redirect packet thường được dùng bởi các router để báo cho các host biết lộ trình đi đúng của một package là gì. Nếu một kẻ tấn công có thể giả mạo ICMP redirect packets, anh ta hay cô ta có thể thay đổi các bảng route trên host và có thể làm phá vỡ việc bảo mật host bởi nguyên nhân vận chuyển tới luồng thông qua một đường dẫn mà bạn không dự định. Điều đáng được khuyên là bạn phải disable ICMP Redirect Acceptance để bảo vệ máy server của bạn từ lỗ hổng này.

Trên Red Hat Linux 6.1

```
[root@deep /]# for f in /proc/sys/net/ipv4/conf/*/accept_redirects; do
> echo 0 > $f
> done
[root@deep /]#
```

Thêm vào các lệnh trên vào tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó nữa ở lần kế tiếp bạn khởi động lại hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Disable ICMP Redirect Acceptance
net.ipv4.conf.all.accept_redirects = 0
```

Bạn phải khởi động lại hệ thống mạng của bạn cho việc thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                [ OK ]
Bringing up interface lo                  [ OK ]
Bringing up interface eth0                [ OK ]
Bringing up interface eth1                [ OK ]
```

Ghi chú: *Có một ghi chú là lệnh bên trên cho Red Hat Linux 6.1 hoặc 6.2 sẽ diable Redirect Acceptance Packets trên tất cả các giao tiếp của bạn (lo, ethN, pppN , v.v...).*

Làm cho có hiệu lực sự bảo vệ always-defragging

Việc bảo vệ này phải được cho phép (enable) nếu bạn dùng Linux server của bạn như một gateway để đánh lừa việc lưu thông bên trong tới internet (IP Masquerading).

Trên Red Hat Linux 6.1

```
[root@deep /]# echo 1 > /proc/sys/net/ipv4/ip_always_defrag
```

Thêm vào các lệnh trên vào tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó nữa ở lần kế tiếp bạn khởi động lại hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Enable always defragging Protection
net.ipv4.ip_always_defrag = 1
```

Bạn phải khởi động lại hệ thống mạng của bạn cho việc thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                [ OK ]
Bringing up interface lo                  [ OK ]
Bringing up interface eth0                [ OK ]
Bringing up interface eth1                [ OK ]
```

Làm cho có hiệu lực sự bảo vệ bad error message

Tùy chọn này sẽ báo cho bạn biết tất cả các thông điệp lỗi hỏng trong hệ thống mạng của bạn.

Trên Red Hat Linux 6.1

```
[root@deep /]# echo 1 >
/proc/sys/net/ipv4/icmp_ignore_bogus_error_responses
```

Thêm vào các lệnh trên vào tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó nữa ở lần kế tiếp bạn khởi động lại hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Enable bad error message Protection
net.ipv4.icmp_ignore_bogus_error_responses = 1
```

Bạn phải khởi động lại hệ thống mạng của bạn cho việc thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                [ OK ]
Bringing up interface lo                  [ OK ]
Bringing up interface eth0                [ OK ]
Bringing up interface eth1                [ OK ]
```

Làm cho có hiệu lực sự bảo vệ IP spoofing

Spoofing protection ngăn chặn hệ thống mạng tránh việc trao đổi thông tin bị đáng lừa mà thường được dùng trong DoS Attacks.

Trên Red Hat Linux 6.1

```
[root@deep /]# for f in /proc/sys/net/ipv4/conf/*/rp_filter; do
> echo 1 > $f
> done
[root@deep /]#
```

Thêm vào các lệnh trên vào tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó nữa ở lần kế tiếp bạn khởi động lại hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Enable IP spoofing protection, turn on Source Address Verification
net.ipv4.conf.all.rp_filter = 1
```

Bạn phải khởi động lại hệ thống mạng của bạn cho việc thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                 [ OK ]
Bringing up interface lo                   [ OK ]
Bringing up interface eth0                 [ OK ]
Bringing up interface eth1                 [ OK ]
```

Logfile cho Spoofed Packets, Source Routed Packets, Redirect Packets

Việc bảo vệ này sẽ log tất cả Spoofed Packets, Source Routed Packets, Redirect Packets tới các tập tin log của bạn.

Trên Red Hat Linux 6.1

```
[root@deep /]# for f in /proc/sys/net/ipv4/conf/*/log_martians; do
> echo 1 > $f
> done
[root@deep /]#
```

Thêm vào các lệnh trên vào tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó nữa ở lần kế tiếp bạn khởi động lại hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Log Spoofed Packets, Source Routed Packets, Redirect Packets
net.ipv4.conf.all.log_martians = 1
```

Bạn phải khởi động lại hệ thống mạng của bạn cho việc thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                 [ OK ]
Bringing up interface lo                   [ OK ]
```

```
Bringing up interface eth0                [ OK ]
Bringing up interface eth1                [ OK ]
```

Các tập tin khác thường hoặc ẩn (Unusual or hidden files)

Điều quan trọng không quên là nhìn xem khắp nơi trong hệ thống của bạn có các tập tin ẩn hay bất thường (các tập tin mà bắt đầu với một dấu chấm và chúng sẽ không được hiển thị bình thường bởi lệnh "ls"), và chúng có thể được dùng như các công cụ (tool) để lấy thông tin quan trọng trên máy của bạn (các chương trình bẻ khóa mật mã, các tập tin mật khẩu từ các hệ thống khác, v.v...). Một kỹ thuật phổ biến trên các hệ điều hành Unix là phải đặt một thư mục hay tập tin ẩn trong một tài khoản (account) của người sử dụng với một tên không dùng đến, một cái gì đó tựa '...' hay '.. ' (dot dot space) hay '..^G' (dot dot control-G). Chương trình "find" có thể được dùng tìm kiếm các tập tin ẩn.

Như một ví dụ:

```
[root@deep /]# find / -name ".. " -print -xdev
[root@deep /]# find / -name ".*" -print -xdev | cat –v
```

Ghi chú: *Các tập tin với các tên chẳng hạn như '.xx' và '.mail' sẽ được dùng (đó là các tập tin phải xuất hiện bình thường).*

Tìm tất cả các tập tin với các bit SUID/SGID được bật.

Tất cả các tập tin SUID hoặc SGID vẫn tồn tại trên hệ thống của bạn sau khi chúng ta đã xóa chúng mà những tập tin này hoàn toàn không yêu cầu nhiều đặc quyền là một rủi ro bảo mật tiềm năng, và nên được theo dõi kỹ càng.

Bởi vì các chương trình này gán các quyền hạn đặc biệt cho người sử dụng là người thi hành chúng, điều đó cần thiết để bảo đảm rằng các chương trình không an toàn thì không được cài đặt.

Một mưu mẹo được ưa thích của những kẻ bẻ khoá là khai thác các chương trình SUID "root", và để lại một chương trình SUID như cửa sau để vào lại trong ở lần kế tiếp. Tìm tất cả các chương trình SUID và SGID trên hệ thống của bạn, và theo dõi xem chúng là những gì vậy, khi đó bạn sẽ biết được bất kỳ thay đổi nào, và có thể chỉ rõ một kẻ xâm phạm hệ thống tiềm ẩn.

- Dùng lệnh sau để tìm tất cả các chương trình SUID/SGID trên hệ thống của bạn:

```
[root@deep /]# find / -type f \( -perm -04000 -o -perm -02000 \) \-exec ls -lg {} \;
```

Ghi chú: *Xem trong quyển sách này trong chương 10 "Securities Software (Monitoring Tools)" để biết thêm chi tiết về phần mềm sXid mà sẽ thực hiện một công việc cho bạn một cách tự động mỗi ngày và báo cáo các kết quả thông qua mail.*

Tìm các tập tin và thư mục nhóm và có thể ghi

Các tập tin nhóm và có thể ghi cũng như các tập tin hệ thống thư mục riêng có thể là một lỗ hổng bảo mật nếu một kẻ bẻ khóa chiếm được quyền truy cập tới

hệ thống của bạn và hiệu chỉnh chúng. Hơn nữa các thư mục dùng chung có thể ghi là rất nguy hiểm, khi mà chúng cho phép các kẻ bẻ khóa tùy ý thêm hay xóa các tập tin trong các thư mục này. Trong các thao tác của các công việc bình thường, nhiều tập tin sẽ có thể được ghi, bao gồm một số từ các thư mục "/dev", "/var/catman", và tất cả các liên kết biểu tượng trên hệ thống của bạn.

- Để định vị tất cả tập tin nhóm và có thể ghi trên hệ thống của bạn, dùng lệnh:

```
[root@deep /]# find / -type f \( -perm -2 -o -perm -20 \) -exec ls -lg {} \;
```

- Để định vị tất cả thư mục nhóm và có thể ghi trên hệ thống của bạn, dùng lệnh:

```
[root@deep /]# find / -type d \( -perm -2 -o -perm -20 \) -exec ls -ldg {} \;
```

Ghi chú: *Một trình kiểm tra tính nguyên vẹn của tập tin hay thư mục tựa như phần mềm Tripwire cũng có thể được dùng một cách hợp lý để quét, quản lý và tìm một cách dễ dàng các tập tin và thư mục nhóm và có thể ghi. Xem trong quyển sách này trong chương 10 "Công cụ giám sát phần mềm bảo mật (Securities Software Monitoring Tools)" để biết thêm thông tin về Tripwire.*

Các tập tin không thuộc sở hữu riêng (Unwoned files)

Không cho phép bất kỳ các tập tin không thuộc sở hữu riêng. Các tập tin đó cũng có thể có một dấu hiệu là một người xâm phạm hệ thống đã truy cập tới hệ thống. Nếu bạn tìm các tập tin hay thư mục sở hữu trên hệ thống của bạn, kiểm tra tính toàn vẹn của nó, và nếu tất cả đều tốt thì cho chúng một tên riêng. Đôi khi bạn có thể gỡ bỏ một chương trình và lấy một tập tin hay thư mục không được sở hữu riêng liên quan tới phần mềm này; trong trường hợp này bạn có thể xoá tập tin hay thư mục đó một cách an toàn.

- Để xoá các tập tin tên hệ thống của bạn mà không có thuộc sở hữu riêng sử dụng lệnh sau:

```
[root@deep /]# find / -nouser -o -nogroup
```

Ghi chú: *Một lần nữa, các báo cáo về các tập tin trong thư mục "/dev" không được tính.*

Tìm các tập tin ".rhost"

Tìm tất cả các tập tin ".rhost" có thể tồn tại trên máy server của bạn nên là một phần trách nhiệm của người quản trị hệ thống hợp lệ của bạn, các tập tin này không nên được phép trên hệ thống của bạn. Hãy nhớ rằng một cracker chỉ cần một tài khoản không bảo mật để chiếm quyền truy cập tiềm ẩn tới hệ thống mạng của bạn.

- Bạn có thể định vị tất cả các tập tin ".rhost' trên hệ thống của bạn với lệnh sau:

```
[root@deep /]# find /home -name .rhosts
```

Bạn cũng có thể dùng một cron job để kiểm tra một cách định kỳ, báo cáo nội dung, và xoá tập tin $HOME/.rhosts. Cũng vậy, các người sử dụng nên nhận

biết rằng bạn thực hiện kiểu soạn thảo này một cách hợp lý, cũng như được trực tiếp bởi chính sách.

- Để dùng một cron job một cách định kỳ kiểm tra và báo cáo thông qua mail tất cả các tập tin ".rhosts", thực hiện lệnh sau:

Tạo tập tin script **"find_rhosts_files"** như "root" trong thư mục "/etc/cron.daily" (touch /etc/cron.daily/find_rhosts_files) và thêm vào các dòng sau trong tập tin script:

```
#!/bin/sh
/usr/bin/find /home -name .rhosts | (cat <<EOF
This is an automated report of possible existent ".rhosts" files on the server
deep.openna.com, generated by the find utility command.
New detected ".rhosts" files under the "/home" directory include:
EOF
cat
) | /bin/mail -s "Content of .rhosts file audit report" root
```

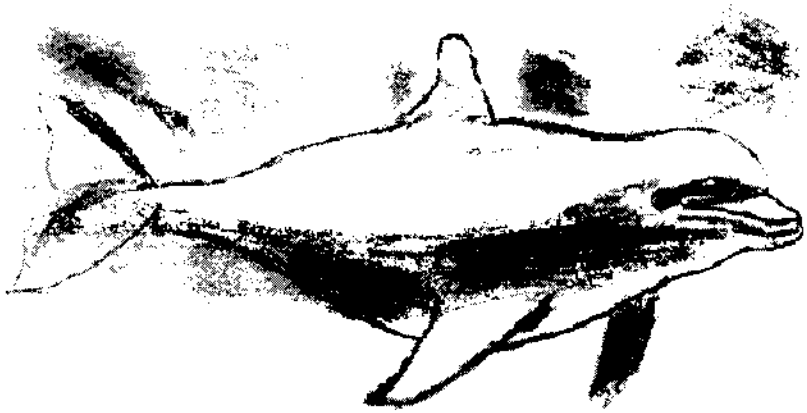
Bây giờ thi hành tập tin script này, kiểm tra tính sở hữu riêng, và thay đổi nhóm sang "root".

```
[root@deep /]# chmod 755 /etc/cron.daily/find_rhosts_files
[root@deep /]# chown 0.0 /etc/cron.daily/find_rhosts_files
```

Mỗi ngày mail sẽ được gửi tới "root" với một chủ đề : "Content of .rhosts file audit report" chứa đựng các tập tin tiềm ẩn mới ".rhosts".

Chương 4:

TỔNG QUAN VỀ SỰ TỐI ƯU LINUX
(Linux General Optimization)



TỔNG QUAN VỀ SỰ TỐI ƯU LINUX

Giới thiệu tổng quan

Ở giai đoạn này bạn đã có một máy Linux server được cấu hình và được bảo mật tốt. Máy chủ server của bạn chứa đựng hầu hết các chương trình và các gói dữ liệu (package) chính yếu, chúng được cài đặt và có thể làm việc một cách chính xác với cấu hình bảo mật hệ thống cao nhất có thể. Trước khi chúng ta bắt đầu tiếp tục các bước cài đặt các dịch vụ (service) mà chúng ta cần phải chia sẻ với các khách hàng của chúng ta, điều rất quan trọng là cần tinh chỉnh máy Linux server của chúng ta. Việc điều chỉnh chúng ta sẽ thực hiện trong các phần sau sẽ được thích ứng với toàn bộ hệ thống. Nó cũng sẽ thích ứng tốt với các chương trình hiện có và tương lai, ví dụ như là các dịch vụ mà chúng ta sẽ cài đặt sau này. Nói chung, nếu bạn không dùng một bộ xử lý Intel x386, Red Hat Linux là không được tối ưu hóa cho kiến trúc CPU cụ thể nào đó của bạn (hầu hết bây giờ mọi người đều chạy Linux trên một bộ xử lý Pentium). Các phần bên dưới sẽ chỉ dẫn cho bạn thông qua các bước khác nhau để tối ưu hóa Linux server của bạn cho các bộ xử lý cụ thể nào đó, bộ nhớ, và hệ thống mạng, và cũng như hệ thống tập tin của bạn.

Tập tin "/etc/profile"

Tập tin "/etc/profile" chứa đựng mọi thứ về môi trường hệ thống và các chương trình sẽ được kích hoạt lúc khởi động. Tất cả các tùy biến của bạn mà bạn đặt vào trong tập tin này sẽ được áp dụng cho các biến môi trường trên hệ thống của bạn, vì vậy đặt các cờ hiệu tối ưu hoá trong tập tin này là một sự lựa chọn tốt nhất. Để ép sự thi hành hiệu quả nhất từ các chương trình x86 của bạn, bạn có thể sử dụng hoàn toàn việc tối ưu hoá khi biên dịch với cờ hiệu –O9. Nhiều chương trình chứa đựng –O2 trong Makefile. –O9 là cấp độ cao nhất của việc tối ưu hoá. Nó sẽ gia tăng kích thước của những gì nó sinh ra, nhưng nó chạy nhanh hơn.

Ghi chú: *Điều không luôn luôn đúng là cờ hiệu –O9 sẽ tạo sự thực hiện hiệu quả nhất cho bộ xử lý của bạn. Nếu bạn có một bộ xử lý x686 hay cao hơn thì điều này sẽ bảo đảm chắc chắn, nhưng thấp hơn x686 thì cũng không thật sự cần thiết.*

Khi biên dịch, sử dụng khóa chuyển đổi "**-fomit-frame-pointer**" cho bất kỳ bộ xử lý mà bạn có. Điều này sẽ dùng ngăn xếp cho việc truy xuất các biến. Thật không may, việc chạy từng bước (debugging) là hầu như không thể làm được với tùy chọn này. Bạn cũng có thể sử dụng các khóa chuyển đổi mcpu=cpu_type và –march=cpu_type để tối ưu hoá chương trình cho CPU được liệt kê để thích ứng tốt nhất với khả năng của GCC. Tuy nhiên, mã (code) kết quả chỉ sẽ có thể chạy trên CPU được chỉ rõ hay cao hơn.

Bên dưới là các cờ hiệu tối ưu hóa mà chúng tôi khuyên bạn nên đặt trong tập tin "/etc/profile" của bạn phụ thuộc vào cấu trúc CPU của bạn. Các tùy chọn tối ưu hoá chỉ được áp dụng khi bạn biên dịch và cài đặt một chương trình mới trên máy chủ server của chúng ta. Các tối ưu hoá này sẽ không đóng bất kỳ vai trò nào trong hệ thống nền tảng cơ bản Linux của chúng ta; nó chỉ bảo trình biên dịch của

chúng ta tối ưu hoá các chương trình mới mà chúng ta sẽ cài đặt với các cờ hiệu tối ưu hoá mà chúng ta đã chỉ định ra rõ trong tập tin "/etc/profile".

Bước 1

Dành cho CPU i686 hay PentiumPro, Pentium II, Pentium III

Trong tập tin "/etc/profile", đặt dòng này cho một họ bộ vi xử lý PentiumPro, Pentium II, và III Pro:

```
CFLAGS='-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro
-march=pentiumpro -fomit-frame-pointer -fno-exceptions'
```

Dành cho CPU i568 hay Pentium

Trong tập tin "/etc/profile", đặt dòng này cho một họ bộ vi xử lý Pentium:

```
CFLAGS='-O3 -march=pentium -mcpu=pentium -ffast-math -funroll-loops
-fomit-frame-pointer -fforce-mem -fforce-addr -malign-double -fno-exceptions'
```

Dành cho CPU i468

Trong tập tin "/etc/profile", đặt dòng này cho một họ bộ vi xử lý i468:

```
CFLAGS='-O3 -funroll-all-loops -malign-double -mcpu=i486 -march=i486
-fomit-frame-pointer -fno-exceptions'
```

Bước 2

Và bây giờ sau khi việc chọn các thông số cài đặt CPU (bao gồm i686, i568, hay i468) và một ít bên dưới trong tập tin "/etc/profile", thêm vào "**CFLAGS LANG LESSCHARSET**" vào dòng "export" :

```
export PATH PS1 HOSTNAME HISTSIZE HISTFILESIZE USER LOGNAME
MAIL INPUTRC CFLAGS LANG LESSCHARSET
```

Bước 3

Thoát ra và trở vào lại; sau việc làm này, biến môi trường CFPLAGS mới được đặt lại giá trị, phần mềm và các công cụ "cấu hình" khác sẽ nhận biết được điều đó. Việc tối ưu hoá Pentium (Pro/II/III) sẽ chỉ làm việc với các trình biên dịch egcs hay pgcc. Trình biên dịch egcs được cài đặt sẵn trên máy server của bạn một cách mặc nhiên do vậy bạn không cần phải lo lắng về điều đó.

Bên dưới là sự giải thích các tùy chọn tối ưu hóa khác nhau mà chúng ta sử dụng:

- **"-funroll-loops"**

  Tùy chọn tối ưu hóa "-funroll-loops" sẽ thực hiện sự tối ưu hoá vòng lặp đang thực thi (unrolling loop) và sẽ làm việc này chỉ cho các vòng lặp có số lần lặp có thể được xác định lúc biên dịch hoặc lúc thi hành.

- **"-funroll-all-loops"**

  Tùy chọn tối ưu hóa "-funroll-all-loops" cũng sẽ thực hiện sự tối ưu hoá cho vòng lặp đang thực thi và thực hiện việc này cho tất cả các vòng lặp.

- **"-ffast-math"**

  Tùy chọn tối ưu hóa "-ffast-math" sẽ chấp nhận các trình biên dịch GCC. Trong tầm quan trọng của qui luật tối ưu cho tốc độ để can thiệp vào một vài qui tắc hay chi tiết kỹ thuật ANSI hay IEEE.

- **"-malign-double"**

  Tùy chọn tối ưu hóa "-malign-double" sẽ điều khiển có hay không trình biên dịch GCC sắp hàng các biến double, long double, và long long trên một đường biên hai từ (two-word boundary) hay một đường biên một từ (one-word boundary). Việc này sẽ sinh ra qui luật mà khi chạy có phần nhanh hơn trên một "Pentium" với nhiều bộ nhớ.

- **"-mcpu=cpu_type"**

  Tùy chọn tối ưu hóa "-mcpu=cpu_type" sẽ thiết đặt CPU mặc định để sử dụng cho loại máy khi ra các chỉ thị lập chương trình.

- **"-march=cpu_type"**

  Tùy chọn tối ưu hóa "-march=cpu_type" sẽ sinh ra các chỉ thị cho máy và loại cpu. Việc chỉ rõ tùy chọn "-march=cpu_type" cũng bao hàm chỉ rõ tùy chọn "-mcpu=cpu_type" trong việc biên dịch của bạn.

- **"-fforce-mem"**

  Tùy chọn tối ưu hóa "-fforce-mem" sẽ đưa ra các qui luật tốt bởi sự ép buộc các phép toán về bộ nhớ được sao chép sang các thanh ghi (register) trước khi thực hiện phép tính số học trên chúng và tạo ra tất cả các tham chiếu bộ nhớ tiềm ẩn các biểu thức con chung.

- **"-fforce-addr"**

  Tùy chọn tối ưu hóa "-fforce-addr" sẽ đưa ra các qui luật tốt bởi sự ép buộc các địa chỉ bộ nhớ có giá trị không đổi sao chép sang các thanh ghi (register) trước khi thực hiện phép tính số học trên chúng.

- **"-fomit-frame-pointer"**

  Tùy chọn tối ưu hóa "-fomit-frame-pointer" là một trong những tùy chọn đáng quan tâm nhất sẽ cho phép chương trình không lưu giữ con trỏ khung (frame pointer) trong một thanh ghi cho các hàm không cần đến nó. Để tránh các chỉ thị lưu trữ, cài đặt và khôi phục lại các frame pointer; nó cũng tạo ra một biến thanh ghi đặc biệt trong nhiều hàm và việc thi hành debug là hầu như không thể làm được trên hầu hết các máy.

Ghi chú quan trọng: *Tất cả các tối ưu hoá mà chúng ta sẽ mô tả sắp tới trong quyển sách này tham chiếu một cách mặc nhiên tới một họ bộ vi xử lý Pentium II/III. Do vậy nếu được yêu cầu bạn phải sửa lại cho đúng các cờ hiệu tối ưu hóa cho một loại bộ xử lý CPU cụ thể nào đó trong tập tin "/etc/profile" và cũng trong suốt thời gian biên dịch của bạn.*

Các kết quả chuẩn

Các tóm tắt bởi kiến trúc

Phụ thuộc vào kiến trúc của bộ vi xử lý và phiên bản của trình biên dịch của bạn (GCC/EGCS), các kết quả tối ưu hóa có thể thay đổi khác nhau. Các sơ đồ bên dưới sẽ giúp cho bạn chọn các cờ hiệu biên dịch tốt nhất cho trình biên dịch/kiến trúc CPU của bạn.

Phiên bản của trình biên dịch được cài đặt trên Red Hat Linux 6.1 và 6.2 là egcs 2.91.66, nhưng để bảo đảm phải kiểm tra nó ngay trước khi chọn các tùy chọn tối ưu hóa trình biên dịch của bạn.

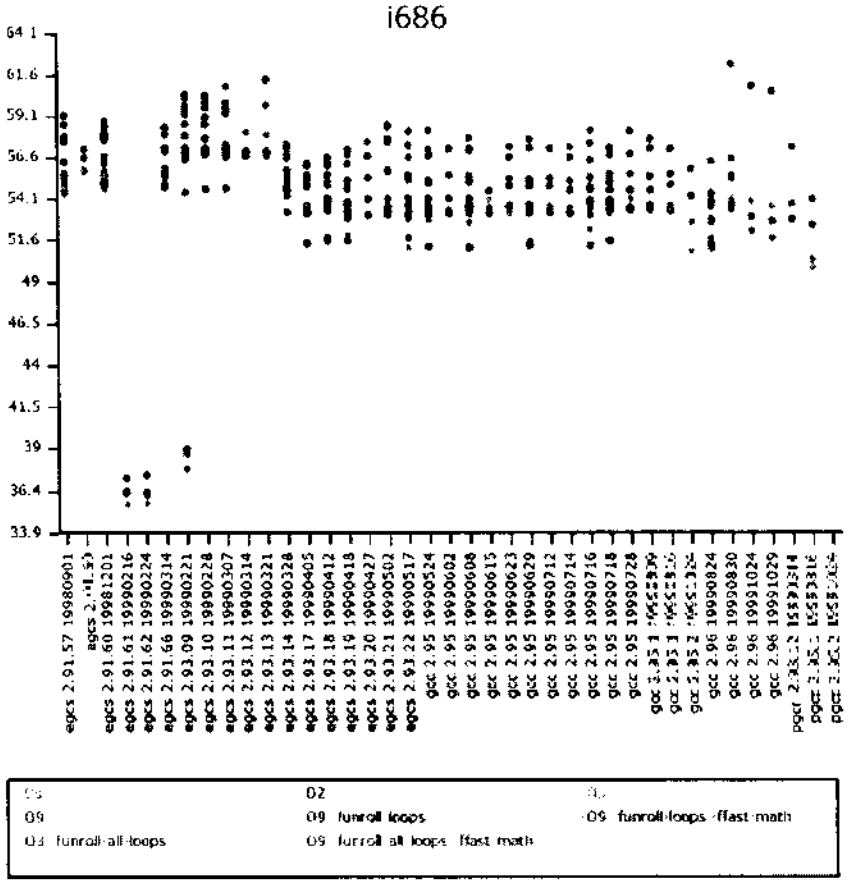
- Để kiểm tra phiên bản của trình biên dịch được cài đặt trên hệ thống, sử dụng lệnh sau:

```
[root@deep /]# egcs --version
egcs-2.91.66
```

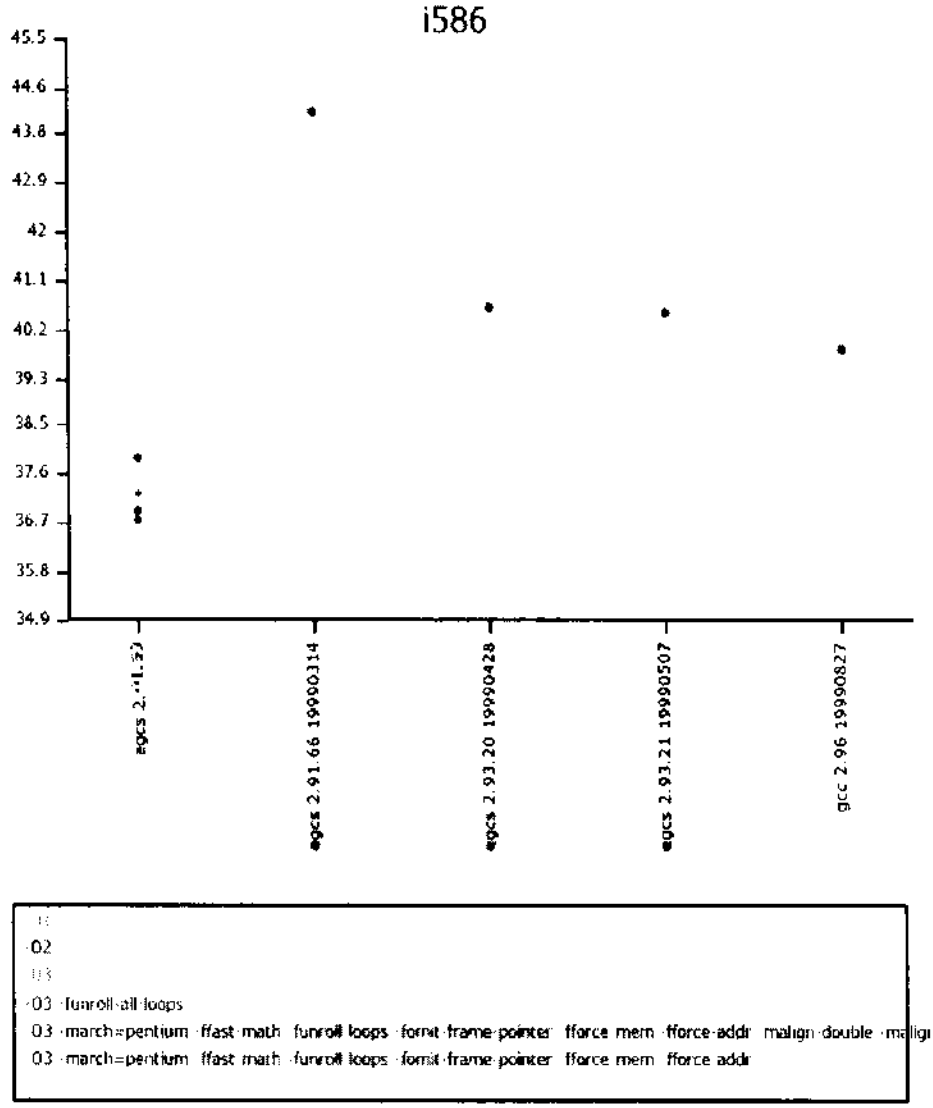
Ghi chú: *Tất cả các kết quả chuẩn, và các kết quả tương lai, có thể được lưu trữ từ trang chủ của GCC ở địa chỉ sau: http://egcs.cygnus.com/*

Bây giờ như một ví dụ:

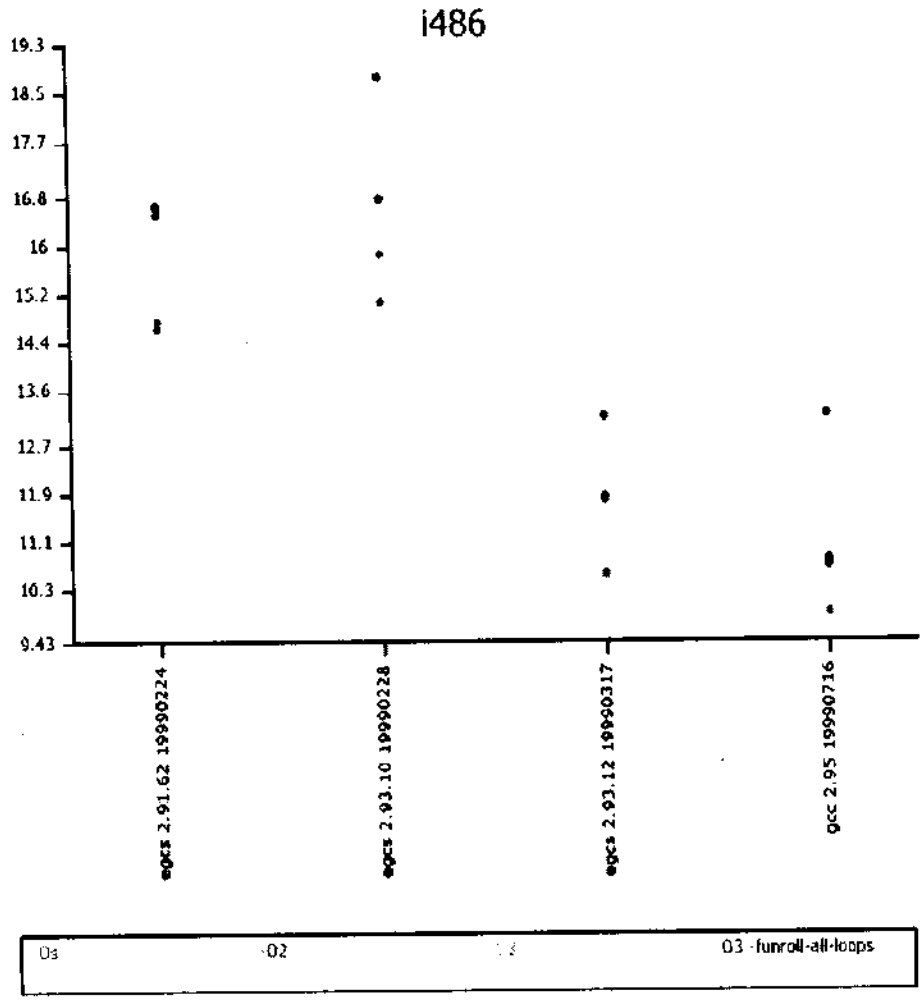
Đối với một bộ vi xử lý Pentium II/III CPU (i686) với phiên bản của trình biên dịch egcs-2.91.66, các tùy chọn tối ưu hoá tốt nhất sẽ là: **CFLAGS='-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -march=pentiumpro -fomit-frame-pointer -fno-exceptions'**



Đối với một bộ vi xử lý Pentium CPU (i586) với phiên bản của trình biên dịch egcs-2.91.66, các tùy chọn tối ưu hoá tốt nhất sẽ là: **CFLAGS='-O3 -march=pentium -mcpu=pentium -ffast-math -funroll-loops -fomit-frame-pointer -fforce-mem -fforce-addr -malign-double -fno-exceptions'.**



Đối với một bộ vi xử lý CPU (i486) với phiên bản của trình biên dịch egcs-2.91.66, các tùy chọn tối ưu hoá tốt nhất sẽ là: **CFLAGS='-O3 -funroll-all-loops -malign-double -mcpu=i486 -march=i486 -fomit-frame-pointer -fno-exceptions'.**



Các tham số "bdflush"

Tập tin bdflush liên quan mật thiết với các thao tác của bộ nhớ ảo (virtual memory = VM) hệ thống con của Linux Kernel và một ít ảnh hưởng đến cách sử dụng trên đĩa. Tập tin này (/proc/sys/vm/bdflush) điều khiển sự hoạt động của bdflush kernel daemon. Chúng ta hiệu chỉnh hợp lý tập tin này nhằm để cải tiến việc thi hành của hệ thống tập tin. Bằng cách thay đổi một vài giá trị từ các giá trị mặc nhiên được hiển thị bên dưới, hệ thống dường như phản ứng nhanh hơn; ví dụ thời gian chờ đợi để ghi vô đĩa ít hơn và như vậy tránh một vài sự kẹt nối truy cập đĩa.

Cài đặt mặc nhiên cho các thông số "bdflush" trên Red Hat Linux là:

"40 500 64 256 500 3000 500 1884 2"

- Để thay đổi các giá trị bdflush, gõ vào các lệnh sau trên máy trạm (terminal) của bạn:

Trên Red Hat Linux 6.1

```
[root@deep /]# echo "100 1200 128 512 15 5000 500 1884
2">/proc/sys/vm/bdflush
```

Bạn có thể thêm các lệnh trên vào trong tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó lại trong lần kế tiếp khi bạn khởi động hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Improve file system performance
vm.bdflush = 100 1200 128 512 15 5000 500 1884 2
```

Bạn phải khởi động lại hệ thống mạng cho việc thay đổi có hiệu quả. Lệnh để khởi động lại hệ thống mạng là như sau:

- Để khởi động lại tất cả các thiết bị mạng được thực hiện bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters          [ OK ]
Bringing up interface lo            [ OK ]
Bringing up interface eth0          [ OK ]
Bringing up interface eth1          [ OK ]
```

Trong ví dụ trên, tham khảo các hướng dẫn trong tập tin "/usr/src/linux/Documentation/sysctl/vm.txt", tham số đầu tiên (100%) quản lý một số tối đa các bộ đệm dirty (dirty buffer) trong nơi cất giữ bộ đệm (buffer cache). Dirty có nghĩa là nội dung của vùng đệm (buffer) vẫn phải được ghi vào đĩa (trái với việc là phải xóa sạch buffer, hay không nhớ về nó). Cài đặt thông số này tới một giá trị cao có nghĩa là Linux có thể trì hoãn việc ghi đĩa trong một thời gian dài, nhưng nó cũng nói lên rằng nó sẽ phải làm nhiều I/O ngay lập tức khi bộ nhớ không còn nhiều. Một giá trị thấp sẽ làm cho việc đọc/ghi đĩa từ nhiều hơn.

Tham số thứ hai (1200) (ndirty) cung cấp một số tối đa các dirty buffer mà bdflush có thể ghi ra đĩa từ trong một thời điểm. Một giá trị cao sẽ hiểu là được trì hoãn, bursty I/O, trong khi một giá trị nhỏ có thể dẫn đến việc thiếu bộ nhớ trong khi bdflush thì vẫn thường chưa đủ để kích động.

Tham số thứ ba (128) (nrefill) là số buffer mà bdflush sẽ thêm vào danh sách của các free buffer (buffer tổng) khi refill_freelist() được gọi. Điều này cũng rất cần thiết để chỉ định các free buffers sẵn sàng trước, vì các buffers thông thường có một kích thước khác nhau hơn các trang bộ nhớ (memory page) và một

vài sự tính toán cần phải làm sẵn sàng trước. Một giá trị cao hơn, thì nhiều bộ nhớ sẽ bị lãng phí và thường ít cần refill_freelist() để chạy.

Khi refill_freelist() (mặc định 512) có giá trị lớn hơn các bộ đệm nref_dirt dirty, nó sẽ kích động bdflush.

Cuối cùng, các thông số age_buffer(50*HZ) và age_super(5*HZ) quản lý thời gian tối đa Linux chờ trước khi ghi một dirty buffer ra đĩa. Giá trị thì được biểu lộ trong tích tắc, số tích tắc là trên một phần trăm giây. Age_buffer là tuổi thọ tối đa cho các khối dữ liệu, trong khi age_super thì cho hệ thống tập tin metadata.

Tham số 15 và hai thông số sau cùng (1884 và 2) thì không được sử dụng bởi hệ thống do vậy chúng ta không cần thay đổi các giá trị mặc định.

Ghi chú: *Cần xem tập tin "/usr/src/linux/Documentation/sysctl/vm.txt" cho nhiều thông tin hơn về làm thế nào cải tiến các thông số kernel liên quan tới bộ nhớ ảo.*

Các thông số "buffermem"

Tập tin buffermem thì cũng rất liên quan mật thiết với sự hoạt động của bộ nhớ ảo (Virtual Memory) hệ thống con của Linux kernel. Gía trị trong tập tin "/proc/sys/vm/buffermem" điều khiển bao nhiêu bộ nhớ sẽ được dùng cho bộ nhớ buffer (tính theo phần trăm (%)). Điều quan trọng phải lưu ý rằng tỉ lệ phần trăm thì được tính toán như một tỉ lệ phần trăm của tất cả các bộ nhớ của hệ thống.

Các cài đặt mặc nhiên cho các thông số "buffermem" trên Red Hat Linux là:

```
"2 10 60"
```

- Để thay đổi các giá trị của buffermem, gõ vào các lệnh sau trên máy trạm (terminal) của bạn:

Trên Red Hat Linux 6.1

```
[root@deep /]# echo "80 10 60" >/proc/sys/vm/buffermem
```

Bạn có thể thêm các lệnh trên vào trong tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó lần nữa trong lần kế tiếp khi bạn khởi động lại hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Improve virtual memory performance
vm.buffermem = 80 10 60
```

Bạn phải khởi động lại hệ thống mạng cho việc thay đổi có hiệu quả. Lệnh để khởi động lại hệ thống mạng là như sau:

- Để khởi động lại tất cả các thiết bị mạng được thực hiện bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
```

```
Setting network parameters          [ OK ]
Bringing up interface lo            [ OK ]
Bringing up interface eth0          [ OK ]
Bringing up interface eth1          [ OK ]
```

Trong ví dụ trên, thông qua tập tin "/usr/src/linux/Documentation/sysctl/vm.txt", thông số đầu tiên (80%) nghĩa là sử dụng tối thiểu 80% bộ nhớ cho buffer cache; tỉ lệ phần trăm tối thiểu của bộ nhớ sẽ được dùng trên bộ nhớ đệm (buffer memory).

Hai tham số sau cùng (10 và 60) không được hệ thống dùng đến vì vậy chúng ta không cần thay đổi các giá trị mặc định.

Ghi chú: *Xem lại tập tin "/usr/src/linux/Documentation/sysctl/vm.txt" cho nhiều thông tin hơn về làm thế nào cải tiến các thông số kernel liên quan tới bộ nhớ ảo.*

Các thông số "ip_local_port_range"

Tham số ip_local_port_range"/proc/sys/net/ipv4/ip_local_port_range" định nghĩa phạm vi cổng nội bộ (local port range) được sử dụng bởi sự lưu thông trên TCP và UDP để chọn cổng nội bộ. Bạn sẽ thấy trong các tham số của tập tin này hai số: số đầu tiên là cổng cục bộ đầu tiên cho phép cho việc lưu thông TCP và UDP trên máy server, số thứ hai là số cổng cục bộ sau cùng. Đối với các hệ thống có nhu cầu sử dụng cao bạn có thể thay đổi các thông số mặc nhiên của nó tới 32768-61000 (đầu-cuối).

Các cài đặt mặc nhiên cho các thông số "ip_local_port_range" trên Red Hat Linux là: **"1024 4999"**

- Để thay đổi các giá trị của ip_local_port_range, gõ vào các lệnh sau trên máy trạm (terminal) của bạn:

Trên Red Hat Linux 6.1

```
[root@deep /]# echo "32768 61000" >/proc/sys/net/ipv4/ip_local_port_range
```

Bạn có thể thêm các lệnh trên vào trong tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó lần nữa trong lần kế tiếp khi bạn khởi động lại hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Improve virtual memory performance
net.ipv4.ip_local_port_range = 32768 61000
```

Bạn phải khởi động lại hệ thống mạng cho việc thay đổi có hiệu quả. Lệnh để khởi động lại hệ thống mạng là như sau:

- Để khởi động lại tất cả các thiết bị mạng được thực hiện bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                 [ OK ]
Bringing up interface lo                   [ OK ]
Bringing up interface eth0                 [ OK ]
Bringing up interface eth1                 [ OK ]
```

Tập tin "/etc/nsswitch.conf"

Tập tin "/etc/nsswitch.conf" được dùng để cấu hình các dịch vụ mà được dùng để xác định các thông tin chẳng hạn như các hostname, các tập tin mật mã(password files), và các tập tin nhóm (group files). Hai thông tin, "password files" và "group files" trong trường hợp của chúng ta thì không được dùng, bởi vì chúng ta không sử dụng các dịch vụ NIS trên máy server của chúng ta. Chính vì vậy, chúng ta sẽ tập trung vào dòng "hosts" trong tập tin này.

Soạn thảo tập tin nsswitch.conf (vi /etc/nsswitch.conf) và thay đổi dòng "hosts" trở thành:

```
"hosts:  dns files"
```

Điều này có nghĩa là đối với các chương trình mà cần phải phân giải địa chỉ. Chúng sẽ sử dụng đặc trưng dns đầu tiên và sau đó dùng đến tập tin "/etc/hosts" nếu các DNS servers chưa sẵn sàng hoặc không thể phân giải địa chỉ.

Và chúng tôi cũng khuyên bạn xóa tất cả các ví dụ của các dịch vụ NIS từ mỗi dòng của tập tin này trừ khi bạn đang dùng NIS! kết quả trông giống như thế này:

```
passwd:      files
shadow:      files
group:       files
hosts:       dns files
bootparams:  files
ethers:      files
netmasks:    files
networks:    files
protocols:   files
rpc:         files
services:    files
automount:   files
aliases:     files
```

Thông số "file-max"

Tập tin file-max "/proc/sys/fs/file-max" đặt tối đa số tập tin xử lý (file-handle) mà Linux kernel sẽ cấp phát. Chúng ta hiệu chỉnh tập tin này để cải tiến số lượng tập tin được mở bằng cách tăng giá trị của "/proc/sys/fs/file-max" tới một giá trị hợp lý nào đó như 256 cho máy tính có 4M RAM: ví dụ đối với một máy tính có 128MB RAM , đặt giá trị nó là 8192 (128/4=32 32*256=8192).

Các cài đặt mặc nhiên cho các thông số "file-max" trên Red Hat Linux là:

"4096"

- Để thay đổi các giá trị của file-max, gõ vào các lệnh sau trên máy trạm (terminal) của bạn:

Trên Red Hat Linux 6.1

[[root@deep /]# echo "8192" >/proc/sys/fs/file-max

Bạn có thể thêm các lệnh trên vào trong tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó lần nữa trong lần kế tiếp khi bạn khởi động lại hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Improve the number of open files
fs.file-max = 8192
```

Bạn phải khởi động lại hệ thống mạng cho việc thay đổi có hiệu quả. Lệnh để khởi động lại hệ thống mạng là như sau:

- Để khởi động lại tất cả các thiết bị mạng được thực hiện bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters                [ OK ]
Bringing up interface lo                  [ OK ]
Bringing up interface eth0                [ OK ]
```

***Ghi chú**: Khi bạn nhận thường xuyên từ máy server của bạn nhiều thông điệp với các lỗi về mở quá nhiều tập tin, bạn cũng phải cần nâng lên một giá trị hữu hạn này. Giá trị mặc nhiên là 4096. Một file server (máy chủ về tập tin) hay một web server (máy chủ về web) rất cần nhiều tập tin mở.*

Thông số "inode-max"

Tập tin inode-max "/proc/sys/fs/file-max" đặt lại tối đa số inodes-handler. Chúng ta hiệu chỉnh tập tin này phù hợp để cải tiến số lượng inode được mở bằng cách tăng giá trị của "/proc/sys/fs/inode-max" tới một giá trị xấp xỉ 3 đến 4 lần (8192*4=32768) số lượng tập tin mở mà chúng ta đã đặt thông số "file-max" ở

trên. Điều này được làm bởi vì số các inode được mở là tối thiểu một trên một tập tin được mở, và thường lớn hơn nhiều đối với các tập tin lớn.

Các cài đặt mặc nhiên cho các thông số "file-max" trên Red Hat Linux là:

"16376"

- Để điều chỉnh giá trị của inode-max lên 128MB RAM, gõ vào các lệnh sau trên máy trạm (terminal) của bạn:

Trên Red Hat Linux 6.1

```
[root@deep /]# echo "32768" >/proc/sys/fs/inode-max
```

Thêm vào các lệnh trên vào trong tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó lần nữa trong lần kế tiếp khi bạn khởi động lại hệ thống.

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Improve the number of inodes opened
fs.inode-max = 32768
```

Bạn phải khởi động lại hệ thống mạng cho việc thay đổi có hiệu quả. Lệnh để khởi động lại hệ thống mạng là như sau:

- Để khởi động lại tất cả các thiết bị mạng được thực hiện bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters          [ OK ]
Bringing up interface lo            [ OK ]
Bringing up interface eth0          [ OK ]
```

Ghi chú: *Nếu các inode trên hệ thống của bạn thường xuyên vượt qua giá trị inode đang được thiết đặt, bạn nên tăng giá trị này. Đừng quên rằng giá trị mà bạn nhập vào cho thông số "inode-max" phụ thuộc vào giá trị của "file-max". Một tập tin server hay một web server cần rất nhiều các inodes được mở.*

Thông số "ulimit"

Tự bản thân Linux có một "Max Processes" trên giới hạn của người sử dụng hệ thống. Đặc trưng này cho phép chúng ta điều khiển số tiến trình (processes) một người sử dụng đang tồn tại trên server được phép để có. Để cải tiến việc thực hiện, chúng ta có thể an toàn đặt giới hạn các tiến trình (processes) cho người sử dụng có quyền cao nhất trong hệ thống là "root" để không bị giới hạn.

Soạn thảo tập tin **.bashrc** (vi /root/.bashrc) và thêm vào dòng sau:

```
ulimit -u unlimited
```

Bạn phải thoát ra và vào lại hệ thống từ máy trạm của bạn để việc thay đổi các tham số có hiệu lực. Để kiểm tra rằng bạn đã sẵn sàng chưa, bảo đảm rằng khi bạn gõ lệnh **ulimit –a** với quyền root trên máy trạm của bạn, nó hiển thị "unlimited" kế tiếp là tối đa tiến trình người sử dụng (max user processes).

```
[root@deep /]# ulimit –a
core file size (blocks)     1000000
data seg size (kbytes)      unlimited
file size (blocks)          unlimited
max memory size (kbytes)    unlimited
stack size (kbytes)         8192
cpu time (seconds)          unlimited
max user processes          unlimited ___ this line.
pipe size (512 bytes)       8
open files                  1024
virtual memory (kbytes)     2105343
```

Ghi chú: *Bạn cũng có thể thực hiện ulimit –u unlimited tại dấu nhắc lệnh thay vì thêm nó vào tập tin "/root/.bashrc".*

Tăng giới hạn hệ thống trên các tập tin mở

Một tiến trình (process) trên Red Hat 6.0 với Kernel 2.2.5 có thể mở ít nhất 31000 tập tin mô tả, theo cách này và một process trên kernel 2.2.12 có thể mở ít nhất 90000 tập tin. Ranh giới cao thấp dường như là bộ nhớ sẵn có trong hệ thống. Để tăng số các tập tin mở tới 90000 cho tài khoản root thực hiện như sau:

Soạn thảo tập tin .bashrc (vi /root/.bashrc) và thêm vào dòng sau đây:

```
ulimit -n 90000
```

Bạn phải thoát ra và vào lại hệ thống từ máy trạm của bạn để việc thay đổi các tham số có hiệu lực. Để kiểm tra rằng bạn đã sẵn sàng chưa, bảo đảm rằng khi bạn gõ lệnh **ulimit –a** với quyền root trên máy trạm của bạn, nó hiển thị "90000" kế tiếp là **open files.**

```
[root@deep /]# ulimit –a
core file size (blocks)     1000000
data seg size (kbytes)      unlimited
file size (blocks)          unlimited
max memory size (kbytes)    unlimited
stack size (kbytes)         8192
cpu time (seconds)          unlimited
max user processes          unlimited
pipe size (512 bytes)       8
open files                  90000 ← this line.
virtual memory (kbytes)     2105343
```

Ghi chú: *Trong kernel cũ 2.2, số tập tin được mở trên process vẫn được giới hạn là 1024, cho dù các thông tin bên trên thay đổi.*

Thuộc tính "atime"

Linux ghi lại thông tin về các tập tin được tạo và được hiệu chỉnh sau cùng cũng như nó được truy cập lần sau cùng khi nào. Có một đòi hỏi được kết hợp với việc ghi lại thời gian lần truy cập sau cùng. Hệ thống tập tin ext2 của Linux có một thuộc tính cho phép người sử dụng có quyền cao nhất đánh dấu các tập tin riêng rẽ để sau đó thời gian truy cập sau cùng của chúng không được ghi lại. Điều này có thể đưa đến việc cải tiến thực hiện đáng kể trên các tập tin hay thường được truy cập và thay đổi thường xuyên ví dụ như nội dung của thư mục "/var/spoot/news".

Để đặt thuộc tính cho một tập tin, dùng:

```
[root@deep /]# chattr +A filename
```

Cho toàn bộ cây thư mục, thực hiện một số việc sau:

```
[root@deep /root]# chattr -R +A /var/spool/
[root@deep /root]# chattr -R +A /cache/
[root@deep /root]# chattr -R +A /home/httpd/ona/
```

Thuộc tính "noatime"

Linux có một tùy chọn gắn vào đặc biệt cho hệ thống tập tin được gọi là **noatime** mà có thể được thêm vào mỗi dòng mà nó chỉ rõ một hệ thống tập tin trong tập tin "/etc/fstab". Nếu một hệ thống tập tin đã được gắn vào với tùy chọn này, người đọc truy xuất tới hệ thống tập tin sẽ không đưa đến kết quả trong việc cập nhật thông tin **atime** được kết hợp với tập tin giống như chúng ta đã giải thích bên trên. Tầm quan trọng của cài đặt noatime là nó loại trừ những nhu cầu cần thiết bởi hệ thống để thực hiện việc ghi vào tập tin hệ thống cho các tập tin mà chúng đang được đọc. Bởi vì việc ghi có thể tốn hao mức độ nào đó với hệ thống, điều này có thể đưa đến hữu ích trong việc đo lường sự thực hiện của hệ thống. Chú ý rằng thời gian ghi thông tin vào một tập tin sẽ tiếp tục được cập nhật bất cứ lúc nào tập tin được ghi. Trong ví dụ của chúng ta bên dưới, chúng ta sẽ đặt tùy chọn noatime tới hệ thống tập tin "/chroot" của chúng ta.

Soạn thảo tập tin **fstab** (vi /etc/fstab) và thêm trong dòng mà tham chiếu tới hệ thống tập tin "/chroot" tùy chọn noatime sau tùy chọn defaults như được trình bày bên dưới:

```
E.I: /dev/sda7 /chroot ext2 defaults,noatime 1 2
```

Bạn phải khởi động lại hệ thống của bạn (Reboot) cho việc thay đổi có hiệu lực:

```
[root@deep /]# reboot
```

Sau đó kiểm tra các kết quả của bạn với các lệnh sau:

```
[root@deep]# cat /proc/mounts
/dev/root / ext2 rw 0 0
/proc /proc proc rw 0 0
/dev/sda1 /boot ext2 rw 0 0
```

```
/dev/sda8 /cache ext2 rw 0 0
/dev/sda7 /chroot ext2 rw,noatime 0 0
/dev/sda6 /home ext2 rw 0 0
/dev/sda11 /tmp ext2 rw 0 0
/dev/sda5 /usr ext2 rw 0 0
/dev/sda9 /var ext2 rw 0 0
none /dev/pts devpts rw 0 0
```

Nếu bạn thấy cái gì đó như thế này : /dev/sda7 /chroot ext2 rw,**noatime** 0 0, thì xin chúc mừng bạn!

Swap partition

Đặt các swap partition của bạn gần nơi bắt đầu của ổ đĩa của bạn có thể cung cấp cho bạn một số cải tiến có thể chấp nhận được. Nơi bắt đầu của ổ đĩa của bạn là phần vật lý được đặt ở bên ngoài của phần chia rãnh (cylinder), và đầu đọc/ghi có thể bao phủ nhiều tiếp xúc trên vòng quay của đĩa. Cụ thể chúng ta thấy các phân vùng (partition) được đặt ở cuối của ổ đĩa làm việc 3MB/s chậm hơn khi sử dụng lệnh hdparm –t.

Điều chỉnh sự thực hiện đĩa cứng IDE

Sự thực thi (performance) tăng được tường trình trên sự hoạt động đọc/ghi đĩa (I/O disk) bằng việc cài đặt các ổ đĩa IDE sử dụng DMA, tốc độ truyền 32-bit và các chế độ multiple sector. Kernel dường như sử dụng nhiều sự sắp đặt để duy trì ngoại trừ nó thực hiện các việc khác. Lệnh sau thay đổi việc cài đặt ổ đĩa của bạn là "hdparm".

- Để hỗ trợ 32-bit I/O trên các PCI bus, dùng lệnh sau:

  [root@deep /]# **/sbin/hdparm -c 1 /dev/hda** (or hdb, hdc etc).

Điều này sẽ như thường lệ, phụ thuộc vào kiểu ổ cứng IDE, cắt thời gian đọc bộ đệm đĩa bằng 2. Trang liệt kê hướng dẫn chi tiết (manpage) "hpparm" (8) nói rằng bạn có thể cần dùng –c 3 cho một số chipset. Tất cả ổ đĩa (E)IDE vẫn chỉ có một kết nối 16-bit trên ribbon cable từ card giao tiếp.

- Để hỗ trợ DMA, dùng lệnh sau:

  [root@deep /]# **/sbin/hdparm -d 1 /dev/hda** (or hdb, hdc etc).

Điều này có thể phụ thuộc vào việc hỗ trợ cho motherboard chipset của bạn, được biên dịch vào kernel của bạn. Cũng lệnh này sẽ cho phép hỗ trợ DMA cho ổ đĩa cứng của bạn, nó sẽ cắt bỏ thời gian dành để đọc đĩa và cải tiến sự thực thi là 2.

- Để hỗ trợ truyền multiword DMA mode 2, dùng lệnh sau:

  [root@deep /]# **/sbin/hdparm -d 1 -X34 /dev/hda** (or hdb, hdc etc).

Việc đặt mode truyền IDE này cho các ổ đĩa mới (E)IDE/ATA2. (Kiểm tra phần hướng dẫn phần cứng của bạn để xem nếu bạn có nó).

- Để hỗ trợ truyền dữ liệu UltraDMA mode 2, dùng lệnh sau:

  [root@deep /]# **/sbin/hdparm -d 1 -X66 /dev/hda** (or hdb, hdc etc)

Bạn sẽ cần phải chuẩn bị chipset cho UltraDMA biết trước. Và cũng vậy, xem trang hướng dẫn về "hdparm" để biết nhiều thông tin hơn. Dùng lệnh này hết sức thận trọng!

- Để hỗ trợ multiple sector mode I/O, dùng lệnh sau:

```
[root@deep /]# /sbin/hdparm -m XX /dev/hda (or hdb, hdc etc)
```

Trong đó "XX" là cài đặt tối đa được hỗ trợ bởi ổ đĩa của bạn. Cờ hiệu –i có thể dùng tìm việc cài đặt tối đa được hỗ trợ bởi ổ đĩa cài đặt: tìm kiếm MaxMultiSect trong kết quả xuất.

```
[root@deep /]# /sbin/hdparm -i /dev/hda (or hdb, hdc etc)
/dev/hda:
Model=Maxtor 7540 AV, FwRev=GA7X4647, SerialNo=L1007YZS
Config={ HardSect NotMFM HdSw>15uSec Fixed DTR>5Mbs FmtGapReq }
RawCHS=1046/16/63, TrkSize=0, SectSize=0, ECCbytes=11
BuffType=3(DualPortCache), BuffSize=32kB, MaxMultSect=8, MultSect=8
DblWordIO=yes, maxPIO=2(fast), DMA=yes, maxDMA=1(medium)
CurCHS=523/32/63, CurSects=379584528, LBA=yes, LBA=yes, LBAsects=1054368
tDMA={min:150,rec:150}, DMA modes: sword0 sword1 *sword2 *mword0
IORDY=on/off, tPIO={min:240,w/IORDY:180}, PIO modes: mode3
```

Chế độ multiple sector (aka IDE Block Mode), là một đặc trưng của hầu hết các ổ đĩa cứng IDE, cho phép việc truyền của nhiều sector trên ngắt vào ra (I/O interrupt) hơn thường lệ là một sector trên ngắt. Khi đặc trưng này được hỗ trợ, nó giảm tổng phí (overhead) hệ điều hành cho việc đọc/ghi đĩa khoảng 30-50%. Trên nhiều hệ thống nó cũng tăng năng suất truyền dữ liệu của bất cứ nơi đâu từ 5% tới 50%.

Bạn có thể kiểm tra các kết quả của việc thay đổi bằng cách chạy "hdparm" trong mode test performance:

```
[root@deep /]# /sbin/hdparm -t /dev/hda (or hdb, hdc etc).
```

Ghi chú: *Một khi bạn có một tập hợp các tùy chọn "hdparm", bạn có thể đặt các lệnh trong tập tin "/etc/rc.d/rc.local" để chạy nó mỗi lần khi bạn khởi động lại máy. Cũng vậy, đừng quên khởi động lại máy server cho các tùy chọn có tác dụng.*

Xử lý nhiều các kết nối theo thời gian với TCP/IP

Lợi điểm này chỉ tạo các giá trị thời gian mặc định cho kết nối TCP/IP thấp hơn do vậy nhiều kết nối có thể được xử lý cùng thời gian trên giao thức TCP/IP của bạn. Theo sau đây sẽ giảm tổng số thời gian Linux của bạn cố gắng hoàn thành việc đóng một kết nối và tổng số thời gian trước khi nó sẽ hủy bỏ một kết nối cũ. Điều này sẽ giảm một số phần mở rộng IP không cần đến:

Cài đặt mặc định cho các thông số TCP/IP chúng ta sẽ thay đổi dưới Red Hat Linux là:

For the tcp_fin_timeout "180"
For the tcp_keepalive_time "7200"
For the tcp_window_scaling "1"
For the tcp_sack "1"
For the tcp_timestamps "1"

- Để điều chỉnh các giá trị TCP/IP mới, gõ các lệnh sau trên máy terminal của bạn:

Trên Red Hat Linux 6.1

```
[root@deep /]# echo 30 > /proc/sys/net/ipv4/tcp_fin_timeout
[root@deep /]# echo 1800 >/proc/sys/net/ipv4/tcp_keepalive_time
[root@deep /]# echo 0 > /proc/sys/net/ipv4/tcp_window_scaling
[root@deep /]# echo 0 > /proc/sys/net/ipv4/tcp_sack
[root@deep /]# echo 0 > /proc/sys/net/ipv4/tcp_timestamps
```

***Ghi chú**: Thực thi các lệnh trên và đặt chúng vào trong tập tin "/etc/rc.d/rc.local" như vậy bạn không cần gõ chúng lần nữa mỗi khi bạn khởi động lại hệ thống:*

Trên Red Hat Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào các dòng sau:

```
# Decrease the time default value for tcp_fin_timeout connection
net.ipv4.tcp_fin_timeout = 30
# Decrease the time default value for tcp_keepalive_time connection
net.ipv4.tcp_keepalive_time = 1800
# Turn off the tcp_window_scaling
net.ipv4.tcp_window_scaling = 0
# Turn off the tcp_sack
net.ipv4.tcp_sack = 0
# Turn off the tcp_timestamps
net.ipv4.tcp_timestamps = 0
```

Bạn phải khởi động lại hệ thống mạng của bạn để các thay đổi có tác dụng. Lệnh để khởi động lại mạng như sau:

- Để khởi động lại tất cả các thiết bị mạng được thực hiện bằng tay trên hệ thống của bạn, dùng lệnh sau:

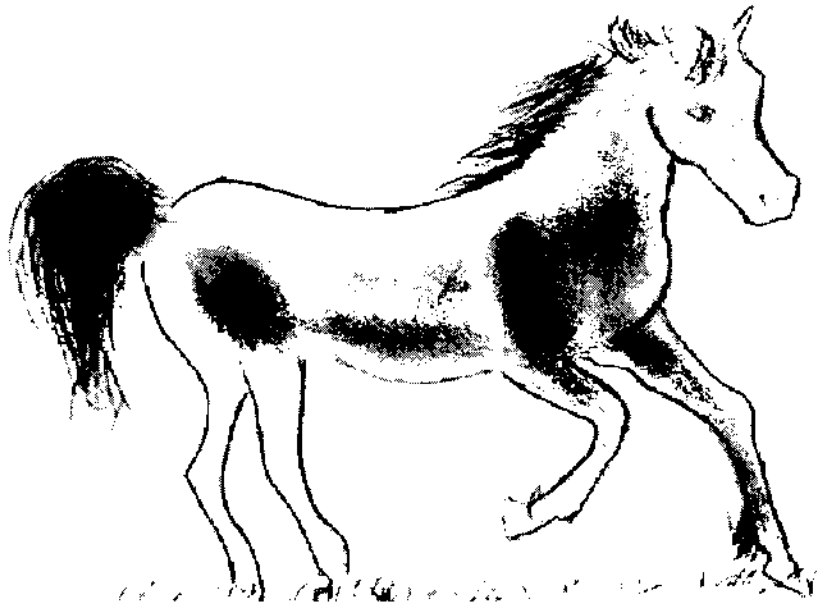
```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters          [ OK ]
Bringing up interface lo            [ OK ]
Bringing up interface eth0          [ OK ]
```

Chương 5:

CẤU HÌNH VÀ XÂY DỰNG MỘT KERNEL
(Bảo mật và tối ưu)

Trong chương này chúng ta sẽ đề cập:

- Linux Kernel.
- Làm một đĩa mềm boot cấp cứu.
- Bảo mật Kernel.
- Sự cấu hình Kernel.
- Cài đặt kernel mới.
- Xoá bỏ chương trình, file, dòng lệnh liên quan tới các module.
- Tạo một đĩa cứu nguy mới.
- Tạo đĩa mềm khởi động cấp cứu.
- Cập nhật /dev.



LINUX KERNEL

Tổng quan

Thế là bây giờ chúng ta đã hình dung ra Linux server, nhưng điều quan trọng là chúng ta phải biết thành phần nào là quan trọng nhất của server chúng ta, câu trả lời là kernel. Linux kernel là nòng cốt của hệ điều hành, không có nó thì sẽ không có Linux. Vì vậy chúng ta phải quan tâm tới kernel, cấu hình nó cho thích hợp những gì chúng ta cần và biên dịch đúng những đặc trưng chúng ta thật sự cần. Việc đầu tiên cần làm là xây dựng một kernel thích hợp nhất cho hệ thống của bạn. Thật ra thì rất đơn giản để làm việc này nhưng trong bất kỳ trường hợp nào bạn nên tham khảo tập tin README trong thư mục /usr/src/linux. Khi cấu hình kernel, bạn chỉ biên dịch những gì bạn cần và sử dụng. Bốn lý do chính để suy nghĩ là: Kernel sẽ nhanh hơn (giảm bớt code để chạy), bạn sẽ có nhiều bộ nhớ (những phần kernel thì không bao giờ được trao đổi với vùng nhớ ảo), nhiều sự ổn định (luôn thăm dò cho một card không tồn tại) và những phần không cần thiết có thể được sử dụng bởi các tin tặc như công cụ xâm nhập server hoặc những máy khác trong mạng của bạn. Nhiều môđun chương trình (module) cũng có thể chậm hơn việc hỗ trợ được biên dịch một cách trực tiếp trong kernel.

Trong việc cấu hình và biên dịch, chúng ta sẽ biên dịch một Monolithic kernel . Muốn có Monolithic kernel chúng ta chỉ trả lời câu hỏi Yes hoặc No (không làm bất kỳ module nào khác) và bỏ qua những bước make modules và make modules_install. Ngoài ra chúng ta sẽ thêm vào kernel mới của chúng ta với sự bảo vệ tràn bộ đệm từ những phần bổ xung kernel. Những phần bổ sung cho kernel đang có sẵn, giống như phần bổ xung không thực thi ngăn sếp, nó không cho phép thực thi những đoạn chương trình trong ngăn xếp nên sẽ gây khó khăn trong việc tấn công gây tràn bộ đệm và làm thất bại hoàn toàn một số kiểu khai thác hiện tại được sử dụng trong các script trên thế giới

Nhớ là chỉ trả lời câu hỏi Yes hoặc No khi cấu hình kernel mới của bạn nếu bạn đang có ý định xây dựng một monolithic kernel. Nếu bạn dự định sử dụng chức năng firewall masquerading hoặc sự kết nối dial-up PPP bạn không có thể xây dựng một monolithic kernel, bởi mặc định những chức năng này yêu cầu xây dựng một vài module.

Một kernel mới liên quan rất cụ thể đến phần cứng máy tính của bạn. Trong phần cấu hình kernel chúng ta giả sử rằng máy tính của chúng ta có những hardware như được liệt kê dưới đây, tất nhiên bạn phải thay đổi chúng sao cho phù hợp với cấu hình máy bạn có.

1 Pentium II 400 MHz (i686) processor.

1 Motherboard SCSI.

1 Hard Disk SCSI.

1 SCSI Controler Adaptec AIC 7xxx.

1 Floppy Disk.

2 Ethernet Cards Intel EtherExpressPro 10/100.

1 Mouse PS/2.

Các hướng dẫn cài đặt này giả sử rằng

Các lệnh tương thích với unix

Đường dẫn nguồn là /usr/src

Sự hướng dẫn cài đặt dưới đây đã được kiểm tra với Red Hat Linux 6.1 và 6.2.

Tất cả các bước cài đặt được thực hiện bởi user root.

Version của kernel là 2.2.14 và version của Secure Linux Kernel là 2_2_14_ow2.

Packages

Kernel Homepage: http://www.kernelnotes.org.

Bạn phải đảm bảo là download tập tin linux-2_2_14.tar.gz.

Kernel FTP Site: 139.142.90.113.

Secure Linux Kernel Patches Homepage: http://www.openwall.com/linux/

Bạn phải đảm bảo là download tập tin linux-2_2_14.tar.gz.

Secure Linux Kernel Patches FTP Site: 195.42.162.180.

Làm một đĩa mềm boot cấp cứu

Trước khi install kernel mới bạn nên tạo một đĩa mềm cấp cứu. Linux có một tiện ích nhỏ để làm việc việc này với tên là "mkbootdisk". Bước đầu tiên là tìm ra version của kernel hiện tại bạn đang dùng là gì. Kiểm tra tập tin "/etc/lilo.conf" xem image nào đã được chọn để boot và từ image này chúng ta có thể tìm version của kernel chúng ta cần làm đĩa mềm cấp cứu. Trong ví dụ dưới đây tôi có tập tin lilo.conf với nội dung như sau.

```
[root@deep /]# cat /etc/lilo.conf
boot=/dev/sda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
image=/boot/vmlinuz-2.2.12-20   ← xem version của kernel
    label=linux                 ← xem image boot mặc định
    root=/dev/sda6
```

```
initrd=/boot/initrd-2.2.12-20.img
read-only
```

Bây giờ bạn sẽ cần tìm image được dùng để boot. Trên một tiến trình cài đặt chuẩn lần đầu label sẽ có tên là linux. Trong ví dụ ở trên chúng ta thấy rằng máy của bạn được khởi động sử dụng version kernel nguyên thuỷ của hệ thống "/boot/vmlinuz-2.2.12-20". Bây giờ đơn giản là chúng ta đặt đĩa mềm (1.44MB) đã format vào trong hệ thống và thực thi theo lệnh sau với quyền của root.

```
[root@deep /]# mkbootdisk --device /dev/fd0 2.2.12-20
Insert a disk in /dev/fd0. Any information on the disk will be lost.
Press <Enter> to continue or ^C to abort:
```

Theo những chỉ dẫn trên, bây giờ bạn đã có một đĩa mềm khởi động với một kernel đang làm việc được nhận biết. Tôi đề nghị bạn boot lại máy với đĩa mềm vừa tạo để đảm bảo rằng đĩa mềm làm việc một cách chính xác.

Sự loại bỏ và tối ưu

Bước 1

Chúng ta phải copy tập tin nguồn của kernel tới thư mục "/usr/src" và chuyển tới thư mục này.

```
[root@deep /]# cp linux-version.tar.gz /usr/src/
[root@deep /]# cd /usr/src/
```

Nếu trước đây bạn đã cài đặt một linux kernel với một tập tin nguồn ở dạng .tar

Những bước này được yêu cầu chỉ khi trước đây bạn đã cài đặt một Linux kernel với tập tin nguồn dạng .tar. Nếu đây là lần cài đặt Linux kernel mới, đầu tiên, thay cho tháo gỡ những package kernel-headers-version.i386.rpm, kernel-version.i386.rpm trên hệ thống của bạn.

- Gỡ bỏ những liên kết tượng trưng linux với lệnh:

```
[root@deep src]# rm -rf linux
```

- Gỡ bỏ thư mục Linux kernel header với lệnh:

```
[root@deep src]# rm -rf linux-2.2.xx (xx là một số, ví dụ như 14)
```

- Gỡ bỏ thư mục Linux kernel modules với lệnh:

```
[root@deep src]# rm -rf /lib/modules/2.2.xx
```

Chú ý: *Việc remove những module kernel cũ chỉ được yêu cầu nếu trước đây bạn đã install một version của modularized kernel. Nếu thư mục module không tồn tại dưới thư mục /lib thì đó là vì version kernel cũ của bạn nó không là một modularized kernel.*

Nếu các package RPM nguyên thủy của kernel được cài đặt trên hệ thống của bạn

Nếu package RPM nguyên thủy của kernel được cài đặt trên hệ thống của bạn thay cho dạng lưu trữ tar, tức là bạn đã hoàn thành việc install hệ thống linux mới của bạn hoặc trước đó đã sử dụng package RPM để nâng cấp Linux của bạn thì sử dụng lệnh sau để tháo gỡ (tháo gỡ) linux kernel:

- Bạn có thể kiểm tra package RPM kernel đã được cài đặt trên hệ thống với lệnh:

```
[root@deep src]# rpm -qa | grep kernel
kernel-headers-2.2.xx.i386.rpm
kernel-2.2.xx.i386.rpm
```

- Tháo gỡ package kernel RPM với lệnh:

```
[root@deep src]# rpm -e --nodeps kernel-headers kernel
cannot remove /usr/src/linux-2.2.xx - directory not empty
cannot remove /lib/modules/2.2.xx - directory not empty
[root@deep src]# rm -rf /usr/src/linux-2.2.xx/
[root@deep src]# rm -rf /lib/modules/2.2.xx/
```

Trong những bước trên chúng ta remove bằng tay những thư mục rỗng "/usr/src/linux-2.2.xx" và /lib/modules/2.2.xx" sau sự tháo gỡ của các kernel RPM (chương trình tháo gỡ sẽ không hoàn thành việc remove những thư mục này).

Bước 2

Giải nén Linux kernel archive.

Bây giờ chúng ta giải nén dạng lưu trữ dạng tar của kernel và sau đó remove dạng lưu trữ tar từ hệ thống.

```
[root@deep src]# tar xzpf linux-version.tar.gz
[root@deep src]# rm -f linux-version.tar.gz
```

Bước 3:

Tăng thêm các nhiệm vụ (tối ưu).

Tăng thêm các tác vụ được phép (tối đa số tiến trình trên một user) bạn cần hiệu chỉnh tập tin "/usr/src/linux/include/linux/tasks.h" và thay đổi những tham số sau:

- vi +14 /usr/src/linux/include/linux/tasks.h và thay đổi các tham số

  NR_TASKS từ 512 tới 3072

  MIN_TASKS_LEFT_FOR_ROOT từ 4 tới 24

Chú ý: *Giá trị trong dòng NR_TASKS biểu thị số lượng tối đa của các tác vụ xử lý mà linux kernel sẽ chỉ định trên các user. Tăng tác vụ sẽ cho phép bạn xử lý nhiều sự kết nối từ các client trên server (Ví dụ một web server sẽ có thể phục vụ nhiều các kết nối client). Ngoài ra Linux được bảo vệ từ sự chỉ định của tất cả các tiến trình cho các user thường. Có một tham số dòng MIN_TASKS_LEFT_FOR_ROOT được dành riêng một*

cách đặt biệt cho super-user (root) mà bạn có thể thiết đặt số tiến trình dành riêng cho root (24 là một giá trị hợp lý)

Bước 4:

Tối ưu kernel.

Tối ưu Linux kernel phù hợp với cấu trúc và cờ hiệu tối ưu của một CPU cụ thể của bạn, bạn có thể edit tập tin "/usr/src/linux/Makefile" và thay đổi những tham số sau.

- Hiệu chỉnh tập tin Makefile (vi +18 /usr/src/linux/Makefile) và thay đổi dòng:

  HOSTCC =gcc

  Trở thành:

  HOSTCC =egcs

- Hiệu chỉnh tập tin Makefile (vi +25 /usr/src/linux/Makefile) và thay đổi dòng:

  CC =$(CROSS_COMPILE)gcc –D__KERNEL__ -I$(HPATH)

  Trở thành:

  CC =$(CROSS_COMPILE)egcs –D__KERNEL__ -I$(HPATH)

- Hiệu chỉnh tập tin Makefile (vi +90 /usr/src/linux/Makefile) và thay đổi dòng:

  CFLAGS = -Wall -Wstrict-prototypes -O2 -fomit-frame-pointer

  Trở thành:

  CFLAGS = -Wall -Wstrict-prototypes -O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -march=pentiumpro -fomit-frame-pointer -fno-exceptions

- Hiệu chỉnh tập tin Makefile (vi +19 /usr/src/linux/Makefile) và thay đổi dòng:

  HOSTCFLAGS =-Wall -Wstrict-prototypes -O2 -fomit-frame-pointer

  Trở thành:

  HOSTCFLAGS =-Wall -Wstrict-prototypes -O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -march=pentiumpro -fomit-frame-pointer -fno-exceptions

Những thay đổi này tùy thuộc vào những mánh khoé tối ưu linh hoạt có thể hoặc không có thể làm việc với tất cả các kernel. Nếu những cờ hiệu ở trên hoặc những tham số nào bạn chọn cho cấu trúc CPU của bạn làm cho máy của bạn không làm việc, đừng cố gắng và ép buộc nó làm việc. Vì có thể làm hệ thống của bạn không ổn định giống Microsoft Windows.

Bảo mật Kernel

Những phần bổ sung cho bảo mật Linux kernel từ dự án Openwall là một phương pháp tốt để ngăn chặn những kiểu tấn công như Stack Buffer Overflow và những cái khác. Phần bổ sung của Openwall là một sự tập hợp những đặc trưng liên quan đến bảo mật cho Linux Kernel, tất cả có thể định hình thông qua phần cấu hình gọi là "Security options" mà sẽ được thêm vào Linux kernel mới của bạn.

Những phần bổ sung này thay đổi theo từng version và có thể có những sửa lỗi khác nhau.

Những đặc trưng của phần bổ sung version linux-2_2_14-ow2_tar.gz là:

Không có thể thực thi trong vùng ngăn xếp.

Giới hạn các liên kết trong /tmp.

Giới hạn FIFOs trong /tmp.

Giới hạn /proc.

Đặc biệt xử lý fd 0, 1, và 2.

Làm cho có hiệu lực RLIMIT_NPROC trong execve(2).

Phá hủy các phân đoạn bộ nhớ chia sẻ không sử dụng.

Chú ý: *Khi ghép linux-2_2_14-ow2 vào kernel, một phần mới là "Security options" sẽ được thêm ở cuối của sự cấu hình kernel của bạn. Cho nhiều thông tin và sự diễn giải của của những đặc trưng khác nhau sẵn có trong patch này, xem tập tin README trong source của patch.*

Gắn patch vào Kernel

Thực hiện theo những bước sau:

```
[root@deep /]# cp linux-2_2_14-ow2_tar.gz /usr/src/
[root@deep /]# cd /usr/src/
[root@deep src]# tar xzpf linux.2_2_14-ow2_tar.gz
[root@deep src]# cd linux-2.2.14-ow2/
[root@deep linux-2.2.14-ow2]# mv linux-2.2.14-ow2.diff /usr/src/
[root@deep linux-2.2.14-ow2]# cd ..
[root@deep src]# patch -p0 < linux-2.2.14-ow2.diff
[root@deep src]# rm -rf linux-2.2.14-ow2
[root@deep src]# rm -f linux-2.2.14-ow2.diff
[root@deep src]# rm -f linux-2_2_14-ow2_tar.gz
```

Đầu tiên copy tập tin nguồn vào thư mục /usr/src, sau đó chuyển vào thư mục này và giải nén tập tin nguồn (linux-2_2_14ow2_tar.gz). Sau đó chuyển vào thư mục chứa source được giải nén, chuyển tập tin linux-2.2.14-ow2.diff (file chứa patch) vào thư mục /usr/src. Chuyển về lại thư mục /usr/src và kernel bạn bây giờ có thêm tập tin linux-2.2.14-ow2.diff, sau đó chúng ta xoá tất cả các tập tin liên quan đến patch.

Chú ý: *Tất cả những message bảo mật liên quan đến phần bổ sung linux-2.2.14-ow2, giống như phần không thực thi trong ngăn xếp sẽ được ghi lại trong logfile "/var/log/messages".*

Các bước của việc thêm patch vào kernel đã hoàn thành, phần còn lại của tiến trình cài đặt là bạn sẽ biên dịch kernel mới và sau đó là boot lại máy.

Sự biên dịch:

Điều rất quan trọng là đảm bảo rằng những thư mục con như "/usr/include/asm", "/usr/include/linux", và "/usr/include/scsi" chỉ là những liên kết tới chương trình nguồn kernel.

Bước 1:

Những thư mục con như "asm", "linux", và "scsi" là những liên kết tới những thư mục chính trong chương trình nguồn kernel cần cho cấu trúc Linux của chúng ta., ví dụ "/usr/src/linux/include/asm-i386" cho "asm".

- Gõ những lệnh sau đây trên terminal của bạn:

```
[root@deep src]# cd /usr/include/
[root@deep include]# rm -rf asm linux scsi
[root@deep include]# ln -s /usr/src/linux/include/asm-i386 asm
[root@deep include]# ln -s /usr/src/linux/include/linux linux
[root@deep include]# ln -s /usr/src/linux/include/scsi scsi
```

Đây là một phần rất quan trọng của sự cấu hình, chúng ta loại bỏ (remove) những thư mục "asm", "linux" và "scsi" dưới thư mục "/usr/include", sau đó xây dựng những liên kết mới trỏ tới các thư mục cùng tên dưới thư mục của version nguồn linux kernel mới. Thư mục "include" chứa đựng những tập tin header linux kernel cần thiết và các chương trình có thể biên dịch trên hệ thống.

Bước 2:

Đảm bảo rằng bạn không có những tập tin .o cũ và những sự phụ thuộc khác.

- Gõ những lệnh sau đây trên terminal của bạn:

```
[root@deep include]# cd /usr/src/linux/
[root@deep linux]# make mrproper
```

Chú ý: *Hai bước trên đơn giản chỉ là để xoá bất kỳ những gì có thể bỏ sót một cách tình cờ trong thư mục nguồn của kernel do những người, nhóm làm việc trước đó.*

Bây giờ bạn đã có một chương trình nguồn đúng cho việc xây dựng một kernel linux mới. Bạn có thể cấu hình Linux kernel theo một trong ba phương pháp.

Phương pháp đầu tiên là sử dụng lệnh make config, phương pháp này cung cấp cho bạn một giao tiếp ở dạng text cho việc trả lời tất cả những tùy chọn cấu hình, bạn được nhắc nhở cho tất cả các tùy chọn bạn cần cài đặt kernel.

Phương pháp thứ hai là sử dụng lệnh make menuconfig, phương pháp này cung cấp tất cả các tùy chọn ở dạng menu rất dễ sử dụng.

Phương pháp cuối cùng là là sử dụng lệnh make xconfig, phương pháp này cung cấp một giao tiếp đồ họa của tất cả các tùy chọn của kernel.

Bước 3:

Phần cấu hình trong chương này, bạn sẽ sử dụng lệnh make config bởi vì chúng ta không cài đặt chương trình giao diện đồ hoạ XFree86 trên kernel của chúng ta.

- Gõ những lệnh sau trên terminal để tải chương trình cấu hình kernel:

```
[root@deep /]# cd /usr/src/linux/ (Nếu bạn không ở trong thư mục này).
[root@deep linux]# make config
rm -f include/asm
( cd include ; ln -sf asm-i386 asm)
/bin/sh scripts/Configure arch/i386/config.in
#
# Using defaults found in arch/i386/defconfig
#
```

Cấu hình kernel

Lưu ý: *Trong phần cấu hình dưới đây, nếu bạn không thay đổi cấu hình nào thì coi như bạn đã chọn theo mặc định của hệ thống. Phần trả lời in hoa là được chọn theo mặc định.*

Code maturity level options

Prompt for development and/or incomplete code/drivers (CONFIG_EXPERIMENTAL) [N/y/?]

Processor type and features

Processor family (386, 486/Cx486, 586/K5/5x86/6x86, Pentium/K6/TSC, PPro/6x86MX) [PPro/6x86MX]

Maximum Physical Memory (1GB, 2GB) [1GB]

Math emulation (CONFIG_MATH_EMULATION) [N/y/?]

MTRR (Memory Type Range Register) support (CONFIG_MTRR) [N/y/?]

Symmetric multi-processing support (CONFIG_SMP) [Y/n/?] N

Loadable module support

Enable loadable module support (CONFIG_MODULES) [Y/n/?] N

General setup

Networking support (CONFIG_NET) [Y/n/?]

PCI support (CONFIG_PCI) [Y/n/?]

PCI access mode (BIOS, Direct, Any) [Any]

PCI quirks (CONFIG_PCI_QUIRKS) [Y/n/?] N

Backward-compatible /proc/pci (CONFIG_PCI_OLD_PROC) [Y/n/?] N

MCA support (CONFIG_MCA) [N/y/?]

SGI Visual Workstation support (CONFIG_VISWS) [N/y/?]

System V IPC (CONFIG_SYSVIPC) [Y/n/?]

BSD Process Accounting (CONFIG_BSD_PROCESS_ACCT) [N/y/?]

Sysctl support (CONFIG_SYSCTL) [Y/n/?]
Kernel support for a.out binaries (CONFIG_BINFMT_AOUT) [Y/n/?]
Kernel support for ELF binaries (CONFIG_BINFMT_ELF) [Y/n/?]
Kernel support for MISC binaries (CONFIG_BINFMT_MISC) [Y/n/?]
Parallel port support (CONFIG_PARPORT) [N/y/?]
Advanced Power Management BIOS support (CONFIG_APM) [N/y/?]

Plug and Play support

Plug and Play support (CONFIG_PNP) [N/y/?]

Block devices

Normal PC floppy disk support (CONFIG_BLK_DEV_FD) [Y/n/?]
Enhanced IDE/MFM/RLL disk/cdrom/tape/floppy support (CONFIG_BLK_DEV_IDE) [Y/n/?]
Use old disk-only driver on primary interface (CONFIG_BLK_DEV_HD_IDE) [N/y/?]
Include IDE/ATA-2 DISK support (CONFIG_BLK_DEV_IDEDISK) [Y/n/?]
Include IDE/ATAPI CDROM support (CONFIG_BLK_DEV_IDECD) [Y/n/?]
Include IDE/ATAPI TAPE support (CONFIG_BLK_DEV_IDETAPE) [N/y/?]
Include IDE/ATAPI FLOPPY support (CONFIG_BLK_DEV_IDEFLOPPY) [N/y/?]
SCSI emulation support (CONFIG_BLK_DEV_IDESCSI) [N/y/?]
CMD640 chipset bugfix/support (CONFIG_BLK_DEV_CMD640) [Y/n/?] N
RZ1000 chipset bugfix/support (CONFIG_BLK_DEV_RZ1000) [Y/n/?] N
Generic PCI IDE chipset support (CONFIG_BLK_DEV_IDEPCI) [Y/n/?]
Generic PCI bus-master DMA support (CONFIG_BLK_DEV_IDEDMA) [Y/n/?]
Boot off-board chipsets first support (CONFIG_BLK_DEV_OFFBOARD) [N/y/?]
Use DMA by default when available (CONFIG_IDEDMA_AUTO) [Y/n/?]
Other IDE chipset support (CONFIG_IDE_CHIPSETS) [N/y/?]
Loopback device support (CONFIG_BLK_DEV_LOOP) [N/y/?]
Network block device support (CONFIG_BLK_DEV_NBD) [N/y/?]
Multiple devices driver support (CONFIG_BLK_DEV_MD) [N/y/?]
RAM disk support (CONFIG_BLK_DEV_RAM) [N/y/?]
XT hard disk support (CONFIG_BLK_DEV_XD) [N/y/?]
Mylex DAC960/DAC1100 PCI RAID Controller support (CONFIG_BLK_DEV_DAC960) [N/y/?] (NEW)
Parallel port IDE device support (CONFIG_PARIDE) [N/y/?]
Compaq SMART2 support (CONFIG_BLK_CPQ_DA) [N/y/?] (NEW)

Networking options

Packet socket (CONFIG_PACKET) [Y/n/?]
Kernel/User netlink socket (CONFIG_NETLINK) [N/y/?]
Network firewalls (CONFIG_FIREWALL) [N/y/?] Y
Socket Filtering (CONFIG_FILTER) [N/y/?]
Unix domain sockets (CONFIG_UNIX) [Y/n/?]
TCP/IP networking (CONFIG_INET) [Y/n/?]

IP: multicasting (CONFIG_IP_MULTICAST) [N/y/?]

IP: advanced router (CONFIG_IP_ADVANCED_ROUTER) [N/y/?]

IP: kernel level autoconfiguration (CONFIG_IP_PNP) [N/y/?]

IP: firewalling (CONFIG_IP_FIREWALL) [N/y/?] (NEW) Y

IP: transparent proxy support (CONFIG_IP_TRANSPARENT_PROXY) [N/y/?] (NEW)

IP: masquerading (CONFIG_IP_MASQUERADE) [N/y/?] (NEW)

IP: optimize as router not host (CONFIG_IP_ROUTER) [N/y/?]

IP: tunneling (CONFIG_NET_IPIP) [N/y/?]

IP: GRE tunnels over IP (CONFIG_NET_IPGRE) [N/y/?]

IP: aliasing support (CONFIG_IP_ALIAS) [N/y/?]

IP: TCP syncookie support (not enabled per default) (CONFIG_SYN_COOKIES) [N/y/?] Y

IP: Reverse ARP (CONFIG_INET_RARP) [N/y/?]

IP: Allow large windows (not recommended if <16Mb of memory) (CONFIG_SKB_LARGE) [Y/n/?]

The IPX protocol (CONFIG_IPX) [N/y/?]

Appletalk DDP (CONFIG_ATALK) [N/y/?]

Telephony support

Linux telephony support (CONFIG_PHONE) [N/y/?] (NEW)

SCSI support

SCSI support (CONFIG_SCSI) [Y/n/?]

SCSI disk support (CONFIG_BLK_DEV_SD) [Y/n/?]

SCSI tape support (CONFIG_CHR_DEV_ST) [N/y/?]

SCSI CD-ROM support (CONFIG_BLK_DEV_SR) [N/y/?]

SCSI generic support (CONFIG_CHR_DEV_SG) [N/y/?]

Probe all LUNs on each SCSI device (CONFIG_SCSI_MULTI_LUN) [Y/n/?] N

Verbose SCSI error reporting (kernel size +=12K) (CONFIG_SCSI_CONSTANTS) [Y/n/?] N

SCSI logging facility (CONFIG_SCSI_LOGGING) [N/y/?]

SCSI low-level drivers

7000FASST SCSI support (CONFIG_SCSI_7000FASST) [N/y/?]

ACARD SCSI support (CONFIG_SCSI_ACARD) [N/y/?]

Adaptec AHA152X/2825 support (CONFIG_SCSI_AHA152X) [N/y/?]

Adaptec AHA1542 support (CONFIG_SCSI_AHA1542) [N/y/?]

Adaptec AHA1740 support (CONFIG_SCSI_AHA1740) [N/y/?]

Adaptec AIC7xxx support (CONFIG_SCSI_AIC7XXX) [N/y/?] Y

Enable Tagged Command Queueing (TCQ) by default [N/y/?] (NEW) Y

Maximum number of TCQ commands per device (CONFIG_AIC7XXX_CMDS_PER_DEVICE) [8] (NEW)

Collect statistics to report in /proc (CONFIG_AIC7XXX_PROC_STATS) [N/y/?] (NEW)

Delay in seconds after SCSI bus reset (CONFIG_AIC7XXX_RESET_DELAY) [5] (NEW)

IBM ServeRAID support (CONFIG_SCSI_IPS) [N/y/?] (NEW)

AdvanSys SCSI support (CONFIG_SCSI_ADVANSYS) [N/y/?]

Always IN2000 SCSI support (CONFIG_SCSI_IN2000) [N/y/?]

AM53/79C974 PCI SCSI support (CONFIG_SCSI_AM53C974) [N/y/?]

AMI MegaRAID support (CONFIG_SCSI_MEGARAID) [N/y/?]

BusLogic SCSI support (CONFIG_SCSI_BUSLOGIC) [N/y/?]

DTC3180/3280 SCSI support (CONFIG_SCSI_DTC3280) [N/y/?]

EATA ISA/EISA/PCI (DPT and generic EATA/DMA) support (CONFIG_SCSI_EATA) [N/y/?]

EATA-DMA [Obsolete] (DPT, NEC, AT&T, SNI, AST, Olivetti, Alphatronix) support (CONFIG_SCSI_EATA_DMA) [N/y/?]

EATA-PIO (old DPT PM2001, PM2012A) support (CONFIG_SCSI_EATA_PIO) [N/y/?]

Future Domain 16xx SCSI/AHA-2920A support (CONFIG_SCSI_FUTURE_DOMAIN) [N/y/?]

GDT SCSI Disk Array Controller support (CONFIG_SCSI_GDTH) [N/y/?]

Generic NCR5380/53c400 SCSI support (CONFIG_SCSI_GENERIC_NCR5380) [N/y/?]

Initio 9100U(W) support (CONFIG_SCSI_INITIO) [N/y/?]

Initio INI-A100U2W support (CONFIG_SCSI_INIA100) [N/y/?]

NCR53c406a SCSI support (CONFIG_SCSI_NCR53C406A) [N/y/?]

symbios 53c416 SCSI support (CONFIG_SCSI_SYM53C416) [N/y/?]

Simple 53c710 SCSI support (Compaq, NCR machines) (CONFIG_SCSI_SIM710) [N/y/?] (NEW)

NCR53c7,8xx SCSI support (CONFIG_SCSI_NCR53C7xx) [N/y/?]

NCR53C8XX SCSI support (CONFIG_SCSI_NCR53C8XX) [N/y/?]

SYM53C8XX SCSI support (CONFIG_SCSI_SYM53C8XX) [Y/n/?] N

PAS16 SCSI support (CONFIG_SCSI_PAS16) [N/y/?]

PCI2000 support (CONFIG_SCSI_PCI2000) [N/y/?]

PCI2220i support (CONFIG_SCSI_PCI2220I) [N/y/?]

PSI240i support (CONFIG_SCSI_PSI240I) [N/y/?]

Qlogic FAS SCSI support (CONFIG_SCSI_QLOGIC_FAS) [N/y/?]

Qlogic ISP SCSI support (CONFIG_SCSI_QLOGIC_ISP) [N/y/?]

Qlogic ISP FC SCSI support (CONFIG_SCSI_QLOGIC_FC) [N/y/?]

Seagate ST-02 and Future Domain TMC-8xx SCSI support (CONFIG_SCSI_SEAGATE) [N/y/?]

Tekram DC390(T) and Am53/79C974 SCSI support (CONFIG_SCSI_DC390T) [N/y/?]

Trantor T128/T128F/T228 SCSI support (CONFIG_SCSI_T128) [N/y/?]

UltraStor 14F/34F support (CONFIG_SCSI_U14_34F) [N/y/?]

UltraStor SCSI support (CONFIG_SCSI_ULTRASTOR) [N/y/?]

Network device support

Network device support (CONFIG_NETDEVICES) [Y/n/?]

ARCnet devices

ARCnet support (CONFIG_ARCNET) [N/y/?]

Dummy net driver support (CONFIG_DUMMY) [Y/n/?]
EQL (serial line load balancing) support (CONFIG_EQUALIZER) [N/y/?]
General Instruments Surfboard 1000 (CONFIG_NET_SB1000) [N/y/?] (NEW)

Ethernet (10 or 100Mbit)

Ethernet (10 or 100Mbit) (CONFIG_NET_ETHERNET) [Y/n/?]
3COM cards (CONFIG_NET_VENDOR_3COM) [N/y/?]
AMD LANCE and PCnet (AT1500 and NE2100) support (CONFIG_LANCE) [N/y/?]
Western Digital/SMC cards (CONFIG_NET_VENDOR_SMC) [N/y/?]
Racal-Interlan (Micom) NI cards (CONFIG_NET_VENDOR_RACAL) [N/y/?]
Other ISA cards (CONFIG_NET_ISA) [N/y/?]
EISA, VLB, PCI and on board controllers (CONFIG_NET_EISA) [Y/n/?]
AMD PCnet32 (VLB and PCI) support (CONFIG_PCNET32) [N/y/?]
Apricot Xen-II on board Ethernet (CONFIG_APRICOT) [N/y/?]
CS89x0 support (CONFIG_CS89x0) [N/y/?]
DM9102 PCI Fast Ethernet Adapter support (EXPERIMENTAL) (CONFIG_DM9102) [N/y/?] (NEW)
Generic DECchip & DIGITAL EtherWORKS PCI/EISA (CONFIG_DE4X5) [N/y/?]
DECchip Tulip (dc21x4x) PCI support (CONFIG_DEC_ELCP) [N/y/?]
Old DECchip Tulip (dc21x4x) PCI support (CONFIG_DEC_ELCP_OLD) [N/y/?] (NEW)
Digi Intl. RightSwitch SE-X support (CONFIG_DGRS) [N/y/?]
EtherExpressPro/100 support (CONFIG_EEXPRESS_PRO100) [Y/n/?]
PCI NE2000 support (CONFIG_NE2K_PCI) [N/y/?]
TI ThunderLAN support (CONFIG_TLAN) [N/y/?]
VIA Rhine support (CONFIG_VIA_RHINE) [N/y/?]
SiS 900/7016 PCI Fast Ethernet Adapter support (CONFIG_SIS900) [N/y/?] (NEW)
Pocket and portable adaptors (CONFIG_NET_POCKET) [N/y/?]

Ethernet (1000 Mbit)

SysKonnect SK-98xx support (CONFIG_SK98LIN) [N/y/?] (NEW)
FDDI driver support (CONFIG_FDDI) [N/y/?]
PPP (point-to-point) support (CONFIG_PPP) [N/y/?]
SLIP (serial line) support (CONFIG_SLIP) [N/y/?]
Wireless LAN (non-hamradio) (CONFIG_NET_RADIO) [N/y/?]

Token ring devices

Token Ring driver support (CONFIG_TR) [N/y/?]
Fibre Channel driver support (CONFIG_NET_FC) [N/y/?] (NEW)

Wan interfaces

MultiGate (COMX) synchronous serial boards support (CONFIG_COMX) [N/y/?] (NEW)
Frame relay DLCI support (CONFIG_DLCI) [N/y/?]
WAN drivers (CONFIG_WAN_DRIVERS) [N/y/?]
SBNI12-xx support (CONFIG_SBNI) [N/y/?] (NEW)

Amateur Radio support

Amateur Radio support (CONFIG_HAMRADIO) [N/y/?]

IrDA subsystem support

IrDA subsystem support (CONFIG_IRDA) [N/y/?]

ISDN subsystem

ISDN support (CONFIG_ISDN) [N/y/?]

Old CD-ROM drivers (not SCSI, not IDE)

Support non-SCSI/IDE/ATAPI CDROM drives (CONFIG_CD_NO_IDESCSI) [N/y/?]

Character devices

Virtual terminal (CONFIG_VT) [Y/n/?]

Support for console on virtual terminal (CONFIG_VT_CONSOLE) [Y/n/?]

Standard/generic (dumb) serial support (CONFIG_SERIAL) [Y/n/?]

Support for console on serial port (CONFIG_SERIAL_CONSOLE) [N/y/?]

Extended dumb serial driver options (CONFIG_SERIAL_EXTENDED) [N/y/?]

Non-standard serial port support (CONFIG_SERIAL_NONSTANDARD) [N/y/?]

Unix98 PTY support (CONFIG_UNIX98_PTYS) [Y/n/?]

Maximum number of Unix98 PTYs in use (0-2048) (CONFIG_UNIX98_PTY_COUNT) [256] 128

Mouse Support (not serial mice) (CONFIG_MOUSE) [Y/n/?]

Mice

ATIXL busmouse support (CONFIG_ATIXL_BUSMOUSE) [N/y/?]

Logitech busmouse support (CONFIG_BUSMOUSE) [N/y/?]

Microsoft busmouse support (CONFIG_MS_BUSMOUSE) [N/y/?]

PS/2 mouse (aka "auxiliary device") support (CONFIG_PSMOUSE) [Y/n/?]

C&T 82C710 mouse port support (as on TI Travelmate) (CONFIG_82C710_MOUSE) [Y/n/?] N

PC110 digitizer pad support (CONFIG_PC110_PAD) [N/y/?]

Joystick support

Joystick support (CONFIG_JOYSTICK) [N/y/?]

QIC-02 tape support (CONFIG_QIC02_TAPE) [N/y/?]

Watchdog Timer Support (CONFIG_WATCHDOG) [N/y/?]

/dev/nvram support (CONFIG_NVRAM) [N/y/?]

Enhanced Real Time Clock Support (CONFIG_RTC) [N/y/?]

Video for Linux

Video For Linux (CONFIG_VIDEO_DEV) [N/y/?]

Double Talk PC internal speech card support (CONFIG_DTLK) [N/y/?]

Ftape, the floppy tape device driver

Ftape (QIC-80/Travan) support (CONFIG_FTAPE) [N/y/?]

Filesystems

Quota support (CONFIG_QUOTA) [N/y/?] Y
Kernel automounter support (CONFIG_AUTOFS_FS) [Y/n/?] N
Amiga FFS filesystem support (CONFIG_AFFS_FS) [N/y/?]
Apple Macintosh filesystem support (experimental) (CONFIG_HFS_FS) [N/y/?]
DOS FAT fs support (CONFIG_FAT_FS) [N/y/?]
ISO 9660 CDROM filesystem support (CONFIG_ISO9660_FS) [Y/n/?]
Microsoft Joliet CDROM extensions (CONFIG_JOLIET) [N/y/?]
Minix fs support (CONFIG_MINIX_FS) [N/y/?]
NTFS filesystem support (read only) (CONFIG_NTFS_FS) [N/y/?]
OS/2 HPFS filesystem support (read only) (CONFIG_HPFS_FS) [N/y/?]
/proc filesystem support (CONFIG_PROC_FS) [Y/n/?]
/dev/pts filesystem for Unix98 PTYs (CONFIG_DEVPTS_FS) [Y/n/?]
ROM filesystem support (CONFIG_ROMFS_FS) [N/y/?]
Second extended fs support (CONFIG_EXT2_FS) [Y/n/?]
System V and Coherent filesystem support (CONFIG_SYSV_FS) [N/y/?]
UFS filesystem support (CONFIG_UFS_FS) [N/y/?]

Network tập tin Systems

Coda filesystem support (advanced network fs) (CONFIG_CODA_FS) [N/y/?]
NFS filesystem support (CONFIG_NFS_FS) [Y/n/?] N
SMB filesystem support (to mount WfW shares etc.) (CONFIG_SMB_FS) [N/y/?]
NCP filesystem support (to mount NetWare volumes) (CONFIG_NCP_FS) [N/y/?]

Partition Types

BSD disklabel (BSD partition tables) support (CONFIG_BSD_DISKLABEL) [N/y/?]
Macintosh partition map support (CONFIG_MAC_PARTITION) [N/y/?]
SMD disklabel (Sun partition tables) support (CONFIG_SMD_DISKLABEL) [N/y/?]
Solaris (x86) partition table support (CONFIG_SOLARIS_X86_PARTITION) [N/y/?]

Console drivers

VGA text console (CONFIG_VGA_CONSOLE) [Y/n/?]
Video mode selection support (CONFIG_VIDEO_SELECT) [N/y/?]

Sound

Sound card support (CONFIG_SOUND) [N/y/?]

(Security options will appear only if you are patched your kernel with the Openwall Project patch).

Security options

Non-executable user stack area (CONFIG_SECURE_STACK) [Y]
Autodetect and emulate GCC trampolines (CONFIG_SECURE_STACK_SMART) [Y]
Restricted links in /tmp (CONFIG_SECURE_LINK) [Y]
Restricted FIFOs in /tmp (CONFIG_SECURE_FIFO) [Y]
Restricted /proc (CONFIG_SECURE_PROC) [N] Y

Special handling of fd 0, 1, and 2 (CONFIG_SECURE_FD_0_1_2) [Y]
Enforce RLIMIT_NPROC on execve(2) (CONFIG_SECURE_RLIMIT_NPROC) [Y]
Destroy shared memory segments not in use (CONFIG_SECURE_SHM) [N] Y

Kernel hacking
Magic SysRq key (CONFIG_MAGIC_SYSRQ) [N/y/?]

Bây giờ trở lại thư mục "/usr/src/linux" (nếu bạn không có ở trong đó). Bạn cần biên dịch kernel mới, bạn làm việc này bằng cách dùng những lệnh sau:

```
[root@deep linux]# make dep; make clean; make bzImage
```

Câu lệnh trên chứa đựng ba lệnh trong cùng một dòng, lệnh đầu tiên nắm bắt cấu hình và xây dựng cây thư mục phụ thuộc tương ứng. Quá trình này sẽ xác định được thành phần nào được biên dịch và những thành phần nào không biên dịch. Lệnh kế tiếp make clean xoá bỏ tất cả những dấu vết của sự biên dịch trước đây và cũng tránh bất kỳ những nhầm lẫn trong các version của một đặc trưng dùng trong kernel này. Cuối cùng make bzImage sẽ biên dịch đầy đủ kernel

Sau khi tiến trình hoàn thành, kernel được nén và sẵn sàng để cài đặt trên hệ thống của bạn. Trước khi cài đặt kernel mới, chúng ta phải biết chúng ta cần biên dịch những module tương ứng. Việc này chỉ yêu cầu nếu chúng ta trả lời Yes trong tùy chọn "Enable loadable module support (CONFIG_MODULES)" và biên dịch vài tùy chọn trong cấu hình kernel ở trên như một module, nếu vậy bạn phải thực thi lệnh sau:

- Biên dịch những module tương ứng cho kernel của bạn sử dụng những lệnh dưới đây

```
[root@deep linux]# make modules
[root@deep linux]# make modules_install
```

Chú ý: *Những lệnh make modules và make modules_install chỉ yêu cầu nếu bạn chọn YES trong tùy chọn "Enable loadable module support (CONFIG_MODULES)" trong cấu hình kernel ở trên.*

Cài đặt kernel mới:

1. Copy tập tin /usr/src/linux/arch/i386/boot/bzImage đã được sinh ra trong quá trình biên dịch tới thư mục /boot và đặt cho chúng một tên riêng mới.

```
[root@deep linux]# cp /usr/src/linux/arch/i386/boot/bzImage /boot/vmlinuz-kernel.version.number
```

Chú ý: *Một tên mới thích hợp là một tên mà chúng ta đặt giống như vmlinuz-2.2.14. Việc này rất quan trọng nếu bạn muốn tạo một đĩa mềm cấp cứu mới sử dụng chương trình mkbootdisk, việc này yêu cầu một vài điều cần cụ thể, ví dụ vmlinuz-2.2.14 thay cho vmlinuz-2.2.14.a*

2. Copy tập tin System.map từ thư mục /usr/src/linux tới thư mục /boot và cũng đặt cho nó một tên mới thích hợp.

```
[root@deep linux]# cp /usr/src/linux/System.map /boot/System.map-
kernel.version.number
```

3. Chuyển vào thư mục /boot và xây dựng lại những liên kết vmlinuz và System.map với những lệnh sau:

```
[root@deep linux]# cd /boot
[root@deep /boot]# ln -fs vmlinuz-kernel.version.number vmlinuz
[root@deep /boot]# ln -fs System.map-kernel.version.number System.map
```

Chúng ta phải xây dựng lại những liên kết của "vmlinuz" và "System.map" trỏ chúng tới version kernel mới được cài đặt. Không có những liên kết mới này chương trình LILO sẽ mặc định trỏ đến version cũ của Linux kernel

4. Remove những tập tin cũ và không cần thiết dưới thư mục /boot.

```
[root@deep /boot]# rm -f module-info
[root@deep /boot]# rm -f initrd-2.2.xx.img
```

Liên kết "module-info" trỏ tới thư mục những module cũ của kernel ban đầu. Kể từ khi chúng ta cài đặt một kernel mới, chúng ta không cần giữ những liên kết bị bé gãy này. "initrd-2.2.xx" là một tập tin, nó chứa đựng một ảnh RAM disk ban đầu đáp ứng như một hệ thống trước khi disk có hiệu lực. Tập tin này chỉ có hiệu lực và được cài đặt từ tiến trình cài đặt Linux nếu hệ thống của bạn có hiện diện một SCSI card. Nếu chúng ta có một hệ thống SCSI, các driver sẽ được kết hợp chặt chẽ vào trong Linux kernel mới kể từ khi chúng ta xây dựng một monolithic kernel, vì thế chúng ta có thể remove tập tin này (initrd-2.2.xx.imp) một cách an toàn.

5. Tạo một thư mục Linux kernel mới, thư mục này sẽ quản lý tất cả các tập tin header liên quan tới Linux kernel cho những sự biên dịch tương lai của những chương trình khác trên hệ thống của bạn.

Nhớ lại chúng ta đã tạo ba liên kết dưới thư mục /usr/include, ba liên kết này trỏ tới Linux kernel và có thể biên dịch chúng mà không nhận bất kỳ lỗi nào và để có thể biên dịch những chương trình tương lai. /usr/include là thư mục mà ở đó tất cả các tập tin header của hệ thống Linux của bạn được giữ cho việc tham chiếu và những phần phụ thuộc khi bạn biên dịch và cài đặt những chương trình mới. Những liên kết asm, linux và scsi được sử dụng khi chương trình cần biết một vài tính năng từ thời gian biên dịch cụ thể kernel được cài đặt trên hệ thống của bạn. Những chương trình gọi những tập tin header khác nhau trong thư mục include khi chúng phải biết những thông tin cụ thể, sự phụ thuộc... của hệ thống của bạn.

```
[root@deep /]# mkdir -p /usr/src/linux-2.2.14/include
[root@deep /]# cp -r /usr/src/linux/include/asm-generic /usr/src/linux-2.2.14/include
[root@deep /]# cp -r /usr/src/linux/include/asm-i386 /usr/src/linux-2.2.14/include
[root@deep /]# cp -r /usr/src/linux/include/linux /usr/src/linux-2.2.14/include
[root@deep /]# cp -r /usr/src/linux/include/net /usr/src/linux-2.2.14/include
[root@deep /]# cp -r /usr/src/linux/include/video /usr/src/linux-2.2.14/include
```

```
[root@deep /]# cp -r /usr/src/linux/include/scsi /usr/src/linux-2.2.14/include
[root@deep /]# rm -rf /usr/src/linux
[root@deep /]# cd /usr/src
[root@deep src]# ln -s /usr/src/linux-2.2.14 linux
```

Đầu tiên chúng ta tạo một thư mục mới được đặt tên là "linux-2.2.14" giống như version của kernel chúng ta đã cài đặt, sau đó chúng ta copy những thư mục như asm-generic, asm-i386, linux, net, và scsi từ thư mục "/usr/linux/include" tới vị trí mới "/usr/src/linux-2.2.14/include". Sau khi chúng ta remove toàn bộ thư mục nguồn nơi mà chúng ta đã biên dịch kernel mới, chúng ta tạo một liên kết mới được đặt tên là linux dưới thư mục /usr/src trỏ tới thư mục mới "/usr/src/linux-2.2.14/include. Với những bước trên những chương trình được biên dịch trong tương lai sẽ biết được ở đâu có thể tìm kiếm những tập tin header liên quan đến kernel trên server của chúng ta.

Chú ý: *Bước này sẽ cho phép chúng ta thu được những khoảng trống đáng kể trên hard disk và cũng giảm bớt nguy hiểm trong bảo mật. Thư mục nguồn Linux kernel quản lý nhiều tập tin và chiếm kích thước khoảng 75MB khi được giải nén. Với qui trình được diễn giải ở trên thư mục Linux kernel của chúng ta kích thước khoảng chừng 3MB, vì vậy chúng ta tiết kiệm 72MB nhưng vẫn đáp ứng được cùng chức năng.*

6. Cuối cùng, bạn cần edit tập tin /etc/lilo.conf để chọn image của kernel mới và thay đổi tham số như thời gian boot.

Bước 1:

Edit tập tin lilo.conf (vi /etc/lilo.conf) và làm những thay đổi thích hợp trên dòng có nội dung "image=/boot"

```
[root@deep /]# vi /etc/lilo.conf
boot=/dev/sda
map=/boot/map
install=/boot/boot.b
prompt
timeout=00
restricted
password=somepasswd
image=/boot/vmlinuz-kernel.version.number #(đưa vào tên tập tin kernel mới ở đây).
label=linux
root=/dev/sda6
read-only
```

Chú ý: *Đừng quên remove dòng "initrd=/boot/initrd-2.2.12-20.img" trong tập tin lilo.conf, kể từ bây giờ dòng này không cần thiết (monolithic kernel không cần tập tin initrd).*

Bước 2:

Một khi tên của version kernel mới được đặt trong tập tin lilo.conf như được trình bày ở trên, chúng ta cập nhật tập tin "lilo.conf" cho những thay đổi có hiệu lực với những lệnh sau:

```
[root@deep /]# /sbin/lilo -v
LILO version 21, [Copyright 1992-1998 Werner Almesberger
Reading boot sector from /dev/sda
Merging with /boot/boot.b
Boot image: /boot/vmlinuz-2.2.14
Added linux *
/boot/boot.0800 exits – no backup copy made.
Writing boot sector.
```

Một chú ý quan trọng là nếu bạn trả lời NO trong tùy chọn "Unix98 PTY support (CONFIG_UNIX98_PTYS)" trong lúc cấu hình kernel của bạn, bạn phải edit tập tin /etc/fstab và remove dòng:

```
none   /dev/pts devpts   gid=5,mode=620     0 0
```

Xóa những chương trình, tập tin, dòng có liên quan tới các module

Mặc định khi bạn cài đặt Red Hat Linux trong lần đầu tiên (chưa biên dịch lại kernel), kernel được xây dựng như một modularized kernel. Điều này có ý nghĩa rằng mỗi thiết bị, hoặc mỗi chức năng mà chúng ta cần hiện có như các module sẽ được điều khiển bởi chương trình Kernel Deamon tên là kmod, chương trình này tải một cách tự động một vài module và chức năng hỗ trợ vào trong bộ nhớ khi nó cần và loại bỏ khỏi bộ nhớ nếu không được sử dụng trong một thời gian dài.

Bước 1:

kmod và các chương trình quản lý module khác kể cả trong package RPM "modutils" sử dụng tập tin conf.modules ở trong thư mục /etc, tập tin này cho biết bạn đang sử dụng loại card mạng nào và những cấu hình đặc biệt của nó. Bởi vì chúng ta không sử dụng bất kỳ những module nào trong kernel mới vừa được compile, chúng ta có thể gỡ bỏ (remove) tập tin "conf.modules" và tháo gỡ (tháo gỡ) hoàn toàn package "modutils".

- Remove tập tin "conf.modules" sử dụng lệnh:

```
[root@deep /]# rm -f /etc/conf.modules
```

- Unintall package modutils sử dụng lệnh:

```
[root@deep /]# rpm -e --nodeps modutils
```

Bước 2:

Một điều cuối cùng cần làm là hiệu chỉnh tập tin "rc.sysinit" và ghi chú tất cả các dòng có quan hệ tới "depmod –a" bằng cách đặt dấu "#" ở đầu của mỗi

dòng. Công việc này rất cần vì mặc định ở thời điểm khởi động, hệ thống đọc tập tin rc.sysinit để tìm phần phụ thuộc module trong kernel.

Dưới Red Hat Linux 6.1

Ghi chú: *Dòng thứ 260 trong tập tin rc.sysinit như sau: (vi +260 /etc/rc.d/rc.sysinit):*

```
if [ -x /sbin/depmod -a -n "$USEMODULES" ]; then
```

Trở thành:

```
#if [ -x /sbin/depmod -a -n "$USEMODULES" ]; then
```

Ghi chú: *Các dòng từ 272 đến 277 trong tập tin rc.sysinit như sau: (vi +272 /etc/rc.d/rc.sysinit)*

```
if [ -L /lib/modules/default ]; then
    INITLOG_ARGS= action "Finding module dependencies" depmod -a default
    else
    INITLOG_ARGS= action "Finding module dependencies" depmod -a
    fi
```

Trở thành:

```
# if [ -L /lib/modules/default ]; then
# INITLOG_ARGS= action "Finding module dependencies" depmod -a default
# else
# INITLOG_ARGS= action "Finding module dependencies" depmod -a
# fi
```

Chú ý: *Thủ tục được diễn giải ở trên liên quan tới package initscripts-4_70-1 trong Red Hat Linux version 6.1*

Dưới Red Hat Linux 6.2

Ghi chú: *Dòng thứ 243 trong tập tin rc.sysinit như sau: (vi +243 /etc/rc.d/rc.sysinit):*

```
if [ -x /sbin/depmod -a -n "$USEMODULES" ]; then
```

Trở thành:

```
#if [ -x /sbin/depmod -a -n "$USEMODULES" ]; then
```

Ghi chú: *Các dòng từ 255 đến 260 trong tập tin rc.sysinit như sau: (vi +255 /etc/rc.d/rc.sysinit):*

```
    if [ -L /lib/modules/default ]; then
INITLOG_ARGS= action "Finding module dependencies" depmod -a default
    else
INITLOG_ARGS= action "Finding module dependencies" depmod -a
    fi
```

Trở thành:

```
# if [ -L /lib/modules/default ]; then
```

```
# INITLOG_ARGS= action "Finding module dependencies" depmod -a default
# else
# INITLOG_ARGS= action "Finding module dependencies" depmod -a
# fi
```

Chú ý: *Xin nhắc lại lần nữa, tất cả phần này (xoá các chương trình, tập tin, và những dòng liên quan tới các module) chỉ yêu cầu nếu như bạn trả lời No trong tùy chọn "Enable loadable module support (CONFIG_MODULES)" trong cấu hình kernel ở trên.*

Bước 3:

Bây giờ bạn khởi động lại máy và kiểm tra lại kết quả của bạn.

[root@deep /]# reboot (lệnh khởi động lại máy)

Khi hệ thống đã được khởi động và bạn đã login vào, kiểm tra version mới của kernel với lệnh:

- [root@deep /]# uname -a

  Linux deep.openna.com 2.2.14 #1 Mon Jan 10 10:40:35 EDT 2000 i686 unknown

  [root@deep]#

Đến đây việc compile kernel mới của bạn đã thành công

Tạo một đĩa cứu nguy mới (rescue floppy):

Sau khi reboot lại Linux server, bạn đã có một hệ thống với một kernel mới vừa được cập nhật, công việc tiếp tục của bạn là tạo một đĩa mềm khởi động cho hệ thống với kernel mới này để phòng ngừa sự cố cho tương lai khi hệ thống không thể boot được. Để làm việc này bạn theo những bước sau:

- Login với root, đưa đĩa mềm vào ổ đĩa A, thực thi lệnh sau:

  [root@deep /]# mkbootdisk --device /dev/fd0 2.2.14

  Insert a disk in /dev/fd0. Any information on the disk will be lost.

  Press <Enter> to continue or ^C to abort:

Chú ý: *Chương trình mkbootdisk chỉ chạy trên modularized kernel, như vậy bạn không thể sử dụng nó trên một monolithic kernel. Để thay thế việc này, tạo một đĩa mềm cấp cứu như được trình bày dưới đây nếu bạn có một vấn đề với hệ thống của bạn trong tương lai.*

Tạo đĩa mềm boot cấp cứu (emergency boot floppy disk)

Như đã nói ở trên chúng ta chỉ có thể tạo một rescue floppy trên modularized kernel, chúng ta phải tìm con đường khác để có thể khởi động hệ thống nếu Linux kernel trên đĩa cứng đã bị hỏng. Việc này có thể làm được với Linux emergency boot floppy disk. Bạn nên tạo đĩa mềm này ngay sau khi bạn khởi động thành công hệ thống của bạn và login với user root.

- Tạo một emergency boot floppy disk, theo ba bước:

1. Đưa đĩa mềm vào và format với lệnh:

```
[root@deep /]# fdformat /dev/fd0H1440
Double-sided, 80 tracks, 18 sec/track. Total capacity 1440 kB.
Formatting ... done
Verifying ... done
```

2. Copy tập tin "vmlinuz" từ thư mục /boot tới đĩa mềm:

```
[root@deep /]# cp /boot/vmlinuz  /dev/fd0
cp: overwrite `/dev/fd0'? y
```

tập tin "vmlinuz là một liên kết tượng trưng trỏ tới Kernel Linux thực (real Linux kernel).

3. Xác định thiết bị gốc của kernel với lệnh:

```
[root@deep /]# rdev /dev/sda12 /
```

Thiết bị gốc của kernel là một partition ở đó tập tin hệ thống root được đặt vào. Trong ví dụ này thiết bị gốc là "/dev/sda12", tên thiết bị này có thể khác so với trên hệ thống của bạn.

4. Thiết lập thiết bị root của kernel với dòng lệnh:

```
[root@deep /]# rdev /dev/fd0 /dev/sda12
```

Trên hệ thống của bạn thiết bị root của kernel được thay thế bởi kết quả nhận được của lệnh rdev trong bước số 3.

5. Đánh dấu thiết bị gốc chỉ cho phép đọc với lệnh:

```
[root@deep /]# rdev -R /dev/fd0 1
```

Bằng cách đặt thiết bị gốc chỉ để đọc, bạn tránh được vài cảnh báo và những messages lỗi.

6. Đặt đĩa mềm vào trong ổ đĩa A và khởi động lại hệ thống với lệnh:

```
[root@deep /]# reboot
```

Cập nhật thư mục "/dev"

Nếu bạn thêm những thiết bị mới vào hệ thống hoặc bạn đã nâng cấp kernel trong thời gian gần đây (Ví dụ bạn đã nâng cấp kernel từ version 2.2.9 lên 2.2.15). Điều quan trọng là bạn phải cập nhật (update) thư mục /dev để tránh những vấn đề liên quan tới việc thiếu thiết bị. Chúng ta có thể hoàn thành công việc này bằng chương trình tiện ích MAKEDEV, chương trình này sẽ tìm trong thư mục /dev những thiết bị có liên quan với những driver trong kernel thì được giữ lại. Một tùy chọn đặc biệt tên là "update" cho phép trình tiện ích tạo những thiết bị mới đã được bạn cấu hình trong kernel và xoá (delete) những thiết bị nào không được cấu hình.

- Cập nhật thư mục /dev thực thi những lệnh sau:

```
[root@deep /]# cd /dev
[root@deep /dev]# ./MAKEDEV update
```

PHẦN 3

LIÊN KẾT MẠNG

Trong phần này chúng ta sẽ đề cập đến các chương sau:

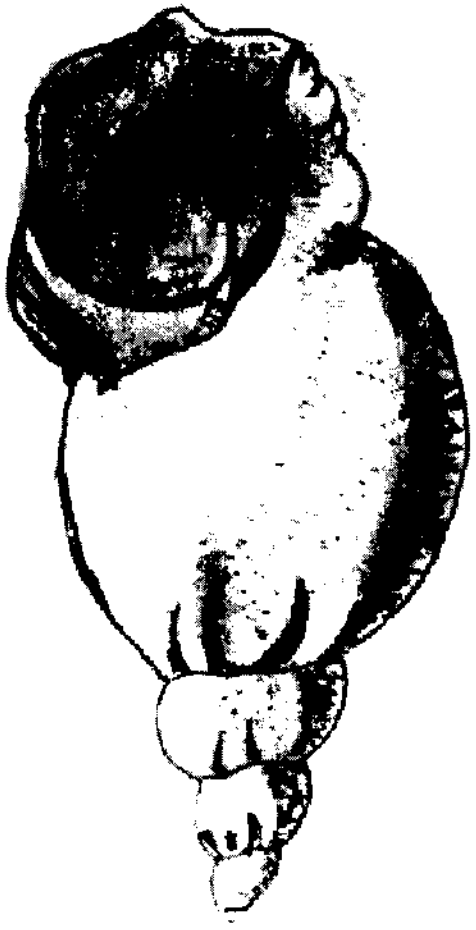
- **Quản Lý Hệ Thống Mạng Linux TCP/IP.**
- **Liên kết firewall.**
- **Liên kết firewall với hỗ trợ giả dạng IP và chuyển tiếp.**

Chương 6:

QUẢN LÝ HỆ THỐNG MẠNG LINUX TCP/IP
(Linux TCP/IP Network Management)

Trong chương này chúng ta sẽ đề cập đến các phần sau:

- Cài đặt nhiều card mạng trên một máy.
- Các tập tin liên quan đến chức năng mạng.
- Cấu hình mạng TCP bằng tay với giao diện dòng lệnh.



Giới thiệu tổng quan

Cho đến bây giờ, chúng ta vẫn chưa tận dụng hết các khả năng về mạng hiện có của hệ điều hành Linux. Linux là một trong những hệ điều hành tốt nhất đang có hiện nay trên thế giới về các đặc tính của mạng. Hầu hết các server web (websites) trên Internet khắp toàn cầu đều biết điều này, và đôi khi cũng có sử dụng nó. Việc hiểu rõ các thiết bị phần cứng của mạng và tất cả các tập tin liên quan với nó là rất quan trọng nếu bạn muốn có một sự điều khiển hoàn toàn tất cả những gì xảy ra trên máy chủ server của bạn. Có kiến thức tốt về các lệnh chủ yếu về mạng là rất quan trọng và cần thiết. Việc quản lý hệ thống mạng bao gồm một sự đa dạng của các chủ đề khác nhau. Nhìn chung, nó bao gồm sự tụ họp các dữ liệu được thống kê và trạng thái của các bộ phận trên mạng của bạn, và nó thực hiện một hành động khi cần thiết để giải quyết các sự cố và các thay đổi khác.

Phần lớn kỹ thuật thô sơ cho việc theo dõi mạng là "pinging" thường kỳ các máy chủ (host) chính yếu. Nhiều kỹ thuật giám sát mạng yêu cầu khả năng lấy được trạng thái và thông tin cụ thể từ các thiết bị khác nhau trong mạng. Những việc này sẽ bao gồm các sự phân loại khác nhau tổng số dữ liệu (datagram), cũng như số các lỗi của các dạng khác nhau. Để nhằm phục vụ cho mục đích trên, trong chương này chúng ta sẽ thử trả lời cho các câu hỏi cơ bản về các thiết bị mạng, các tập tin liên quan đến các chức năng của mạng, và các lệnh liên kết mạng chủ yếu.

Cài đặt nhiều card mạng trên một máy

Bạn có thể sử dụng Linux như một cổng nối (gateway) giữa hai mạng Ethernet (Ethernet - Phần cứng, định ước, và tiêu chuẩn ghép nối của một loại mạng cục bộ, do hãng Xerox Corporation đưa ra đầu tiên, có khả năng liên kết đến 1024 nút trong một mạng buýt.). Trong trường hợp đó, bạn phải có hai card Ethernet trên máy chủ server của bạn. Để loại trừ các vấn đề tại thời điểm khởi động, hạt nhân của Linux (Linux kernel) không phát hiện ra được nhiều card một cách tự động. Nếu bạn có hai hay nhiều card mạng, bạn nên chỉ rõ các thông số của các card trong tập tin "lilo.conf" cho một kernel nguyên khối (monolithic kernel) hoặc là trong tập tin "conf.modules" cho một kernel được module hóa (modularized kernel). Các vấn đề sau bạn có thể bắt gặp với các card mạng của bạn.

Vấn đề 1

Nếu các trình điều khiển (driver(s)) của nhiều card mạng được sử dụng như một module có thể nạp (modularized kernel), trong trường hợp của các trình điều khiển PCI drivers, module sẽ phát hiện ra tất cả các card được cài đặt một cách tự động. Đối với các card ISA, bạn cần phải cung cấp địa chỉ I/O của card như vậy module biết nơi đâu để thấy được. Thông tin này được lưu trữ trong tập tin "/etc/conf.modules".

Như một ví dụ sau, coi như chúng ta có hai card ISA 3c509, một cái tại địa chỉ I/O 0x300 và một cái tại địa chỉ I/O 0x320.

Đối với các card ISA, soạn thảo tập tin **conf.module** (vi /etc/conf.modules) và thêm vào:

```
alias eth0 3c509
alias eth1 3c509
options 3c509 io=0x300,0x320
```

. Điều này nói rằng trình điều khiển 3c509 sẽ được nạp cho hoặc eth0 hoặc là eth1 (bí danh là eth0, eth1) và nó nên được nạp với các tùy chọn io=0x300,0x320 do vậy các trình điều khiển (drivers) biết được nơi để tìm kiếm các card mạng. Ghi chú rằng, 0x thì quan trọng – điều này giống như 300h mà thường được dùng trong thế giới DOS sẽ không làm việc.

Đối với các PCI card, bạn chỉ cần các dòng bí danh (alias lines) để các giao tiếp ethN interface tương quan với tên trình điều khiển thích hợp, bởi vì các địa chỉ I/O cơ bản của PCI card có thể được nhận biết một cách an toàn.

Đối với các card PCI, soạn thảo tập tin **conf.module** (vi /etc/conf.modules) và thêm vào :

```
alias eth0 3c509
alias eth1 3c509
```

Vấn đề 2

Nếu một hoặc nhiều các trình điều khiển (driver(s)) của một hoặc nhiều card mạng là được biên dịch vào trong kernel (monolithic kernel), PCI sẽ dò tìm tất cả các card liên quan một cách tự động. Các card ISA cũng sẽ tìm tất cả các card liên quan một cách tự động, nhưng trong một số trường hợp các card ISA vẫn cần thực hiện các việc sau. Thông tin này cần được lưu trữ trong tập tin "/etc/lilo.conf". Giải pháp là truyền các thông số lúc khởi động tới kernel, mà thường được thực hiện bởi LILO.

Đối với các card ISA, soạn thảo tập tin **lilo.conf** (vi /etc/lilo.conf) và thêm vào:

```
append="ether=0,0,eth1"
```

Ghi chú: *Đầu tiên thử nghiệm các card ISA của bạn không có các thông số lúc khởi động trong tập tin "lilo.conf", và nếu điều này bị lỗi, thì dùng các thông số lúc khởi động.*

Trong trường hợp eth0 và eth1 sẽ được gán theo thứ tự mà các card được tìm ra lúc khởi động. Bởi vì chúng ta đã biên dịch lại kernel, chúng ta phải sử dụng phương pháp thứ hai (nếu mà các trình điều khiển được biên dịch vào trong kernel) để cài đặt card Ethernet thứ hai của bạn trên hệ thống. Hãy nhớ rằng đây chỉ là yêu cầu trong một số trường hợp cho các card ISA, các card PCI sẽ được tìm ra một cách tự động.

Các tập tin liên quan đến chức năng của mạng

Trong Linux, mạng TCP/IP được cấu hình thông qua nhiều tập tin văn bản mà bạn có thể phải soạn thảo để làm việc trên mạng. Đó là một điều rất quan trọng để biết các tập tin cấu hình liên quan tới mạng TCP/IP. Vì vậy bạn có thể

soạn thảo và cấu hình các tập tin nếu cần thiết. Hãy nhớ rằng máy chủ server của bạn không phải có một giao diện Xwindow để cấu hình các tập tin thông qua giao diện đồ hoạ. Thậm chí nếu bạn dùng một giao diện người dùng (GUI – Graphical User Interface) trong các hoạt động hàng ngày của bạn thì đây cũng là điều rất quan trọng để biết làm thế nào để cấu hình mạng trong chế độ văn bản (text mode). Các phần sau đây mô tả các tập tin cấu hình TCP/IP cơ bản.

Tập tin "/etc/HOSTNAME"

Tập tin này lưu trữ tên máy chủ (host name) của hệ thống – tên miền được gọi là đầy đủ (FQDN – Fully Qualified Domain Name) của hệ thống của bạn, chẳng hạn như deep.openna.com.

Sau đây là tập tin mẫu "/etc/HOSTNAME" :

deep.openna.com

Các tập tin "/etc/sysconfig/network-scripts/ifcfg-ethN"

Các cấu hình tập tin cho mỗi thiết bị mạng mà bạn có thể có hoặc muốn thêm vào hệ thống của bạn là thư mục được định vị trong "/etc/sysconfig/network-scripts/" với Red Hat Linux 6.1 hoặc 6.2 và được đặt tên là **ifcfg-eth0** cho giao tiếp đầu tiên và **ifcfg-eth1** cho cái thứ hai, vân vân ...

Sau đây là một tập tin mẫu "/etc/sysconfig/network-scripts/ifcfg-eth0":

```
DEVICE=eth0
IPADDR=208.164.186.1
NETMASK=255.255.255.0
NETWORK=208.164.186.0
BROADCAST=208.164.186.255
ONBOOT=yes
BOOTPROTO=none
USERCTL=no
```

Nếu bạn muốn tự hiệu chỉnh địa chỉ mạng của bạn bằng tay, hoặc thêm một mạng mới trên một giao tiếp mới, hiệu chỉnh tập tin này (ifcfg-ethN), hoặc tạo một tập tin mới với những thay đổi thích hợp.

DEVICE=***devicename***, trong đó ***devicename*** là tên của một thiết bị mạng vật lý.

IPADDR=***ipaddr***, trong đó ***ipaddr*** là địa chỉ IP.

NETMASK=***netmask***, trong đó ***netmask*** là giá trị netmask IP.

NETWORK=***network***, trong đó ***network*** là địa chỉ IP mạng.

BROADCAST=***broadcast***, trong đó ***broadcast*** là địa chỉ IP broadcast.

ONBOOT=***answer***, trong đó ***answer*** là yes hoặc no (Việc làm này tương ứng với interface sẽ được kích hoạt hoặc không vào thời điểm khởi động).

BOOTPROTO=**proto**, trong đó **proto** là một trong những giá trị sau:

- none – Giao thức không được sử dụng lúc khởi động.
- bootp – Giao thức bootp (bây giờ là pump) được sử dụng.
- dhcp – Giao thức dhcp được sử dụng.

USERCTL=***answer***, trong đó ***answer*** là một trong những giá trị sau:

- Yes (Các người sử dụng hệ thống không có quyền hạn cao nhất (root) được phép điều khiển thiết bị này).
- No (Chỉ có người sử dụng hệ thống có quyền hạn cao nhất (root) được phép điều khiển thiết bị này).

Tập tin "/etc/resolv.conf"

Đây là một tập tin văn bản khác, được sử dụng như thiết bị phân giải (resolver) – là một thư viện dùng để xác định địa chỉ IP cho tên của một máy chủ (host-name).

Sau đây là tập tin mẫu "/etc/resolv.conf" :

```
search openna.com
nameserver 208.164.186.1
nameserver 208.164.186.2
```

Ghi chú: *Tên các máy chủ (Name servers) được truy vấn theo thứ tự xuất hiện trong tập tin (chính-primary, phụ-secondary).*

Tập tin "/etc/host.conf"

Tập tin này chỉ rõ có bao nhiêu tên là được phân giải. Linux sử dụng bộ thư viện phân giải để lấy địa chỉ IP tương ứng với một tên máy chủ (host name).

Sau đây là một tập tin mẫu "/etc/host.conf":

```
# Lookup names via DNS first then fall back to /etc/hosts.
order bind,hosts
# We have machines with multiple addresses.
multi on
# Check for IP address spoofing.
nospoof on
```

Tùy chọn **order** nói lên thứ tự của các dịch vụ (services). Dòng ví dụ mẫu trong tập tin ở trên nó chỉ rõ rằng thư viện phân giải đầu tiên nên tra cứu name server (DNS) để phân giải một tên và sau đó kiểm tra tập tin "/etc/hosts".

Tùy chọn **multi** xác định có hay không một máy chủ host trong tập tin "/etc/hosts" có thể có nhiều địa chỉ IP (nhiều giao tiếp ethN) . Các Host mà có nhiều hơn một địa chỉ IP thì được biết đến như là *multihomed*, do sự hiện diện của nhiều địa chỉ IP điều này hàm ý rằng máy chủ có nhiều giao tiếp mạng.

Tùy chọn nospoof cho biết là không được phép đánh lừa trên máy này. Đánh lừa địa chỉ IP (IP-Spoofing) là một việc khai thác bảo mật, chúng làm việc bằng cách đánh lừa mọi người trong một mối quan hệ tin tưởng rằng thông tin gởi từ máy tính bạn là của một ai đó nhưng thật ra không phải như vậy.

Tập tin "/etc/sysconfig/network"

Tập tin "/etc/sysconfig/network" là được sử dụng để chỉ rõ các thông tin về việc cấu hình mạng trên máy chủ server của bạn.

Sau đây là một tập tin mẫu "/etc/sysconfig/network":

```
NETWORKING=yes
FORWARD_IPV4=yes
HOSTNAME=deep. openna.com
GATEWAY=0.0.0.0
GATEWAYDEV=
```

Sau đây là các giá trị có thể được dùng:

NETWORK= *answer*, trong đó *answer* là yes hoặc no (Cấu hình mạng hay không cấu hình mạng).

FORWARD= *answer*, trong đó *answer* là yes hoặc no (Thực hiện việc chuyển IP đi hay là không).

HOSTNAME = *hostname*, trong đó *hostname* là một hostname của máy server của bạn.

GATEWAY=*gwip*, trong đó *gwip* là địa chỉ IP cổng nối (gateway) của mạng từ xa. (nếu có) .

GATEWAYDEV=*gwdev*, trong đó *gwdev* là tên thiết bị (eth#) bạn dùng để truy cập gateway từ xa.

Ghi chú: *Để tương thích với các phần mềm cũ hơn, tập tin "/etc/HOSTNAME" sẽ chứa đựng cùng một giá trị như HOSTNAME*=hostname *ở trên. Với phiên bản mới hơn của Red Hat Linux 6.2 thì thông số "FORWARD_IP4=" bây giờ được định rõ trong tập tin "/etc/sysctl.conf" thay vì tập tin "/etc/sysconfig/network".*

Tập tin "/etc/sysctl.conf"

Trong Red Hat Linux 6.2, có nhiều tùy chọn kernel liên quan tới việc bảo mật hệ thống mạng, chẳng hạn như giảm bớt các packet đến trên các interface chúng sẽ không hoặc lờ đi các yêu cầu ping/broad v..v..có thể được đặt trong tập tin mới "/etc/sysctl.conf" thay vì tập tin "/etc/rc.d/rc.local". Một điều rất đáng quan tâm là thông số IPv4 forwarding mà bây giờ nó được thực hiện thông qua chương trình sysctl, trái ngược với nó được điều khiển bởi nội dung của tập tin "/etc/sysconfig/network". Các thông tin cài đặt sysctl được lưu trữ trong "/etc/sysctl.conf", và nó được nạp mỗi khởi động trước khi tập tin "/etc/rc.d/rc.local" được nạp. Chúng ta cũng đã nói về tất cả các thông số bảo mật mạng mà chúng ta

phải cài đặt vào trong máy chủ server trong chương 3, "Linux General Security", và cũng vì lý do này, chúng ta chỉ sẽ làm rõ hơn về tùy chọn kernel cho IPv4 forwarding.

- Để bật IPv4 forwarding trên hệ thống Red Hat 6.2, sử dụng các lệnh sau:

  Soạn thảo tập tin "/etc/sysctl.conf" và thêm vào các dòng sau:

  ```
  # Enable packet forwarding
  net.ipv4.ip_forward = 1
  ```

Bạn phải khởi động lại hệ thống mạng của bạn để thay đổi có hiệu lực. Lệnh để khởi động lại hệ thống mạng là như sau:

- Để khởi động lại tất cả các thiết bị mạng bằng tay trên hệ thống của bạn, làm theo chỉ dẫn sau:

  ```
  [root@deep /]# /etc/rc.d/init.d/network restart
  Setting network parameters          [ OK ]
  Bringing up interface lo            [ OK ]
  Bringing up interface eth0          [ OK ]
  Bringing up interface eth1          [ OK ]
  ```

Ghi chú: *Bật IPv4 forwarding thông qua tập tin sysctl.conf chỉ hợp lệ cho những người sử dụng Red Hat 6.2 . Các bạn dùng với phiên bản Red Hat 6.1 phải đặt thông số này vào trong "/etc/sysconfig/network" như được giải thích ở trên.*

Tập tin "/etc/hosts"

Khi máy của bạn bắt đầu khởi động, nó sẽ cần biết một sự ánh xạ (mapping) của một số hostname tới các địa chỉ IP trước khi DNS có thể được tham chiếu đến. Việc ánh xạ này sẽ được giữ lại trong tập tin "/etc/hosts". Trong trường hợp không tồn tại một name server, thì bất kỳ một chương trình mạng nào trên hệ thống của bạn đều phải tra cứu tới tập tin này để xác định rõ địa chỉ IP mà nó tương ứng với một tên host.

Sau đây là tập tin mẫu "/etc/hosts" :

IP Address	Hostname	Alias
127.0.0.1	localhost	deep.openna.com
208.164.186.1	deep.openna.com	deep
208.164.186.2	mail.openna.com	mail
208.164.186.3	web.openna.com	web

Cột bên trái nhất là địa chỉ IP được phân giải. Cột kế tiếp là tên của các máy host. Và cột liền theo sau là bí danh (alias) cho các máy host đó. Lấy ví dụ, trong dòng thứ hai ta có địa chỉ IP 208.164.186.1 cho host deep.openna.com. Và một tên khác cho deep.openna.com là deep.

Sau khi bạn thực hiện việc cấu hình trong các tập tin mạng của bạn xong, đừng quên khởi động lại hệ thống mạng để thay đổi một cách hiệu lực.

- Để khởi động lại hệ thống mạng của bạn, sử dụng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters      [ OK ]
Bringing up interface lo        [ OK ]
Bringing up interface eth0      [ OK ]
Bringing up interface eth1      [ OK ]
```

Ghi chú quan trọng: *Các vấn đề Time-out cho các kết nối telnet hay ftp thông thường được gây ra bởi các máy server cố gắng phân giải các địa chỉ IP của máy client tới một tên DNS. Hoặc là DNS không được cấu hình một cách chính xác trên máy server của bạn hoặc là các máy client không biết đến DNS. Nếu bạn có ý định chạy các dịch vụ telnet hay ftp trên máy server của bạn, và không dùng DNS, đừng quên phải thêm vào tên máy client và địa chỉ IP trong tập tin "/etc/hosts" trên server hoặc bạn có thể chờ đợi trong vài phút cho việc tìm kiếm DNS tới lúc time out, trước khi bạn thấy dấu nhắc chờ nhập lệnh.*

Cấu hình mạng TCP/IP bằng tay với giao diện dòng lệnh

Tiện ích ifconfig là công cụ được sử dụng cho việc cài đặt và cấu hình card mạng của bạn. Bạn nên hiểu lệnh này trong sự kiện bạn cần cấu hình mạng bằng tay. Một ghi chú quan trọng là phải cẩn thận với lệnh ifconfig khi sử dụng nó để cấu hình các thiết bị mạng của bạn; Việc thiết lập sẽ không cần khởi động lại hệ thống.

- Để gán eth0 giao tiếp địa chỉ IP của 208.164.186.2 sử dụng lệnh sau:

```
[root@deep /]# ifconfig eth0 208.164.186.2 netmask 255.255.255.0
```

Ghi chú: *Thông thường người ta chỉ cấu hình hoặc thay đổi mạng TCP/IP theo các chỉ thị lệnh thực hiện bằng tay khi chỉ thực hiện một số thử nghiệm trên server. Nếu bạn muốn lưu giữ các giá trị TCP/IP của bạn, việc làm này thích hợp với việc thiết đặt chúng trong các tập tin có liên quan tới các chức năng của mạng.*

- Để hiển thị tất cả các giao tiếp mà bạn có trên máy server của bạn, dùng lệnh sau:

```
[root@deep /]# ifconfig
```

Kết quả mọi thứ được xuất ra như sau:

```
eth0    Link encap:Ethernet HWaddr 00:E0:18:90:1B:56
        inet addr:208.164.186.2 Bcast:208.164.186.255 Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
        RX packets:1295 errors:0 dropped:0 overruns:0 frame:0
        TX packets:1163 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:100
        Interrupt:11 Base address:0xa800
```

```
lo      Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0
        UP LOOPBACK RUNNING MTU:3924 Metric:1
        RX packets:139 errors:0 dropped:0 overruns:0 frame:0
        TX packets:139 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
```

Ghi chú: *Nếu công cụ ifconfig được gọi không có bất kỳ thông số nào, nó hiển thị tất cả các giao tiếp (interface) mà bạn đã cấu hình. Còn tùy chọn "-a" hiển thị luôn các giao tiếp không được kích hoạt.*

- Để hiển thị tất cả các giao tiếp mà bạn có trên máy server của bạn, dùng lệnh sau:

```
[root@deep /]# ifconfig -a
```

Kết quả mọi thứ được xuất ra như sau:

```
eth0    Link encap:Ethernet HWaddr 00:E0:18:90:1B:56
        inet addr:208.164.186.2 Bcast:208.164.186.255
        Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
        RX packets:1295 errors:0 dropped:0 overruns:0 frame:0
        TX packets:1163 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:100
        Interrupt:11 Base address:0xa800

eth1    Link encap:Ethernet HWaddr 00:E0:18:90:1B:56
        inet addr:192.168.1.1 Bcast:192.168.1.255 Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
        RX packets:1295 errors:0 dropped:0 overruns:0 frame:0
        TX packets:1163 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:100
        Interrupt:5 Base address:0xa320

lo      Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0
        UP LOOPBACK RUNNING MTU:3924 Metric:1
        RX packets:139 errors:0 dropped:0 overruns:0 frame:0
        TX packets:139 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
```

Ghi chú: *Điều quan trọng phải ghi nhớ là các thông số cài đặt được cấu hình với công cụ ifconfig cho các thiết bị mạng của bạn sẽ không cần khởi động lại hệ thống.*

- Để gán một cổng nối (gateway) mặc nhiên cho 208.164.186.1 dùng lệnh:

```
[root@deep /]# route add default gw 208.164.186.1
```

Ghi chú: *Trong ví dụ này, tuyến mặc định (default route) được cài đặt tới địa chỉ 208.164.186.1, bởi router của bạn. Một lần nữa, nếu bạn muốn lưu giữ*

các giá trị gateway mặc nhiên, thì điều cần phải thực hiện là cài đặt chúng trong các tập tin liên quan tới chức năng mạng (/etc/sysconfig/network).

Kiểm tra rằng bạn có thể thấy được các host của bạn. Chọn một host từ mạng của bạn, chẳng hạn như 208.164.186.1

- Để kiểm tra rằng bạn có thể thấy các host của bạn, sử dụng lệnh sau:

  [root@deep /]# ping 208.164.186.1

  Kết quả mọi thứ được xuất ra như sau:

  [root@deep networking]# ping 208.164.186.1

  PING 208.164.186.1 (208.164.186.1) from 208.164.186.2 : 56 data bytes

  64 bytes from 208.164.186.2: icmp_seq=0 ttl=128 time=1.0 ms

  64 bytes from 208.164.186.2: icmp_seq=1 ttl=128 time=1.0 ms

  64 bytes from 208.164.186.2: icmp_seq=2 ttl=128 time=1.0 ms

  64 bytes from 208.164.186.2: icmp_seq=3 ttl=128 time=1.0 ms

  (khi nhấn Ctrl-C)

  --- 208.164.186.1 ping statistics ---

  4 packets transmitted, 4 packets received, 0% packet loss

  round-trip min/avg/max = 1.0/1.0/1.0 ms

Bây giờ bạn nên hiển thị các thông tin định tuyến (route) với lệnh route nếu cả hai host có bản route đúng:

- Để hiển thị thông tin dẫn đường, dùng lệnh

  [root@deep /]# route –n

  Kết quả mọi thứ được xuất ra như sau:

  Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
208.164.186.2	0.0.0.0	255.255.255.255	UH	0	0	0	eth0
208.164.186.0	208.164.186.2	255.255.255.0	UG	0	0	0	eth0
208.164.186.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
127.0.0.0	0.0.0.0	255.0.0.0	U	0	0	0	lo

- Để kiểm tra trạng thái của các giao tiếp một cách nhanh chóng, dùng lệnh netstat – i như sau:

  [root@deep /]# netstat –i

  Kết quả mọi thứ được xuất ra như sau:

  Kernel Interface table

Iface	MTU	Met	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
eth0	1500	0	4236	0	0	0	3700	0	0	0	BRU
lo	3924	0	13300	0	0	0	13300	0	0	0	LRU
ppp0	1500	0	14	1	0	0	16	0	0	0	PRU

Một tiện ích khác của netstat là –t , mà hiển thị tất cả các kết nối TCP/IP kích hoạt. Sau đây là kết quả tiêu biểu cho lệnh netstat –t:

- Để hiển thị tất cả các kết nối TCP/IP kích hoạt, dùng lệnh sau:

  [root@deep /]# netstat –t

  Kết quả mọi thứ được xuất ra như sau:

  Các kết nối internet kích hoạt (w/o servers)

Active Internet connections (w/o servers)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
Tcp	0	0	deep.openar:netbios-ssn	gate.openna.com:1045	ESTABLISHED
Tcp	0	0	localhost:1032	localhost:1033	ESTABLISHED
Tcp	0	0	localhost:1033	localhost:1032	ESTABLISHED
Tcp	0	0	localhost:1030	localhost:1034	ESTABLISHED
Tcp	0	0	localhost:1031	localhost:1030	ESTABLISHED
Tcp	0	0	localhost:1028	localhost:1029	ESTABLISHED
Tcp	0	0	localhost:1029	localhost:1028	ESTABLISHED
Tcp	0	0	localhost:1026	localhost:1027	ESTABLISHED
Tcp	0	0	localhost:1027	localhost:1026	ESTABLISHED
Tcp	0	0	localhost:1024	localhost:1025	ESTABLISHED
Tcp	0	0	localhost:1025	localhost:1024	ESTABLISHED

- Để hiển thị tất cả các kết nối TCP kích hoạt và lắng nghe, dùng lệnh sau:

  [root@deep /]# netstat –vat

  Kết quả mọi thứ được xuất ra như sau:

  Các kết nối internet kích hoạt (các server và sự thiết lập kết nối)

Active Internet connections (servers and established)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	deep.openna.co:domain	*:*	LISTEN
tcp	0	0	localhost:domain	*:*	LISTEN
tcp	0	0	deep.openna.com:ssh	gate.openna.com:1682	ESTABLISHED
tcp	0	0	*:webcache	*:*	LISTEN
tcp	0	0	deep.openar:netbios-ssn	*:*	LISTEN
tcp	0	0	localhost:netbios-ssn	*:*	LISTEN
tcp	0	0	localhost:1032	localhost:1033	ESTABLISHED
tcp	0	0	localhost:1033	localhost:1032	ESTABLISHED

tcp	0	0	localhost:1030	localhost:1031	ESTABLISHED
tcp	0	0	localhost:1031	localhost:1030	ESTABLISHED
tcp	0	0	localhost:1028	localhost:1029	ESTABLISHED
tcp	0	0	localhost:1029	localhost:1028	ESTABLISHED
tcp	0	0	localhost:1026	localhost:1027	ESTABLISHED
tcp	0	0	localhost:1027	localhost:1026	ESTABLISHED
tcp	0	0	localhost:1024	localhost:1025	ESTABLISHED
tcp	0	0	localhost:1025	localhost:1024	ESTABLISHED
tcp	0	0	deep.openna.com:www	*:*	LISTEN
tcp	0	0	deep.openna.com:https	*:*	LISTEN
tcp	0	0	*:389	*:*	LISTEN
tcp	0	0	*:ssh	*:*	LISTEN

- Để dừng (stop) tất cả các thiết bị mạng được thực hiện bằng tay trên hệ thống, sử dụng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network stop
Shutting down interface eth0                [ OK ]
Disabling IPv4 packet forwarding            [ OK ]
```

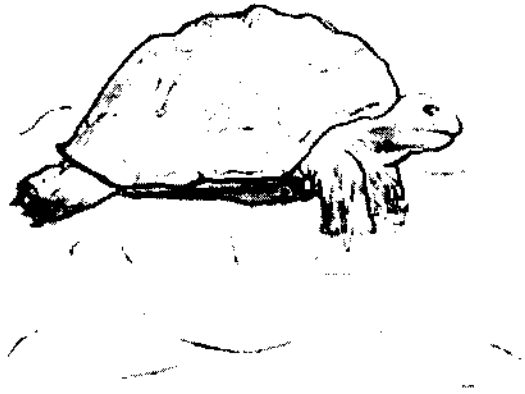
- Để khởi động (start) tất cả các thiết bị mạng được thực hiện bằng tay trên hệ thống, sử dụng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network start
Enabling IPv4 packet forwarding             [ OK ]
Bringing up interface lo                    [ OK ]
Bringing up interface eth0                  [ OK ]
```

Chương 7:
LIÊN KẾT FIREWALL

Trong chương này chúng ta sẽ đề cập đến các nội dung sau:

- Xây dựng một nhân của hệ thống (Kernel) với sự hỗ trợ IPCHAINS Firewall.
- Một vài sự giải thích về các nguyên lý được sử dụng trong các tập tin firewall script.
- Các tập tin firewall scripts.
- Cấu hình tập tin script "/etc/rc.d/init.d/firewall" cho Web Server.
- Cấu hình tập tin scripts "/etc/rc.d/init.d/firewall" cho Mail Server.
- Tóm tắt về IPCHAINS.



Linux IPCHAINS

Giới thiệu tổng quan

Có thể có ai đó nói với tôi rằng, tại sao tôi không dùng một sản phẩm thương mại "bức tường lửa" (Firewall) thay vì đơn giản sử dụng ipchains cho việc giới hạn chính xác các gói dữ liệu (package)? Tôi sẽ mất những gì nếu sử dụng Ipchains? Bây giờ không có lý do gì cho việc tranh cãi này, Ipchains thật sự tuyệt vời, bạn nên giành nhiều thời gian tập trung nghiên cứu nó tốt hơn là cho các sản phẩm firewall thương mại do bởi chức năng và các chuẩn hỗ trợ. Bạn có thể sẽ có cái nhìn sâu hơn vào bên trong hệ thống mạng máy tính của bạn để thấy nó làm việc thế nào khi sử dụng Ipchains hơn là một giải pháp thương mại. Điều đó nói lên rằng, có nhiều tổ chức, công ty cổ phần muốn nói với các cổ đông của họ, CEO/CTO,..v.v..rằng họ có công ty phần mềm bảo mật đáng tin cậy ở đằng sau họ.

Tóm lại, nhiều công ty muốn có thể đòi hỏi một số bồi thường thiệt hại từ một nhà cung cấp nếu hệ thống mạng bị xâm nhập, hoặc họ thật sự không biết có ích lợi gì, hoặc là thậm chí chỉ thứ thôi. Tất cả những nhu cầu trên họ có thể làm với giải pháp tiêu biểu là dùng mã nguồn mở (open source), giải pháp này thật sự hấp dẫn với những người phát triển và hiện thực chúng. Ít nhất là một vài firewall thương mại được phát triển dựa trên hệ điều hành Linux hoặc là một cái gì đó tương tự. Điều đó chứng minh rằng Ipchains có thể đủ bảo mật cho bạn nhưng mà nó không hứa hẹn cho một giải pháp thương mại. Làm một sự phân tích chi phí, tiện lợi và trả lời các câu hỏi thích hợp được đề nghị trước khi chi trả một số tiền lớn cho một sản phẩm firewall thương mại – mặt khác bạn có thể đi đến kết luận cuối cùng với một sản phẩm nào đó mà nó ít tốn kém hơn so với công cụ Ipchains của bạn. Hầu hết các firewall NT là cũng tương tự và cũng không tốt hơn Ipchains.

Cơ chế bảo mật Firewall là gì ?

Cơ chế bảo mật Firewall định nghĩa một số dịch vụ mà nó sẽ cho phép hoặc từ chối các yêu cầu truy cập một cách cụ thể. Những dịch vụ này sẽ được dùng như thế nào và tuân theo những nguyên lý cụ thể nào? Toàn bộ cơ chế bảo mật của một tổ chức phải được xác định rõ tùy theo việc phân tích bảo mật và các yêu cầu kinh doanh cần thiết. Bởi vì một firewall liên quan đến mạng riêng rẽ cần bảo mật, một firewall có hiệu quả tốt nếu như toàn bộ cơ chế bảo mật được định nghĩa một cách chính xác. Mọi nguyên lý trong cơ chế bảo mật mạng firewall nên được thực hiện trên một firewall. Nói chung, firewall sử dụng một trong các phương pháp sau đây:

> *Mọi thứ không được phép một cách rõ ràng thì bị từ chối (Everything not specifically permitted is denied).*

Phương pháp này sẽ đóng tất cả các lưu thông giữa hai mạng ngoại trừ các ứng dụng và các dịch vụ được cho phép tường minh. Vì vậy, mỗi dịch vụ và ứng dụng mong muốn nên được thực hiện lần lượt cho từng cái một. Các dịch vụ hay ứng dụng nào mà có tiềm năng gây lỗ hổng trên firewall thì sẽ không được phép.

Đây là giải pháp bảo mật nhất, từ chối các dịch vụ và các ứng dụng ngoại trừ nó được cho phép bởi người quản trị hệ thống mạng. Mặt khác, từ phía của người sử dụng nó có thể bị hạn chế hơn và ít tiện lợi hơn. Đây là phương pháp mà chúng ta sẽ dùng trong các tập tin cấu hình Firewall của chúng ta trong quyển sách này.

> *Mọi thứ không bị từ chối một cách rõ ràng thì được phép (Everything not specifically denied is permitted).*

Phương pháp này cho phép tất cả các lưu thông giữa hai mạng ngoại trừ đối với các dịch vụ và các ứng dụng mà bị từ chối. Chính vì vậy, mà mỗi dịch vụ hay ứng dụng không được tin cậy hay có tiềm năng gây bất lợi nên bị từ chối một cách lần lượt từng cái một. Mặc dù đây là phương pháp linh động và tiện lợi cho người sử dụng, nhưng nó có thể là tiềm năng gây ra một số vấn đề bảo mật nghiêm trọng.

Quá trình sàng lọc các gói tin là gì ? (What is Packet Filtering?)

Packet Filtering là một loại firewall được xây dựng sẵn bên trong hạt nhân của Linux. Một filtering firewall làm việc ở tầng mạng. Dữ liệu chỉ được phép rời khỏi hệ thống nếu các qui tắc (rule) firewall cho phép. Cũng như các gói tin (packet) đến chúng sẽ được sàng lọc bởi kiểu của chúng, địa chỉ nguồn, địa chỉ đích, và thông tin cổng (port) được chứa đựng bên trong mỗi gói tin (packet).

Hầu hết thời gian, packet filtering được thực hiện bằng cách sử dụng một bộ tìm đường (router), router có thể chuyển tiếp (forward) các packet thông qua các nguyên lý sàng lọc (filtering). Khi mà một packet đến một packet-filtering router , router trích lấy một vài thông tin nào đó từ packet header và ra quyết định y theo filter rules và qua đó các packet sẽ được gửi đi hay bị loại bỏ.

Các thông tin sau đây có thể được trích từ packet header :

- Địa chỉ IP nguồn.
- Địa chỉ IP đích.
- Cổng nguồn TCP/UDP.
- Cổng đích TCP/UDP.
- Loại thông điệp ICMP.
- Thông tin giao thức được đóng gói (TCP, UDP, ICMP hoặc là IP tunnel).

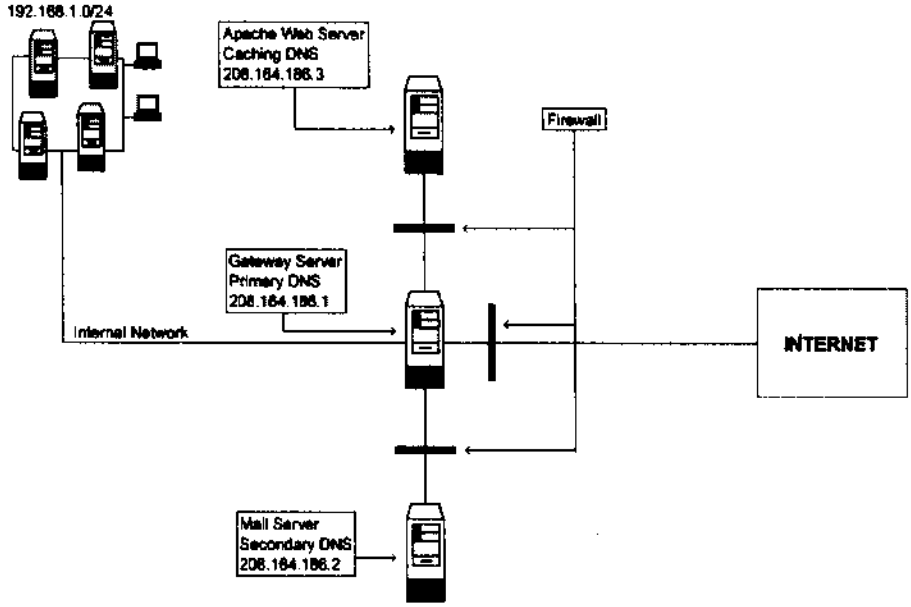
Bởi vì rất ít dữ liệu được phân tích và được ghi chú (log), filtering firewall chiếm lấy ít hơn năng lực của CPU và tạo ra rất ít sự tiềm tàng trong hệ thống mạng của bạn. Có rất nhiều cách cấu trúc hệ thống mạng của bạn để bảo vệ hệ thống của bạn sử dụng một firewall.

Cấu trúc liên kết Mạng (The topology)

Tất cả các máy chủ nên được cấu hình để ngăn chặn ít nhất những cổng (port) không được dùng đến, thậm chí khi không có một máy chủ firewall nào. Đây là một yêu cầu đặt ra cho việc bảo mật. Bạn hãy tưởng tượng rằng có ai đó có

quyền truy cập tới máy chủ firewall gateway server của bạn: nếu các máy chủ servers cận kề xung quanh nó mà chưa được cấu hình ngăn cấm các port chưa được dùng. Đây là một sự mạo hiểm đáng sợ. Tương tự cũng đúng cho các kết nối cục bộ; các nhân viên không có quyền sở hữu có thể có quyền truy cập từ bên trong tới các máy chủ server khác với cùng kiểu như trên.

Trong việc cấu hình dưới đây, chúng tôi sẽ cung cấp cho bạn ba ví dụ khác nhau mà có thể giúp đỡ cho bạn cấu hình các nguyên lý firewall của bạn (firewall rules) phụ thuộc vào loại server mà bạn muốn bảo vệ và nơi đặt các máy chủ server này trong kiến trúc mạng của bạn. Ví dụ đầu tiên, tập tin firewall rules sẽ cho một Web Server, thứ hai sẽ cho một Mail Server và cuối cùng là cho một Gateway Server mà nó hoạt động như một proxy cho các máy Wins, các máy trạm làm việc (Workstation) và các máy chủ Server. Xem hình bên dưới để hiểu rõ hơn.



www.openna.com **Caching Only DNS** **208.164.186.3**	**deep.openna.com** **Master DNS Server** **208.164.186.1**	**mail.openna.com** **Slave DNS Server** **208.164.186.2**
1. Cho phép không giới hạn sự lưu thông trên giao tiếp hồi chuyển (lo interface).	1. Cho phép không giới hạn sự lưu thông trên giao tiếp hồi chuyển (lo interface).	1. Cho phép không giới hạn sự lưu thông trên giao tiếp hồi chuyển (lo interface).

2. Sự lưu thông ICMP là được phép. 3. DNS Caching và Client Server trên port 53 là được phép. 4. SSH Server trên port 22 là được phép. 5. HTTP Server trên port 80 là được phép. 6. HTTPS Server trên port 443 là được phép. 7. SMTP Client trên port 25 là được phép. 8. FTP Server trên port 20, 21 là được phép. 9. Outgoing traceroute request là được phép.	2. Sự lưu thông ICMP là được phép. 3. DNS Caching và Client Server trên port 53 là được phép. 4. SSH Server trên port 22 là được phép. 5. HTTP Server và client trên port 80 là được phép. 6. HTTPS Server và client trên port 443 là được phép. 7. WWW-CACHE Client trên port 8080 là được phép. 8. External POP Client trên port 110 là được phép. 9. External NNTP NEWS Client trên port 119 là được phép. 10. SMTP Server và Client trên port 25 là được phép. 11. IMAP Server trên port 143 là được phép. 12. IRC Client trên port 6667 là được phép. 13. ICQ Client trên port 4000 là được phép. 14. FTP Client trên port 20, 21 là được phép.	2. Sự lưu thông ICMP là được phép. 3. DNS Caching và Client Server trên port 53 là được phép. 4. SSH Server trên port 22 là được phép. 5. SMTP Client trên port 25 là được phép 6. IMAP Server trên port 143 là được phép. 7. Outgoing traceroute request là được phép.

	15. RealAudio/ QuickTime Client là được phép. 16. Outgoing traceroute request là được phép.	

Bảng trên trình bày cho bạn thấy các cổng (port) mà chúng tôi đã sử dụng trên các máy chủ server khác nhau một cách mặc nhiên trong các tập tin dạng kịch bản firewall scripts trong quyển sách này. Phụ thuộc vào các dịch vụ nào mà nó phải có sẵn trên máy chủ server cho bên ngoài, bạn phải cấu hình tập tin firewall script của bạn để cho phép sự lưu thông trên các cổng (port) được chỉ định.

www.openna.com là một Web Server, **mail.openna.com** là một Mail Hub Server cho tất cả hệ thống mạng bên trong và **deep.openna.com** là Gateway Server của chúng ta cho tất cả các ví dụ được giải thích trong chương này.

Xây dựng một nhân của hệ thống (Kernel) với sự hỗ trợ IPCHAINS Firewall

Điều đầu tiên mà bạn cần phải làm là bảo đảm rằng kernel của bạn đã được xây dựng với sự hỗ trợ Network Firewall có hiệu lực và ngăn chặn hiệu quả. **Hãy nhớ rằng, tất cả các server nên được cấu hình để ngăn cấm các port không dùng đến, thậm chí khi không có một firewall server nào**. Trong phiên bản kernel 2.2.14 bạn cần phải chắc rằng bạn có câu trả lời **Y** với các câu hỏi sau đây :

Các tùy chọn mạng :

```
Network firewalls (CONFIG_FIREFALL) [N] Y
IP:Firewalling (CONFIG_IP_FIREWALL) [N] Y
IP:TCP syncookie support (CONFIG_SYN_COOKIES) [N] Y
```

Ghi chú: *Nếu bạn có phần Linux kernel và được biên dịch rồi , thì các tùy chọn "Network firewalls, IP:Firewalling, and IP:TCP syncookie support" được nói bên trên là được cài đặt sẵn.*

Một vài sự giải thích về các nguyên lý được sử dụng trong các tập tin firewall script

Sau đây là một giải thích về một ít các nguyên lý mà nó sẽ được sử dụng trong các ví dụ Firewall sau. Điều này chỉ được trình bày như một tham khảo, các firewall scripts được chú thích rất rõ ràng và dễ dàng hiệu chỉnh.

Các hằng số (constants) được dùng trong các ví dụ về tập tin firewall scripts

Các hằng số là được sử dụng cho hầu hết các giá trị. Các hằng số cơ bản nhất là :

EXTERNAL_INTERFACE

Đây là tên của giao tiếp mạng bên ngoài với Internet. Nó được định nghĩa là **eth0** trong các ví dụ.

LOCAL_INTERFACE_1

Đây là tên của giao tiếp mạng bên ngoài với mạng cục bộ (LAN). Nó được định nghĩa là **eth1** trong các ví dụ.

LOOPBACK_INTERFACE

Đây là tên của giao tiếp hồi chuyển. Nó được định nghĩa là **lo** trong các ví dụ.

IPADDR

Đây là địa chỉ IP của giao tiếp mạng bên ngoài với Internet (external interface). Nó cũng là một địa chỉ IP tĩnh được đăng ký với InterNIC, hoặc là động được gán địa chỉ từ ISP của bạn (thường thông qua DHCP).

LOCALNET_1

Đây là một địa chỉ mạng LAN của bạn, nếu có – toàn bộ miền địa chỉ IP nhập vào được dùng bởi các máy trong mạng LAN của bạn. Tất các các địa chỉ này có thể được gán tĩnh, hoặc bạn có thể chạy một máy chủ DHCP cục bộ để gán cho chúng. Trong các ví dụ này, miền địa chỉ 192.168.1.0/24, Phần miền địa chỉ riêng của lớp C.

ANYWHERE

Anywhere là một ký hiệu cho một địa chỉ được dùng bởi ipchians để xem bất kỳ địa chỉ nào. Cả hai chương trình được cung cấp **any/0** là một ký hiệu cho địa chỉ này, mà nó là 0.0.0.0/0

NAMESERVER_1

Đây là địa chỉ IP của Primary DNS Server từ hệ thống mạng của bạn hoặc ISP của bạn.

NAMESERVER_2

Đây là địa chỉ IP của Secondary DNS Server từ hệ thống mạng của bạn hoặc ISP của bạn.

MY_ISP

Đây là miền địa chỉ ISP & NOC. Giá trị mà bạn chỉ ra ở đây được dùng bởi firewall để cho phép yêu cầu ICMP ping và traceoute. Nếu bạn không chỉ ra một miền địa chỉ IP, khi đó bạn sẽ không thể kiểm tra tính liên thông (ping) mạng Internet từ mạng nội bộ của bạn.

LOOPBACK

Miền địa chỉ loopback là 127.0.0.0/8. Giao tiếp với chính nó là địa chỉ 127.0.0.1 (trong /etc/hosts).

PRIVPORTS

Là các port giành riêng, từ 0 đến 1023, thường được dùng bởi các chương trình nội bộ.

UNPRIVPORTS

Là các cổng không đặc quyền, từ 1024 đến 65535, thường được dùng bởi nhiều ứng dụng. Chúng là các địa chỉ động được gán cho phía client của một kết nối.

Default Policy

Một firewall có một cơ chế mặc định và một tập hợp các hành động để đáp lại các loại thông điệp cụ thể nào đó. Điều này có nghĩa là nếu mà một gói (packet) đưa vào mà không được chọn bởi bất kỳ một nguyên lý nào, khi đó nguyên lý mặc nhiên sẽ được sử dụng.

Ghi chú: *Những người có các địa chỉ IP được gán động từ một ISP có thể gồm có hai dòng sau trong khai báo của họ cho firewall. Các dòng này sẽ xác định địa chỉ IP ppp0, và mạng server ppp từ xa.*

```
IPADDR=`/sbin/ifconfig | grep -A 4 ppp0 | awk '/inet/ { print $2 } ' | sed -e
s/addr://`

MY_ISP=`/sbin/ifconfig | grep -A 4 ppp0 | awk '/P-t-P/ { print $3 } ' | sed -e s/P-
t-P:// | cut -d '.' -f 1-3`.0/24
```

Cho phép lưu thông cục bộ

Bởi vì các cơ chế mặc nhiên (default policies) cho tất cả ví dụ về tập tin firewall rule scripts trong quyển sách này là từ chối mọi thứ, một số các rules này không phải cài đặt. Các dịch vụ mạng cục bộ không đi thông qua external network interface. Chúng đi thông qua một private interface đặc biệt được gọi là loopback interface. Không có chương trình mạng cục bộ nào của bạn sẽ làm việc cho đến khi loopback traffic là được phép.

```
# Unlimited traffic on the loopback interface.
ipchains -A input -i $LOOPBACK_INTERFACE -j ACCEPT
ipchains -A output -i $LOOPBACK_INTERFACE -j ACCEPT
```

Lọc địa chỉ nguồn

Tất cả các packet header chứa đựng các địa chỉ IP nguồn và đích và kiểu giao thức truyền thông điệp IP (ICMP, UDP, hoặc TCP) mà các packet này chứa đựng. Điều này chỉ có ý nghĩa của sự nhận biết dưới Internet Protocol (IP) là địa chỉ nguồn trong packet header IP. Đây là một vấn đề mà nó mở rộng cánh cửa để giả dạng địa chỉ nguồn, ở đây người gửi có thể thay thế địa chỉ của nó với một địa chỉ không có tồn tại, hoặc địa chỉ của một số site khác.

```
# Refuse spoofed packets pretending to be from the external address.
ipchains -A input -i $EXTERNAL_INTERFACE -s $IPADDR -l -j DENY
```

Cũng như vậy, có ít nhất bảy tập địa chỉ nguồn bạn nên từ chối trên external interface của bạn trong tất cả các trường hợp.

Các packets đến được yêu cầu ở dạng:

- Địa chỉ external IP của bạn.
- Địa chỉ IP private của lớp A.
- Địa chỉ IP private của lớp B.
- Địa chỉ IP private của lớp C.
- Địa chỉ IP multicast của lớp D.
- Địa chỉ IP reserved của lớp E.
- Loopback interface.

Với ngoại lệ của địa chỉ IP riêng của bạn, ngăn chặn packet đi ra chứa đựng các địa chỉ nguồn nhằm bảo vệ bạn từ các lỗi cấu hình có thể xảy ra ở phía bạn.

Ghi chú: *Đừng quên loại trừ địa chỉ IP của riêng bạn từ các packet đi ra đã được ngăn chặn. Mặc định tôi đã chọn loại trừ các địa chỉ IP riêng rẽ (private IP) của Class C vì hầu như nó được dùng bởi phần lớn người sử dụng vào thời điểm này. Nếu bạn sử dụng một lớp khác thay cho lớp C, khi đó bạn phải ghi ra các dòng chú thích mà tham chiếu tới Class của bạn ở phần "SPOOFING & BAD ADDRESS" của tập tin firewall script.*

Phần còn lại của các rule

Các nguyên lý khác được dùng trong tập tin firewall scripts là :

- Truy cập một dịch vụ từ thế giới bên ngoài.
- Đưa ra một dịch vụ cho thế giới bên ngoài.
- Giả dạng các máy nội bộ.

Các tập tin firewall scripts

Bộ công cụ ipchains cho phép bạn cài đặt firewall, giả dạng địa chỉ IP (IP masquerading),..v.v...ipchains "nói chuyện" với kernel và bảo cho nó biết các gói thông tin (packets) nào phải sàng lọc (filter). Chính vì vậy mà tất cả các thông tin cài đặt firewall của bạn được lưu trữ trong kernel, và cũng chính vì vậy mà nó sẽ bị mất thông tin trong trường hợp khởi động lại hệ thống. Để tránh trường hợp này, chúng tôi yêu cầu sử dụng System V khởi tạo các scripts để tạo các nguyên lý (rules) của bạn cố định. Để làm điều này, tạo một tập tin firewall script giống như được trình bày bên dưới trong thư mục "/etc/rc.d/init.d/" của bạn cho mỗi máy chủ servers mà bạn có. Dĩ nhiên mỗi máy chủ server có các dịch vụ khác được đưa ra và cần cài đặt firewall khác nhau. Để đáp ứng các lý do được trình bày trong phần

này, chúng tôi cung cấp cho bạn ba thông tin cài đặt firewall khác nhau, mà bạn có thể sử dụng nó vào trong hệ thống của mình và kiểm tra nó để phù hợp với các yêu cầu cần thiết của bạn. Và tôi cũng cho rằng bạn phải có ít kiến thức về filtering firewall và firewall rules làm việc như thế nào.

Cấu hình tập tin script "/etc/rc.d/init.d/firewall" cho Web Server

Đây là tập tin cấu hình script cho Web Server của bạn. Sự cấu hình này cho phép lưu thông không giới hạn trên Loopback interface, ICMP, DNS Caching và Client Server (53), SSH Server (22), HTTP Server (80), HTTPS Server (443), SMTP Client (25), FTP Server (20,21), và OUTGOING TRACEROUTE được yêu cầu bởi mặc định.

Nếu bạn không muốn một số các dịch vụ (services) được liệt kê trong các tập tin firewall rules cho Web Server mà mặc định bạn đã mở, chú thích cho chúng với một ký tự "#" đặt ở đầu dòng. Nếu bạn cần mở một số dịch vụ khác mà bạn đã ghi chú với một ký tự "#", thì xóa ký tự "#" ở đầu dòng của các dịch vụ đó.

Tạo tập tin firewall script (touch /etc/rc.d/init.d/firewall) trên Web Server của bạn và thêm vào :

```
#!/bin/sh
#
# ------------------------------------------------------------------
# Last modified by Gerhard Mourani: 04-25-2000
# ------------------------------------------------------------------
# Copyright (C) 1997, 1998, 1999 Robert L. Ziegler
#
# Permission to use, copy, modify, and distribute this software and its
# documentation for educational, research, private and non-profit purposes,
# without fee, and without a written agreement is hereby granted.
# This software is provided as an example and basis for individual firewall
# development. This software is provided without warranty.
#
# Any material furnished by Robert L. Ziegler is furnished on an
# "as is" basis. He makes no warranties of any kind, either expressed
# or implied as to any matter including, but not limited to, warranty
# of fitness for a particular purpose, exclusivity or results obtained
# from use of the material.
# ------------------------------------------------------------------
#
# Invoked from /etc/rc.d/init.d/firewall.
# chkconfig: - 60 95
```

```
# description: Starts and stops the IPCHAINS Firewall \
# used to provide Firewall network services.
# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
if [ ${NETWORKING} = "no" ]
then
        exit 0
fi

if [ ! -x /sbin/ipchains ]; then
        exit 0
fi

# See how we were called.
case "$1" in
    start)
        echo -n "Starting Firewalling Services: "

# Some definitions for easy maintenance.

# ------------------------------------------------------------------
# EDIT THESE TO SUIT YOUR SYSTEM AND ISP.
EXTERNAL_INTERFACE="eth0"              # Internet connected interface
LOOPBACK_INTERFACE="lo"                # Your local naming convention
IPADDR="my.ip.address"                 # Your IP address
ANYWHERE="any/0"                       # Match any IP address
NAMESERVER_1="my.name.server.1"        # Everyone must have at least one
NAMESERVER_2="my.name.server.2"        # Your secondary name server
MY_ISP="my.isp.address.range/24"       # ISP & NOC address range
SMTP_SERVER="my.smtp.server"           # Your Mail Hub Server.
SYSLOG_SERVER="syslog.internal.server" # Your syslog internal server
SYSLOG_CLIENT="sys.int.client.range/24" # Your syslog internal client range
LOOPBACK="127.0.0.0/8"                 # Reserved loopback address range
CLASS_A="10.0.0.0/8"                   # Class A private networks
CLASS_B="172.16.0.0/12"                # Class B private networks
```

```
CLASS_C="192.168.0.0/16"                # Class C private networks
CLASS_D_MULTICAST="224.0.0.0/4"         # Class D multicast addresses
CLASS_E_RESERVED_NET="240.0.0.0/5"      # Class E reserved addresses
BROADCAST_SRC="0.0.0.0"                 # Broadcast source address
BROADCAST_DEST="255.255.255.255"        # Broadcast destination address
PRIVPORTS="0:1023"                      # Well known, privileged port range
UNPRIVPORTS="1024:65535"                # Unprivileged port range

# ..........
# SSH starts at 1023 and works down to 513 for
# each additional simultaneous incoming connection.
SSH_PORTS="1022:1023"                   # range for SSH privileged ports

# traceroute usually uses -S 32769:65535 -D 33434:33523
TRACEROUTE_SRC_PORTS="32769:65535"
TRACEROUTE_DEST_PORTS="33434:33523"

# ..........
# Default policy is DENY
# Explicitly accept desired INCOMING & OUTGOING connections
   # Remove all existing rules belonging to this filter
   ipchains -F
   # Clearing all current rules and user defined chains
   ipchains -X
   # Set the default policy of the filter to deny.
   # Don't even bother sending an error message back.
   ipchains -P input DENY
   ipchains -P output DENY
   ipchains -P forward DENY
# ..........

# LOOPBACK
   # Unlimited traffic on the loopback interface.
   ipchains -A input -i $LOOPBACK_INTERFACE -j ACCEPT
   ipchains -A output -i $LOOPBACK_INTERFACE -j ACCEPT
# ..........
# Network Ghouls
# Deny access to jerks
   # /etc/rc.d/rc.firewall.blocked contains a list of
   # ipchains -A input -i $EXTERNAL_INTERFACE -s address -j DENY
```

```
   # rules to block from any access.
   # Refuse any connection from problem sites
   #if [ -f /etc/rc.d/rc.firewall.blocked ]; then
   # . /etc/rc.d/rc.firewall.blocked
   #fi
# ------------------------------------------------------------------

# SPOOFING & BAD ADDRESSES
# Refuse spoofed packets.
# Ignore blatantly illegal source addresses.
# Protect yourself from sending to bad addresses.
   # Refuse spoofed packets pretending to be from the external address.
   ipchains -A input -i $EXTERNAL_INTERFACE -s $IPADDR -j DENY -l

   # Refuse packets claiming to be to or from a Class A private network
   ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_A -j DENY -l
   ipchains -A input -i $EXTERNAL_INTERFACE -d $CLASS_A -j DENY -l
   ipchains -A output -i $EXTERNAL_INTERFACE -s $CLASS_A -j REJECT -l
   ipchains -A output -i $EXTERNAL_INTERFACE -d $CLASS_A -j REJECT -l

   # Refuse packets claiming to be to or from a Class B private network
   ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_B -j DENY -l
   ipchains -A input -i $EXTERNAL_INTERFACE -d $CLASS_B -j DENY -l
   ipchains -A output -i $EXTERNAL_INTERFACE -s $CLASS_B -j REJECT -l
   ipchains -A output -i $EXTERNAL_INTERFACE -d $CLASS_B -j REJECT -l

   # Refuse packets claiming to be to or from a Class C private network
#  ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_C -j DENY -l
#  ipchains -A input -i $EXTERNAL_INTERFACE -d $CLASS_C -j DENY -l
#  ipchains -A output -i $EXTERNAL_INTERFACE -s $CLASS_C -j REJECT -l
#  ipchains -A output -i $EXTERNAL_INTERFACE -d $CLASS_C -j REJECT -l

   # Refuse packets claiming to be from the loopback interface
   ipchains -A input -i $EXTERNAL_INTERFACE -s $LOOPBACK -j DENY -l
   ipchains -A output -i $EXTERNAL_INTERFACE -s $LOOPBACK -j REJECT -l

   # Refuse broadcast address SOURCE packets
   ipchains -A input -i $EXTERNAL_INTERFACE -s $BROADCAST_DEST -j DENY -l
   ipchains -A input -i $EXTERNAL_INTERFACE -d $BROADCAST_SRC -j DENY -l
```

```
# Refuse Class D multicast addresses (in.h) (NET-3-HOWTO)
# Multicast is illegal as a source address.
# Multicast uses UDP.
ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_D_MULTICAST -j DENY
-l

# Refuse Class E reserved IP addresses
ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_E_RESERVED_NET -j
DENY -l

# refuse addresses defined as reserved by the IANA
# 0.*.*.*, 1.*.*.*, 2.*.*.*, 5.*.*.*, 7.*.*.*, 23.*.*.*, 27.*.*.*
# 31.*.*.*, 37.*.*.*, 39.*.*.*, 41.*.*.*, 42.*.*.*, 58-60.*.*.*
# 65-95.*.*.*, 96-126.*.*.*, 197.*.*.*, 201.*.*.* (?), 217-223.*.*.*

ipchains -A input -i $EXTERNAL_INTERFACE -s 1.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 2.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 5.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 7.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 23.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 27.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 31.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 37.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 39.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 41.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 42.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 58.0.0.0/7 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 60.0.0.0/8 -j DENY -l

#65: 01000001 - /3 includes 64 - need 65-79 spelled out
ipchains -A input -i $EXTERNAL_INTERFACE -s 65.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 66.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 67.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 68.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 69.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 70.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 71.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 72.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 73.0.0.0/8 -j DENY -l
```

```
ipchains -A input -i $EXTERNAL_INTERFACE -s 74.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 75.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 76.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 77.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 78.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 79.0.0.0/8 -j DENY -l

#80: 01010000 - /4 masks 80-95
ipchains -A input -i $EXTERNAL_INTERFACE -s 80.0.0.0/4 -j DENY -l

# 96: 01100000 - /4 makses 96-111
ipchains -A input -i $EXTERNAL_INTERFACE -s 96.0.0.0/4 -j DENY -l

#126: 01111110 - /3 includes 127 - need 112-126 spelled out
ipchains -A input -i $EXTERNAL_INTERFACE -s 112.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 113.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 114.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 115.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 116.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 117.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 118.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 119.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 120.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 121.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 122.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 123.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 124.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 125.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 126.0.0.0/8 -j DENY -l

#217: 11011001 - /5 includes 216 - need 217-219 spelled out
ipchains -A input -i $EXTERNAL_INTERFACE -s 217.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 218.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 219.0.0.0/8 -j DENY -l

#223: 11011111 - /6 masks 220-223
ipchains -A input -i $EXTERNAL_INTERFACE -s 220.0.0.0/6 -j DENY -l

# ------------------------------------------------------------------
# ICMP
```

```
# To prevent denial of service attacks based on ICMP bombs, filter
# incoming Redirect (5) and outgoing Destination Unreachable (3).
# Note, however, disabling Destination Unreachable (3) is not
# advisable, as it is used to negotiate packet fragment size.

# For bi-directional ping.
# Message Types: Echo_Reply (0), Echo_Request (8)
# To prevent attacks, limit the src addresses to your ISP range.
#
# For outgoing traceroute.
# Message Types: INCOMING Dest_Unreachable (3), Time_Exceeded (11)
# default UDP base: 33434 to base+nhops-1
#
# For incoming traceroute.
# Message Types: OUTGOING Dest_Unreachable (3), Time_Exceeded (11)
# To block this, deny OUTGOING 3 and 11

# 0: echo-reply (pong)
# 3: destination-unreachable, port-unreachable, fragmentation-needed, etc.
# 4: source-quench
# 5: redirect
# 8: echo-request (ping)
# 11: time-exceeded
# 12: parameter-problem

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 0 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 3 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 4 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 11 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 12 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        s $MY_ISP 8 -d $IPADDR -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
```

```
        -s $IPADDR 0 -d $MY_ISP -j ACCEPT
  ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 3 -d $MY_ISP -j ACCEPT
  ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 4 -d $ANYWHERE -j ACCEPT
  ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 8 -d $ANYWHERE -j ACCEPT
  ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 12 -d $ANYWHERE -j ACCEPT
  ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 11 -d $MY_ISP -j ACCEPT
# ..........
  # UDP INCOMING TRACEROUTE
  # traceroute usually uses -S 32769:65535 -D 33434:33523

  ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -s $MY_ISP $TRACEROUTE_SRC_PORTS \
        -d $IPADDR $TRACEROUTE_DEST_PORTS -j ACCEPT -l

  ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -s $ANYWHERE $TRACEROUTE_SRC_PORTS \
        -d $IPADDR $TRACEROUTE_DEST_PORTS -j DENY -l

# ..........

  # ..........
  # DNS forwarding, caching only nameserver (53)
  # ..........

  # server to server query or response
  # Caching only name server only requires UDP, not TCP

  ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -s $NAMESERVER_1 53 \
        -d $IPADDR 53 -j ACCEPT

  ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
        -s $IPADDR 53 \
        -d $NAMESERVER_1 53 -j ACCEPT

  ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -s $NAMESERVER_2 53 \
        -d $IPADDR 53 -j ACCEPT
```

```
    ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
            -s $IPADDR 53 \
            -d $NAMESERVER_2 53 -j ACCEPT

    # DNS client (53)
    # ..............
    ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
            -s $NAMESERVER_1 53 \
            -d $IPADDR $UNPRIVPORTS -j ACCEPT

    ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
            -s $IPADDR $UNPRIVPORTS \
            -d $NAMESERVER_1 53 -j ACCEPT

    ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
            -s $NAMESERVER_2 53 \
            -d $IPADDR $UNPRIVPORTS -j ACCEPT

    ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
            -s $IPADDR $UNPRIVPORTS \
            -d $NAMESERVER_2 53 -j ACCEPT

# TCP client to server requests are allowed by the protocol
# if UDP requests fail. This is rarely seen. Usually, clients
# use TCP as a secondary nameserver for zone transfers from
# their primary nameservers, and as hackers.

    ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
            -s $NAMESERVER_1 53 \
            -d $IPADDR $UNPRIVPORTS -j ACCEPT

    ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
            -s $IPADDR $UNPRIVPORTS \
            -d $NAMESERVER_1 53 -j ACCEPT

    ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
            -s $NAMESERVER_2 53 \
            -d $IPADDR $UNPRIVPORTS -j ACCEPT

    ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
            -s $IPADDR $UNPRIVPORTS \
            -d $NAMESERVER_2 53 -j ACCEPT

# ..........................................................................
    # TCP accept only on selected ports
    # ..............................
    # ..........................................................................

    # SSH server (22)
    # ..............
```

```
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE $UNPRIVPORTS \
        -d $IPADDR 22 -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $IPADDR 22 \
        -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE $SSH_PORTS \
        -d $IPADDR 22 -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $IPADDR 22 \
        -d $ANYWHERE $SSH_PORTS -j ACCEPT

# ------------------------------------------------------------

# HTTP server (80)
# ---------------

ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE $UNPRIVPORTS \
        -d $IPADDR 80 -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $IPADDR 80 \
        -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

# ------------------------------------------------------------

# HTTPS server (443)
# -----------------

ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE $UNPRIVPORTS \
        -d $IPADDR 443 -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $IPADDR 443 \
        -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

# ------------------------------------------------------------

# SYSLOG server (514)
# ------------------

# Provides full remote logging. Using this feature you're able to
# control all syslog messages on one host.
```

```
#       ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
#               -s $SYSLOG_CLIENT \
#               -d $IPADDR 514 -j ACCEPT

  # SYSLOG client (514)
  # ----------

#       ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
#               -s $IPADDR 514 \
#               -d $SYSLOG_SERVER 514 -j ACCEPT

  # ----------
  # AUTH server (113)
  # ----------

  # Reject, rather than deny, the incoming auth port. (NET-3-HOWTO)

  ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE \
        -d $IPADDR 113 -j REJECT

  # ----------

  # SMTP client (25)
  # ----------

  ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $SMTP_SERVER 25 \
        -d $IPADDR $UNPRIVPORTS -j ACCEPT

  ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
        -s $IPADDR $UNPRIVPORTS \
        -d $SMTP_SERVER 25 -j ACCEPT

  # ----------

  # FTP server (20, 21)
  # ----------

  # incoming request

  ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE $UNPRIVPORTS \
        -d $IPADDR 21 -j ACCEPT

  ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $IPADDR 21 \
        -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

  # PORT MODE data channel responses
  #
```

```
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $ANYWHERE $UNPRIVPORTS \
        -d $IPADDR 20 -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
        -s $IPADDR 20 \
        -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

# PASSIVE MODE data channel responses

ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE $UNPRIVPORTS \
        -d $IPADDR $UNPRIVPORTS -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $IPADDR $UNPRIVPORTS \
        -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

# ..........
# OUTGOING TRACEROUTE
# ..........

ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
        -s $IPADDR $TRACEROUTE_SRC_PORTS \
        -d $ANYWHERE $TRACEROUTE_DEST_PORTS -j ACCEPT

# ..........
# Enable logging for selected denied packets
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -d $IPADDR -j DENY -l

ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -d $IPADDR $PRIVPORTS -j DENY -l

ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -d $IPADDR $UNPRIVPORTS -j DENY -l

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 5 -d $IPADDR -j DENY -l

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 13:255 -d $IPADDR -j DENY -l

# ..........

;;
stop)
        echo -n "Shutting Firewalling Services: "

# Remove all existing rules belonging to this filter
```

```
ipchains -F

# Delete all user-defined chain to this filter
ipchains -X

# Reset the default policy of the filter to accept.
ipchains -P input ACCEPT
ipchains -P output ACCEPT
ipchains -P forward ACCEPT

        ;;
status)
status firewall
        ;;
restart|reload)
        $0 stop
        $0 start
        ;;
*)

echo "Usage: firewall {start|stop|status|restart|reload}"
exit 1

esac

exit 0
```

Bây giờ, chúng ta làm cho script này có thể thực thi và thay đổi các quyền mặc định của nó:

```
[root@deep /]# chmod 700 /etc/rc.d/init.d/firewall
[root@deep /]# chown 0.0 /etc/rc.d/init.d/firewall
```

Tạo các liên kết rc.d cho Firewall của bạn với các lệnh sau:

```
[root@deep /]# chkconfig --add firewall
[root@deep /]# chkconfig --level 345 firewall on
```

Bây giờ, các firewall rules của bạn đã được cấu hình để dùng System V init (System V init có nhiệm vụ khởi động tất cả các tiến trình (processes) bình thường mà nó cần thiết vào thời điểm khởi động) và nó sẽ được tự động khởi tạo ở mỗi thời điểm mà máy server của bạn khởi động lại.

- Để tắt (stop) cơ chế firewall trên hệ thống của bạn, sử dụng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/firewall stop
Shutting Firewalling Services:      [ OK ]
```

- Để bật cơ chế firewall trên hệ thống của bạn, sử dụng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/firewall start
Starting Firewalling Services:      [ OK ]
```

Cấu hình tập tin script "/etc/rc.d/init.d/firewall" cho Mail Server

Đây là tập tin cấu hình script cho Mail Server của bạn. Việc cấu hình cho phép lưu thông không giới hạn trên Loopback interface, ICMP, DNS Serve và Client (53), SSH Serer (22), SMTP Server và Client (25), IMAP Server (143), và các yêu cầu OUTGOING TRACEOUTE bởi mặc định.

Nếu bạn không muốn một số dịch vụ được liệt kê trong các tập tin firewall rules cho Mail Server mà bạn đã mở bởi mặc định, ghi chú trong tập tin script với một ký tự "#" ở đầu dòng. Nếu bạn muốn mở một số dịch vụ khác mà nó hiện đang là dòng ghi chú thì bạn chỉ xóa ký tự "#" ở đầu của các dòng đó.

Tạo tập tin firewall script (touch /etc/rc.d/init.d/firewall) trên Mail Server và thêm vào nội dung sau:

```
#!/bin/sh
#
# ------------------------------------------------------------------
# Last modified by Gerhard Mourani: 04-25-2000
# ------------------------------------------------------------------
# Copyright (C) 1997, 1998, 1999 Robert L. Ziegler
#
# Permission to use, copy, modify, and distribute this software and its
# documentation for educational, research, private and non-profit purposes,
# without fee, and without a written agreement is hereby granted.
# This software is provided as an example and basis for individual firewall
# development. This software is provided without warranty.
#
# Any material furnished by Robert L. Ziegler is furnished on an
# "as is" basis. He makes no warranties of any kind, either expressed
# or implied as to any matter including, but not limited to, warranty
# of fitness for a particular purpose, exclusivity or results obtained
# from use of the material.
# ------------------------------------------------------------------
#
# Invoked from /etc/rc.d/init.d/firewall.
# chkconfig: - 60 95
# description: Starts and stops the IPCHAINS Firewall \
# used to provide Firewall network services.

# Source function library.
```

```
. /etc/rc.d/init.d/functions
# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
if [ ${NETWORKING} = "no" ]
then
        exit 0
fi

if [ ! -x /sbin/ipchains ]; then
        exit 0
fi

# See how we were called.
case "$1" in
        start)
                echo -n "Starting Firewalling Services: "

# Some definitions for easy maintenance.

# ..........................................................................
# EDIT THESE TO SUIT YOUR SYSTEM AND ISP.

EXTERNAL_INTERFACE="eth0"              # Internet connected interface
LOOPBACK_INTERFACE="lo"                # Your local naming convention
IPADDR="my.ip.address"                 # Your IP address
ANYWHERE="any/0"                       # Match any IP address
NAMESERVER_1="my.name.server.1"        # Everyone must have at least one
NAMESERVER_2="my.name.server.2"        # Your secondary name server
MY_ISP="my.isp.address.range/24"       # ISP & NOC address range
SMTP_SERVER="my.smtp.server"           # Your Mail Hub Server.
SYSLOG_SERVER="syslog.internal.server" # Your syslog internal server
SYSLOG_CLIENT="sys.int.client.range/24" # Your syslog internal client range
LOOPBACK="127.0.0.0/8"                 # Reserved loopback address range
CLASS_A="10.0.0.0/8"                   # Class A private networks
CLASS_B="172.16.0.0/12"                # Class B private networks
CLASS_C="192.168.0.0/16"               # Class C private networks
CLASS_D_MULTICAST="224.0.0.0/4"        # Class D multicast addresses
CLASS_E_RESERVED_NET="240.0.0.0/5"     # Class E reserved addresses
```

```
BROADCAST_SRC="0.0.0.0"                 # Broadcast source address
BROADCAST_DEST="255.255.255.255"        # Broadcast destination address
PRIVPORTS="0:1023"                      # Well known, privileged port range
UNPRIVPORTS="1024:65535"                # Unprivileged port range

# ------------------------------------------------------------------
# SSH starts at 1023 and works down to 513 for
# each additional simultaneous incoming connection.
SSH_PORTS="1022:1023"                   # range for SSH privileged ports

# traceroute usually uses -S 32769:65535 -D 33434:33523
TRACEROUTE_SRC_PORTS="32769:65535"
TRACEROUTE_DEST_PORTS="33434:33523"

# ------------------------------------------------------------------
# Default policy is DENY
# Explicitly accept desired INCOMING & OUTGOING connections

  # Remove all existing rules belonging to this filter
  ipchains -F

  # Clearing all current rules and user defined chains
  ipchains -X

  # Set the default policy of the filter to deny.
  # Don't even bother sending an error message back.
  ipchains -P input DENY
  ipchains -P output DENY
  ipchains -P forward DENY

# ------------------------------------------------------------------
  # LOOPBACK

  # Unlimited traffic on the loopback interface.
  ipchains -A input -i $LOOPBACK_INTERFACE -j ACCEPT
  ipchains -A output -i $LOOPBACK_INTERFACE -j ACCEPT

# ------------------------------------------------------------------
# Network Ghouls
# Deny access to jerks
```

```
  # /etc/rc.d/rc.firewall.blocked contains a list of
  # ipchains -A input -i $EXTERNAL_INTERFACE -s address -j DENY
  # rules to block from any access.
  # Refuse any connection from problem sites
  #if [ -f /etc/rc.d/rc.firewall.blocked ]; then
  # . /etc/rc.d/rc.firewall.blocked
  #fi

# ......................................................................
# SPOOFING & BAD ADDRESSES
# Refuse spoofed packets.
# Ignore blatantly illegal source addresses.
# Protect yourself from sending to bad addresses.

  # Refuse spoofed packets pretending to be from the external address.
  ipchains -A input -i $EXTERNAL_INTERFACE -s $IPADDR -j DENY -l

  # Refuse packets claiming to be to or from a Class A private network
  ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_A -j DENY -l
  ipchains -A input -i $EXTERNAL_INTERFACE -d $CLASS_A -j DENY -l
  ipchains -A output -i $EXTERNAL_INTERFACE -s $CLASS_A -j REJECT -l
  ipchains -A output -i $EXTERNAL_INTERFACE -d $CLASS_A -j REJECT -l

  # Refuse packets claiming to be to or from a Class B private network
  ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_B -j DENY -l
  ipchains -A input -i $EXTERNAL_INTERFACE -d $CLASS_B -j DENY -l
  ipchains -A output -i $EXTERNAL_INTERFACE -s $CLASS_B -j REJECT -l
  ipchains -A output -i $EXTERNAL_INTERFACE -d $CLASS_B -j REJECT -l

  # Refuse packets claiming to be to or from a Class C private network
#       ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_C -j DENY -l
#       ipchains -A input -i $EXTERNAL_INTERFACE -d $CLASS_C -j DENY -l
#       ipchains -A output -i $EXTERNAL_INTERFACE -s $CLASS_C -j REJECT -l
#       ipchains -A output -i $EXTERNAL_INTERFACE -d $CLASS_C -j REJECT -l

  # Refuse packets claiming to be from the loopback interface
  ipchains -A input -i $EXTERNAL_INTERFACE -s $LOOPBACK -j DENY -l
  ipchains -A output -i $EXTERNAL_INTERFACE -s $LOOPBACK -j REJECT -l

  # Refuse broadcast address SOURCE packets
  ipchains -A input -i $EXTERNAL_INTERFACE -s $BROADCAST_DEST -j DENY -l
  ipchains -A input -i $EXTERNAL_INTERFACE -d $BROADCAST_SRC -j DENY -l

  # Refuse Class D multicast addresses (in.h) (NET-3-HOWTO)
  # Multicast is illegal as a source address.
  # Multicast uses UDP.
  ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_D_MULTICAST -j DENY -l

  # Refuse Class E reserved IP addresses
```

```
ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_E_RESERVED_NET -j
DENY -l

# refuse addresses defined as reserved by the IANA
# 0.*.*.*, 1.*.*.*, 2.*.*.*, 5.*.*.*, 7.*.*.*, 23.*.*.*, 27.*.*.*
# 31.*.*.*, 37.*.*.*, 39.*.*.*, 41.*.*.*, 42.*.*.*, 58-60.*.*.*
# 65-95.*.*.*, 96-126.*.*.*, 197.*.*.*, 201.*.*.* (?), 217-223.*.*.*
ipchains -A input -i $EXTERNAL_INTERFACE -s 1.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 2.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 5.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 7.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 23.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 27.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 31.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 37.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 39.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 41.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 42.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 58.0.0.0/7 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 60.0.0.0/8 -j DENY -l

#65: 01000001 - /3 includes 64 - need 65-79 spelled out
ipchains -A input -i $EXTERNAL_INTERFACE -s 65.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 66.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 67.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 68.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 69.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 70.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 71.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 72.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 73.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 74.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 75.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 76.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 77.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 78.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 79.0.0.0/8 -j DENY -l

#80: 01010000 - /4 masks 80-95
ipchains -A input -i $EXTERNAL_INTERFACE -s 80.0.0.0/4 -j DENY -l

# 96: 01100000 - /4 makses 96-111
ipchains -A input -i $EXTERNAL_INTERFACE -s 96.0.0.0/4 -j DENY -l

#126: 01111110 - /3 includes 127 - need 112-126 spelled out
ipchains -A input -i $EXTERNAL_INTERFACE -s 112.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 113.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 114.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 115.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 116.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 117.0.0.0/8 -j DENY -l
```

```
ipchains -A input -i $EXTERNAL_INTERFACE -s 118.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 119.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 120.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 121.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 122.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 123.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 124.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 125.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 126.0.0.0/8 -j DENY -l

#217: 11011001 - /5 includes 216 - need 217-219 spelled out
ipchains -A input -i $EXTERNAL_INTERFACE -s 217.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 218.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 219.0.0.0/8 -j DENY -l

#223: 11011111 - /6 masks 220-223
ipchains -A input -i $EXTERNAL_INTERFACE -s 220.0.0.0/6 -j DENY -l

# ------------------------------------------------------------------
# ICMP

# To prevent denial of service attacks based on ICMP bombs, filter
# incoming Redirect (5) and outgoing Destination Unreachable (3).
# Note, however, disabling Destination Unreachable (3) is not
# advisable, as it is used to negotiate packet fragment size.

# For bi-directional ping.
#       Message Types: Echo_Reply (0), Echo_Request (8)
#       To prevent attacks, limit the src addresses to your ISP range.
#

# For outgoing traceroute.
#       Message Types: INCOMING Dest_Unreachable (3), Time_Exceeded (11)
#       default UDP base: 33434 to base+nhops-1
#

# For incoming traceroute.
#       Message Types: OUTGOING Dest_Unreachable (3), Time_Exceeded (11)
#       To block this, deny OUTGOING 3 and 11

# 0: echo-reply (pong)
# 3: destination-unreachable, port-unreachable, fragmentation-needed, etc.
# 4: source-quench
# 5: redirect
# 8: echo-request (ping)
# 11: time-exceeded
# 12: parameter-problem

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 0 -d $IPADDR -j ACCEPT
```

```
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 3 -d $IPADDR -j ACCEPT

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 4 -d $IPADDR -j ACCEPT

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 11 -d $IPADDR -j ACCEPT

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 12 -d $IPADDR -j ACCEPT

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $MY_ISP 8 -d $IPADDR -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 0 -d $MY_ISP -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 3 -d $MY_ISP -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 4 -d $ANYWHERE -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 8 -d $ANYWHERE -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 12 -d $ANYWHERE -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
        -s $IPADDR 11 -d $MY_ISP -j ACCEPT

# ..........
  # UDP INCOMING TRACEROUTE
  # traceroute usually uses -S 32769:65535 -D 33434:33523

  ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -s $MY_ISP $TRACEROUTE_SRC_PORTS \
        -d $IPADDR $TRACEROUTE_DEST_PORTS -j ACCEPT -l

  ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -s $ANYWHERE $TRACEROUTE_SRC_PORTS \
        -d $IPADDR $TRACEROUTE_DEST_PORTS -j DENY -l

# ..........

  # DNS server
  # ..........
  # DNS: full server
  # server/client to server query or response
```

```
    ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
            -s $ANYWHERE $UNPRIVPORTS \
            -d $IPADDR 53 -j ACCEPT

    ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
            -s $IPADDR 53 \
            -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

      # DNS client & Zone Transfers (53)
      # ..............
      ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
            -s $NAMESERVER_1 53 \
            -d $IPADDR $UNPRIVPORTS -j ACCEPT

      ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
            -s $IPADDR $UNPRIVPORTS \
            -d $NAMESERVER_1 53 -j ACCEPT

      ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
            -s $NAMESERVER_1 53 \
            -d $IPADDR $UNPRIVPORTS -j ACCEPT

      ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
            -s $IPADDR $UNPRIVPORTS \
            -d $NAMESERVER_1 53 -j ACCEPT

# ..........................................................................
    # TCP accept only on selected ports
    # ..............................
    # ..........................................................
    # SSH server (22)
    # ..............

    ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
            -s $ANYWHERE $UNPRIVPORTS \
            -d $IPADDR 22 -j ACCEPT

    ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
            -s $IPADDR 22 \
            -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

    ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
            -s $ANYWHERE $SSH_PORTS \
            -d $IPADDR 22 -j ACCEPT

    ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
            -s $IPADDR 22 \
            -d $ANYWHERE $SSH_PORTS -j ACCEPT

    # ..........................................................
```

```
# AUTH server (113)
# ----------------

# Reject, rather than deny, the incoming auth port. (NET-3-HOWTO)

ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE \
        -d $IPADDR 113 -j REJECT

# ------------------------------------------------------------
# SYSLOG server (514)
# ----------------

# Provides full remote logging. Using this feature you're able to
# control all syslog messages on one host.

#       ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
#                -s $SYSLOG_CLIENT \
#                -d $IPADDR 514 -j ACCEPT

# SYSLOG client (514)
# ----------------
#       ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
#                -s $IPADDR 514 \
#                -d $SYSLOG_SERVER 514 -j ACCEPT

# ------------------------------------------------------------

# SMTP server (25)
# ---------------

ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE $UNPRIVPORTS \
        -d $IPADDR 25 -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $IPADDR 25 \
        -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

# SMTP client (25)
# ---------------

ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $ANYWHERE 25 \
        -d $IPADDR $UNPRIVPORTS -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
        -s $IPADDR $UNPRIVPORTS \
        -d $ANYWHERE 25 -j ACCEPT
```

```
# ------------------------------------------------------------

# IMAP server (143)
# ----------------

ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -s $ANYWHERE $UNPRIVPORTS \
        -d $IPADDR 143 -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
        -s $IPADDR 143 \
        -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

# POP server (110)
# ----------------
#       ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
#               -s $ANYWHERE $UNPRIVPORTS \
#               -d $IPADDR 110 -j ACCEPT
#       ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
#               -s $IPADDR 110 \
#               -d $ANYWHERE $UNPRIVPORTS -j ACCEPT

# ------------------------------------------------------------
# OUTGOING TRACEROUTE
# -------------------

ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
        -s $IPADDR $TRACEROUTE_SRC_PORTS \
        -d $ANYWHERE $TRACEROUTE_DEST_PORTS -j ACCEPT

# ------------------------------------------------------------
# Enable logging for selected denied packets
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
        -d $IPADDR -j DENY -l

ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -d $IPADDR $PRIVPORTS -j DENY -l

ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
        -d $IPADDR $UNPRIVPORTS -j DENY -l

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 5 -d $IPADDR -j DENY -l

ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
        -s $ANYWHERE 13:255 -d $IPADDR -j DENY -l

# ------------------------------------------------------------

;;
stop)
```

```
        echo -n "Shutting Firewalling Services: "

# Remove all existing rules belonging to this filter
ipchains –F

# Delete all user-defined chain to this filter
ipchains –X

# Reset the default policy of the filter to accept.
ipchains -P input ACCEPT
ipchains -P output ACCEPT
ipchains -P forward ACCEPT

        ;;
status)
        status firewall
        ;;
restart|reload)
        $0 stop
        $0 start
        ;;
*)
        echo "Usage: firewall {start|stop|status|restart|reload}"
        exit 1
esac

exit 0
```

Bây giờ, chúng ta làm cho script này có thể thực thi và thay đổi các quyền mặc định của chúng:

```
[root@deep /]# chmod 700 /etc/rc.d/init.d/firewall
[root@deep /]# chown 0.0 /etc/rc.d/init.d/firewall
```

Tạo các biểu tượng liên kết trong rc.d cho Firewall của bạn với các lệnh sau:

```
[root@deep /]# chkconfig --add firewall
[root@deep /]# chkconfig --level 345 firewall on
```

Và bây giờ, các firewall rules của bạn đã được cấu hình để dùng System V init (System V init có nhiệm vụ khởi động tất cả các tiến trình (processes) bình thường mà nó cần thiết vào thời điểm khởi động) và nó sẽ được tự động khởi tạo ở mỗi thời điểm mà máy server của bạn khởi động lại.

- Để tắt (stop) cơ chế firewall trên hệ thống của bạn, sử dụng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/firewall stop
Shutting Firewalling Services:      [ OK ]
```

- Để bật cơ chế firewall trên hệ thống của bạn, sử dụng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/firewall start
Starting Firewalling Services:      [ OK ]
```

Tóm tắt về IPCHAINS

ipchains - IP firewall administration
ipchains -[ADC] chain rule-specification [options]
ipchains -[RI] chain rulenum rule-specification [options]
ipchains -D chain rulenum [options]
ipchains -[LFZNX] [chain] [options]
ipchains -P chain target [options]
ipchains -M [-L | -S] [options]

MÔ TẢ

Ipchains thường được dùng để cài đặt, bảo trì và kiểm tra các quy tắc (rules) IP firewall trong kernel của hệ điều hành Linux. Các quy tắc hay nguyên lý này có thể được chia làm 4 nhóm danh mục khác nhau: IP input chain, IP output chain, IP forwarding chain, và user defined chains.

Đối với mỗi nhóm danh mục này, một bảng qui tắc được duy trì, bất kỳ nhóm nào cũng có thể tham chiếu đến các qui tắc người sử dụng đã định nghĩa (user defined chains). Xem ipfw(4) để biết thêm thông tin chi tiết.

MỤC ĐÍCH

Một qui tắc Firewall chỉ rõ các tiêu chuẩn cho một gói tin (packet), và một đích đến. Nếu mà packet không so khớp với rule đầu tiên trong chain, thì sẽ được kiểm tra với rule kế tiếp; nếu vẫn không xứng hợp, thì rule kế tiếp được chỉ định bởi giá trị của đích (target), target có thể là tên của một chain do người dùng định nghĩa (user-defined chain), hoặc một trong các giá trị đặc biệt ACCEPT, DENY, MASQ, REDIRECT hoặc RETURN.

ACCEPT có nghĩa là cho phép packet đi qua. DENY có nghĩa là làm cho packet ngừng lại trên firewall hoặc gateway. REJECT nghĩa là giống như làm ngừng lại, nhưng dễ dàng cho việc debug hơn, vì lẽ rằng một thông điệp ICMP được gửi trả về lại bên gửi chỉ rõ rằng packet bị dừng lại. (Ghi chú rằng DENY và REJECT là giống nhau đối với packet ICMP). Chú ý : điều này không đúng; đặt ICMP là REJECT sẽ làm cho không thể hiểu port ICMP để gửi.

MASQ chỉ thích hợp cho các chain forward và user-defined và chỉ có thể được dùng khi kernel được biên dịch với tùy chọn CONFIG_IP_MASQUERADE được chọn. Với điều này, các packet sẽ được giả mạo khi chúng được xuất phát từ các máy (host) cục bộ. Sau đó, các packet cũng sẽ được đảo ngược lại như ban đầu tại các máy (host) nội bộ. Quá trình này được làm một cách tự động.

REDIRECT chỉ hợp pháp cho các chain input và user-defined và chỉ có thể được dùng khi kernel của Linux được biên dịch với tùy chọn CONFIG_IP_TRANSPARENT_PROXY được định nghĩa. Với điều này, các packet sẽ được gửi một lần nữa tới một socket cục bộ, thậm chí chúng được gửi tới một host ở xa. Nếu port được gửi lại (redirection port) được chỉ định là 0, và nó là giá trị mặc nhiên, thì port gởi lại một lần nữa của một packet sẽ được dùng như là port gửi lại lần nữa. Nếu mục đích đến này được dùng, thì thêm một thông số tùy chọn (số thứ tự của port (port number)) có thể được cung cấp.

Nếu cuối một chain do người sử dụng định nghĩa (user-defined chain) có ảnh hưởng đến, hay một nguyên tắc với đích RETURN được tìm thấy, khi đó nguyên tắc kế tiếp trong chuỗi trước (đang gọi) được kiểm tra. Nếu cuối của một qui tắc xây dựng bên trong (buildin chain) bị ảnh hưởng, hay một nguyên tắc trong builtin chain với đích RETURN được thấy, thì đích đến được chỉ rõ bởi chính sách chuỗi (chain policy) xác định số phận của packet.

CÁC TÙY CHỌN

Các tùy chọn được nhận biết bởi ipchains có thể được chia làm nhiều nhóm khác nhau.

CÁC LỆNH

Các tùy chọn này chỉ rõ hành động cụ thể để thực hiện; chỉ một trong những tùy chọn này được ghi rõ trên dòng lệnh, ngoại trừ những thứ khác được chỉ rõ bên dưới. Đối với tất cả các lời giải thích dài dòng của dòng lệnh và các tên tùy chọn, bạn chỉ cần dùng đủ các ký tự để bảo đảm rằng ipchains có thể phân biệt nó từ tất cả các tùy chọn khác.

-A, --append

Nối tiếp một hoặc nhiều rule đến cuối của chain được chọn. Khi các tên nguồn và/hoặc đích phân giải nhiều hơn một địa chỉ, một rule sẽ được thêm vào cho mỗi sự kết hợp địa chỉ hợp lý.

-D, --delete

Xoá một hoặc nhiều rule chuỗi được chọn. Có hai cách thể hiện của lệnh này: rule có thể được chỉ rõ như là một số trong chuỗi (bắt đầu từ 1 cho rule thứ nhất) hay một rule để xứng hợp.

-R, --replace

Thay thế một rule trong một chuỗi được chọn. Nếu các tên nguồn và/hoặc đích phân giải tới nhiều địa chỉ, lệnh sẽ sai. Các rule được đếm bắt đầu từ 1.

-I, --insert

Chèn vào một hay nhiều rule vào chain được chọn y như đưa ra số rule. Như vậy, nếu số thứ tự rule là 1 thì một hay nhiều rule được chèn vào ở đầu của chain.

-L, --list

Liệt kê tất cả các rule trong chain được chọn. Nếu không có chain nào được chọn, thì tất cả các chain được liệt kê. Điều này cũng hợp lệ để chỉ rõ tùy chọn –Z (zero), trong trường hợp không có chain nào có thể được chỉ rõ. Kết quả xuất ra chính xác cũng bị tác động bởi các thông số khác.

-F, --flush

Làm cho bằng phẳng chain được chọn. Điều này tương đương với việc xóa tất cả các rule lần lượt từng cái một.

-Z, --zero

Không có đếm packet và byte trong tất cả các chuỗi. Điều này hợp lý khi chỉ rõ tùy chọn –L, ––list (list), để xem các bộ đếm ngay tức thì trước khi chúng được xóa; nếu điều này được thực hiện xong, khi đó không có chain cụ thể nào được chỉ rõ (chúng sẽ hiển thị và xóa tất cả).

-N, --new-chain

Tạo một chain mới do người dùng định nghĩa (user-defined chain) với tên được đưa ra. Tên chain mới này không trùng với một tên chain đã có sẵn.

-X, --delete-chain

Xoá một chain do người dùng định nghĩa được chỉ rõ. Chain cần xoá phải không được tham chiếu (nếu có bạn phải xóa hay thay thế các rule tham chiếu trước khi chain được xóa). Nếu không tham số nào được cung cấp, nó sẽ cố gắng xoá mọi chain không được xây dựng bên trong (non-builtin chain).

-P, --policy

Đặt chính sách cho chain để đưa ra target. Xem phần TARGETS sẽ cho các target hợp lệ. Chỉ các chain không phải do người dùng định nghĩa có thể có các chính sách.

-M, --masquerading

Tùy chọn này cho phép xem các kết nối giả mạo hiện hành (trong liên kết với tùy chọn -L) hay phải đặt các thông số giả mạo kernel (với tùy chọn -S).

-S, -- set tcp tcpfin udp

Thay đổi các giá trị timeout được dùng cho việc giả mạo (masquerade). Lệnh này luôn luôn có 3 tham số. Mô tả các giá trị timeout (tính bằng giây) đối với các TCP session, các phiên TCP (TCP session) sau khi nhận một packet FIN, và các packet UDP tương ứng. Một giá trị timeout 0 nghĩa là giá trị timeout hiện tại của mục tương ứng được duy trì lâu dài. Tùy chọn này thì chỉ được phép trong việc kết hợp với cờ hiệu -M.

-C, -- check

Kiểm tra packet cho sẵn tương phản với chain được chọn. Điều này là rất tiện ích cho việc thử nghiệm, cũng giống như các tập tin thi hành kernel (kernel routine) thường dùng để kiểm tra các packet mạng "thật" thì được sử dụng kiểm tra packet này. Nó cũng có thể được dùng để kiểm tra chain do người dùng định nghĩa (user-defined chain) cũng như builtin chain. Tương tự các thông số được dùng chỉ rõ các firewall rule cũng được dùng đặt packet để kiểm tra. Trong trường hợp đặc biệt, các cờ hiệu -s (source), -d (destination), -p (protocol) và -i (interface) là bắt buộc.

-h, --help

Cung cấp một sự diễn giải (rất ngắn gọn) của cú pháp lệnh. Nếu được theo sau bởi từ icmp, khi đó một danh sách các tên ICMP được liệt kê.

-V, --version

Đơn giản xuất ra số phiên bản ipchains.

CÁC THAM SỐ

Các tham số sau tạo một sự định rõ rule (như được dùng trong các lệnh add, delete, replace, append và check).

-p, --protocol[!] protocol

Giao thức của rule hay của packet để kiểm tra. Giao thức được chỉ rõ có thể là một trong những giao thức như tcp, udp, icmp, hay tất cả, hoặc nó có thể là một giá trị số. Cũng như tên của một giao thức từ /etc/protocols là được phép. Một thông số "!" trước giao thức đảo có tác dụng đảo ngược việc kiểm tra. Số 0 là tương đương với tất cả. Tất cả giao thức sẽ xứng hợp với tất cả các giao thức và là mặc định khi tùy chọn này bị bỏ sót. Tất cả không thể được dùng trong sự kết hợp với lệnh check.

-s , --source, --src [!] address[/mask] [!] [port[:port]]

Sự định rõ nguồn. Địa chỉ cũng có thể là một hostname, một tên mạng, hoặc là một địa chỉ IP rõ ràng. Mặt nạ (mask) cũng có thể là một mặt nạ mạng hay là một số rõ ràng, chỉ rõ giá trị một số ở bên trái của 1 của mặt nạ mạng. Vì vậy, một mặt nạ của 24 là tương đương với 255.255.255.0. Một thông số "!" trước địa chỉ cụ thể đảo ngược ý nghĩa của địa chỉ.

Nguồn có thể bao gồm một chi tiết cổng (port) hay loại ICMP. Điều này cũng có thể là một tên dịch vụ, một số cổng, một số loại ICMP, hay là một trong các tên ICMP được hiển thị trong lệnh.

Ipchains -h icmp

Chú ý rằng nhiều tên ICMP tham chiếu đến cả hai mã và loại, có nghĩa là một mã ICMP sau cờ hiệu –d là không hợp lệ. Dựa vào đoạn này, một cổng có nghĩa là sự chỉ định một cổng hay một loại ICMP. Một miền giá trị bao gồm cũng có thể được chỉ rõ, dùng định dạng port:port. Nếu cổng đầu tiên là bị bỏ qua, "0" là được lấy; nếu cổng sau cùng cũng bị bỏ qua thì "65535" bị lấy.

Các cổng chỉ có thể được chỉ rõ trong việc kết hợp với các giao thức tcp, udp, hay icmp. Một dấu "!" trước một cổng cụ thể đảo ngược ý nghĩa. Khi việc kiểm tra lệnh được chỉ rõ, chính xác một port được yêu cầu, và nếu cờ hiệu –f (fragment) là được chỉ rõ, không có port nào được phép.

--source-port [!] [port[:port]]

Điều này cho phép sự định rõ riêng lẻ của port nguồn hay miền giá trị port. Xem mô tả của cờ hiệu –s ở trên để biết thêm chi tiết. Cờ hiệu --sport là một bí danh cho tùy chọn này.

-d, --destination, --dst [!] address[/mask] [!] [port[:port]]

Sự chỉ định đích đến. Xem sự diễn giải của cờ hiệu –s (source) để biết sự diễn giải trong của cú pháp. Đối với ICMP, mà không có cổng nào, một "destination port" tham chiếu tới số mã ICMP.

--destination-port [!] [port[:port]]

Điều này cho phép sự chỉ định riêng biệt của các port. Xem mô tả cờ hiệu –s để biết chi tiết. Cờ hiệu --dport là một bí danh cho tùy chọn này.

--icmp-type [!] typename

Điều này cho phép sự định rõ của loại ICMP (dùng tùy chọn –h icmp để xem các tên loại ICMP hợp lệ). Việc này thường tiện lợi hơn gắn nó vào đích đến cụ thể.

-j, --jump target

Điều này chỉ ra mục đích của rule; nghĩa là phải làm gì nếu packet so khớp với nó. Targer có thể là một user-defined chain hoặc là một trong các target đặc biệt quyết định số phận của packet ngay lập tức. Nếu tùy chọn này bị bỏ qua trong một rule, sau khi so khớp với rule sẽ không có sự ảnh hưởng trên số phận của packet, nhưng các bộ đếm trên nguyên tắc sẽ được gia tăng.

-i, --interface [!] name

Tên tùy ý của một giao tiếp mà một packet được nhận đi qua (dành cho các packet đi vào input chain), hoặc các packet dự định được gửi (dành cho các packet đi vào forward hay output chains). Khi tùy chọn này bị bỏ qua, thì một chuỗi rỗng được lấy, mà nó có một ý nghĩa đặc biệt và sẽ phù hợp với bất kỳ tên giao tiếp nào. Khi thông số [!] được dùng trước tên giao tiếp, ý nghĩa sẽ bị đảo ngược. Nếu tên giao tiếp kết thúc bằng một dấu "+", khi đó bất kỳ giao tiếp nào mà bắt đầu với tên này sẽ phù hợp.

[!] –f, --fragment

Điều này có nghĩa là rule chỉ tham chiếu tới các phân đoạn thứ hai và cao hơn của các packet được phân đoạn. Bởi vì không có cách để báo cho các port nguồn hay đích của một packet (hay loại ICMP), ví dụ như một packet sẽ không phù hợp bất kỳ các rule ghi rõ chúng. Khi thông số [!] đứng trước cờ hiệu "-f", ý nghĩa được đảo ngược.

CÁC TÙY CHỌN KHÁC

Các tùy chọn được thêm vào như sau có thể được chỉ rõ:

-b, --bidirectional

Chế độ bidirectional mode. Nguyên tắc sẽ phù hợp với các packet IP trong cả hai hướng; điều này có cùng tác động như việc lặp lại rule với nguồn và đích được đảo ngược. Ghi chú rằng điều này không có nghĩa là khi bạn cho phép TCP syn packet ra ngoài, rule –b sẽ cho phép các non-SYN packet trở lại bên trong: rule đảo ngược chính xác giống như rule bạn đã gõ vào. Điều này có nghĩa là nó thường tốt hơn để đơn giản tránh cờ hiệu –b và chỉ ra các rule một cách rõ ràng.

-v, --verbose

Xuất ra verbose. Tùy chọn này tạo một danh sách lệnh trình bày các địa chỉ giao tiếp, các tùy chọn rule (nếu có), và các mặt nạ TOS. Khi sử dụng trong sự kết hợp với –M, thông tin liên quan với các số thứ tự delta cũng sẽ được liệt kê. Đối với việc nối vào, chèn vào, xóa và thay thế, việc này sinh ra các thông tin chi tiết trên một hay nhiều các rule được in ra.

-n, --numeric

Xuất ra số. Các địa chỉ IP và các số cổng sẽ được in ra theo dạng số. Mặc nhiên, chương trình sẽ cố gắng hiển thị chúng như là các tên host, các tên mạng, hay các dịch vụ (bất cứ lúc nào được yêu cầu).

-l, --log

Ghi vào logfile tất cả các packet xứng hợp. Khi tùy chọn này được đặt cho một rule, Linux kernel sẽ in ra một số thông tin của tất cả các packet thông qua printk().

-o, --output [maxsize]

Sao chép các matching packet tới thiết bị lưu trữ của người dùng (userspace). Phần này chủ yếu cho các nhà phát triển phần mềm, người mà muốn thực hiện với tác động firewall trong userspace. Thông số tùy chọn maxsize có thể được dùng giới hạn số byte tối đa từ packet mà nó được sao chép. Tùy chọn này chỉ hợp lệ nếu kernel đã được biên dịch với CONFIG_IP_FIRE-WALL_NETLINK được đặt.

-m, --mark markvalue

Đánh dấu các packet xứng hợp (matching packet). Các packet cũng được đánh dấu với một giá trị unsigned 32 bit mà nó có thể thay đổi theo cách chúng được xử lý bên trong. Nếu bạn không là một kernel hacker, bạn sẽ không cần chú trọng vấn đề này. Nếu chuỗi giá trị đánh dấu bắt đầu với một + hay - , khi đó giá trị này sẽ được thêm vào hay trừ ra từ giá trị được đánh dấu hiện hành của packet (mà bắt đầu bằng 0).

-t, --TOS andmask xormask

Các mặt nạ (mask) thường dùng cho việc hiệu chỉnh các vùng TOS trong IP header. Khi một packet so khớp với một rule, vùng TOS của nó là bitwise đầu tiên

được 'and' với mask đầu tiên và kết quả của việc này sẽ bitwise được 'xor' với mask thứ hai. Các mask sẽ được chỉ rõ như các giá trị Hexa 8-bit. Giống như LSB của vùng TOS phải không được thay đổi (RFC 1349), các giá trị TOS gây ra nó bị thay đổi sẽ bị loại trừ, cũng như bất kỳ các rule khi luôn đặt nhiều hơn một bit TOS. Các rule có thể đặt nhiều bit TOS cho các packet nào đó, kết quả là các cảnh báo (gửi tới stdout) có thể bị lờ qua nếu bạn biết rằng các packet với các giá trị TOS đó sẽ không bao giờ tham chiếu được đến rule đó. Có thể thấy được việc thao tác TOS là có hàm ý ra hiệu nếu target của các rule là DENY hay REJECT.

-x, --exact

Mở rộng các số. Hiển thị giá trị chính xác của các bộ đếm của packet và byte, thay vì chỉ làm tròn số bằng K (bội số của 1000), M (bội số của 1000K) hay G (bội số của 1000M). Tùy chọn này liên quan đối với lệnh –L.

[!] –y, --syn

Chỉ so khớp các packet TCP với SYN bit được thiết đặt và các bit ACK, FIN được xóa. Nhiều packet thường được dùng để yêu cầu sự khởi tạo kết nối TCP; Ví dụ, chặn các packet đi vào một một giao tiếp (interface) sẽ ngăn chặn các kết nối TCP đang đến, nhưng các kết nối TCP ra ngoài sẽ không bị tác động. Tùy chọn này chỉ có nghĩa khi kiểu giao thức được đặt là TCP. Nếu cờ hiệu "!" đứng trước "-y", thì ý nghĩa của tùy chọn này bị đảo ngược.

--line-numbers

Khi liệt kê các rule, thêm số dòng tới nơi bắt đầu của mỗi rule, tương ứng với vị trí của rule trong chain.

--no-warning

Vô hiệu hóa tất cả các câu cảnh báo.

Các nguyên tắc tính toán bây giờ là được tích hợp dễ dàng vào trong các input và output chain; bạn có mẫu ví dụ như thế này:

```
ipchains -N acctin
ipchains -N acctout
ipchains -N acctio
ipchains -I input -j acctio
ipchains -I input -j acctin
ipchains -I output -j acctio
ipchains -I output -j acctout
```

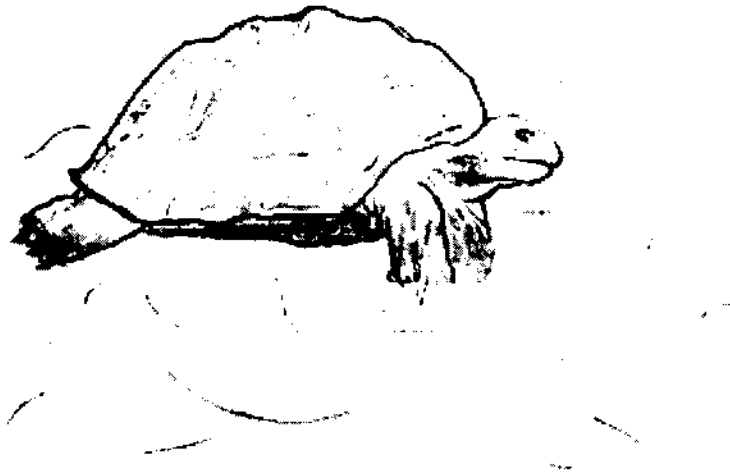
Điều này tạo ba chain do người dùng định nghĩa, acctin, acctout, và acctio, chúng chứa đựng bất kỳ các nguyên tắc tính toán nào (các nguyên tắc này nên được chỉ rõ không có một cờ hiệu –j, vì lẽ đó các packet dễ dàng xuyên qua mà chúng không bị hư hại).

Chương 8:

LIÊN KẾT FIREWALL VỚI HỖ TRỢ TÍNH NĂNG GIẢ MẠO (MASQUERADING) IP VÀ CHUYỂN TIẾP (FORWARDING)

Trong chương này chúng ta sẽ đề cập đến một số vấn đề:

- Xây dựng một kernel hỗ trợ tính năng giả mạo (Masquerading) IP và chuyển tiếp (Forwarding).
- Sự cấu hình tập tin script cho Gateway Server.
- Từ chối truy cập từ một vài địa chỉ.
- Công cụ quản trị IPCHAINS.



Giả mạo và chuyển tiếp IP trong Linux

Tổng quan

Trái với những cấu hình ví dụ trong Chương 7, việc cấu hình một Linux Server để giả mạo và chuyển tiếp lưu thông từ các host bên trong mạng riêng có những địa chỉ IP không đăng ký (ví dụ 192.168.1.0/24) tới mạng bên ngoài (mạng Internet) yêu cầu một tiến trình cài đặt đặc biệt đối với nhân (Kernel) hệ điều hành và các scripts tập tin cấu hình firewall của bạn. Kiểu cài đặt này cũng được biết như một Gateway Server (một máy phục vụ như một gateway điều khiển sự lưu thông bên trong với bên ngoài mạng). Bạn chỉ thiết lập cấu hình gateway nếu bạn có dự định và cần loại dịch vụ này, và đây là lý do cấu hình các tập tin scripts cho Gateway Server là phần chính trong chương này.

Xây dựng một Kernel với Firewall hỗ trợ giả mạo IP và chuyển tiếp

Một lần nữa, việc đầu tiên bạn cần làm là đảm bảo rằng kernel được xây dựng với các option như "Network firewall" và "Firewalling" được chọn. Trong Linux kernel phiên bản 2.2.14 bạn cần đảm bảo rằng bạn đã trả lời "Yes" cho những câu hỏi sau:

Networking options:

Network firewalls (CONFIG_FIREFALL) [N] Y

IP:Firewalling (CONFIG_IP_FIREWALL) [N] Y

IP:TCP syncookie support (CONFIG_SYN_COOKIES) [N] Y

Ghi chú: *Nếu bạn đi theo phần Linux kernel và đã biên dịch lại kernel của bạn, thì những tùy chọn "Network firewalls, IP:Firewalling, và IP:TCP syncookies support" được trình bày ở trên đã được cài đặt sẵn.*

Các tùy chọn IP Masquerading và IP ICMP Masquerading chỉ được yêu cầu cho cấu hình Gateway Server.

IP:Masquerading (CONFIG_IP_MASQUERADE) [N] Y

IP:ICMP Masquerading (CONFIG_IP_MASQUERADE_ICMP) [N] Y

Ghi chú: *Chỉ Gateway Server của bạn cần các tùy chọn "IP:Masquerading" và "IP:ICMP Masquerading". Điều này được yêu cầu để giả mạo bên trong mạng nội bộ của bạn với bên ngoài.*

Masquerade có ý nghĩa là nếu một trong những máy tính thuộc mạng cục bộ với máy Linux của bạn làm việc như một firewall (hoặc gateway) muốn gởi thông tin gì đó ra bên ngoài thì bạn có thể giả dạng như những thông tin được gởi từ máy Linux server này. Nói cách khác Linux sẽ chuyển các thông tin đó đến đích, những thông tin này được xem giống như là chính firewall đã gởi thông tin đó đi.

Nó làm việc theo cách: nếu máy bên ngoài trả lời, thì Linux firewall sẽ tự động chuyển thông tin đến máy tính tương ứng bên trong mạng cục bộ. Theo cách này, những máy tính trên mạng cục bộ của bạn không nhìn thấy hoàn toàn thế giới bên ngoài, mặc dù chúng có thể gởi thông tin và có thể nhận được trả lời. điều này cho phép có thể có những máy tính thuộc mạng cục bộ tham gia vào mạng Internet dù chúng không có những địa chỉ IP được đăng ký chính thức.

Sự giả mạo IP sẽ chỉ làm việc nếu chức năng IP forwarding được cho phép trên hệ thống bạn. Đặc tính này theo mặc định là không có hiệu lực và bạn có thể cho phép nó với lệnh sau :

Đối với Red Hat Linux 6.1

- Để cho phép đặc tính IP forwarding trên server, bạn cần thực hiện lệnh sau:

```
[root@deep /]# echo "1" > /proc/sys/net/ipv4/ip_forward
```

Bạn có thể thêm dòng lệnh ở trên vào trong tập tin script"/etc/rc.d/rc.local" như vậy đặc tính IP forwarding được tự động kích hoạt mỗi khi bạn khởi động server. Trong Linux Red Hat 6.1 đặc tính này cũng có thể được hoàn thành bằng cách thay đổi dòng lệnh sau trong tập tin "/etc/sysconfig/network":

```
FORWARD_IPV4="false"
```

được đổi thành:

```
FORWARD_IPV4="yes"
```

- Bạn phải khởi động lại mạng của bạn để việc thay đổi có hiệu lực:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Bringing up interface lo [ OK ]
Bringing up interface eth0 [ OK ]
Bringing up interface eth1
```

Như vậy bạn có thể hoặc thêm dòng lệnh "echo "1" > /proc/sys/net/ipv4/ip_forward" vào trong tập tin script"rc.local" hoặc đổi giá trị của dòng "FORWARD_IPV4=false" sang "yes" trong tập tin /etc/sysconfig/network. Tuy nhiên cách thứ hai được sử dụng thông dụng hơn.

Đối với Red Hat Linux 6.2

- Để đặc tính IPv4 forwarding được kích hoạt trên hệ thống Red Hat 6.2, bạn thực hiện lệnh sau:

Hiệu chỉnh tập tin "/etc/sysctl.conf" và thêm vào dòng sau:

```
# Enable packet forwarding
net.ipv4.ip_forward = 1
```

Tất nhiên bạn phải khởi động lại mạng để việc thay đổi có hiệu lực. Lệnh sau sẽ khởi động lại mạng:

- Khởi động lại tất cả các thiết bị mạng trên hệ thống dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters      [ OK ]
Bringing up interface lo        [ OK ]
Bringing up interface eth0      [ OK ]
Bringing up interface eth1      [ OK ]
```

Ghi chú: *Dòng IP Forwarding ở trên chỉ được yêu cầu nếu bạn trả lời "yes" trong tùy chọn của kernel "IP:Masquerading (CONFIG_IP_MASQUERADE)" và chọn server của bạn hoạt động như một Gateway và Masquerade cho bên trong mạng của bạn.*

Nếu bạn mở chức năng IP Masquerading, thì các module như ip_masq_ftp.o (dịch vụ truyền tập tin FTP), ip_masq_irc.o (dịch vụ IRC CHAT), ip_masq_quake.o, Ip_masq_vdolive.o (kết nối video VDOLive), ip_masq_cuseeme.o (CU - SeeMe Broadcasts) và ip_masq_raudio.o (dịch vụ tải RealAudio xuống) sẽ được biên dịch tự động. Những dịch vụ này thực sự cần thiết được giả mạo để các protocol làm việc. Đồng thời, bạn sẽ cần xây dựng một modularized kernel và trả lời "YES" cho tùy chọn "Enable loadable module support (CONFIG_MODULES)" thay cho một monolithic kernel để có thể sử dụng các chức năng giả dạng và những module như ip_masq_ftp.o trên máy Gateway Server của bạn (xem chương Linux Kernel ở trên để biết thêm chi tiết).

Cơ bản việc giả dạng cho "IP:masquerading" ở trên chỉ điều khiển các gói tin theo giao thức TCP và UDP (và những lỗi ICMP cho những kết nối sẵn có). Tùy chọn IP:ICMP Masquerading hỗ trợ cho việc giả mạo các gói ICMP, chẳng hạn như lệnh ping hoặc những phương pháp dò tìm được sử dụng bởi chương trình tracer của Windows 95.

Ghi chú: *Các server khác như: Web server, Mail server không cần những tùy chọn này vì chúng đã được gán địa chỉ IP thực (Địa chỉ IP đã đăng ký với các tổ chức quản lý tên miền) và chúng cũng không có chức năng làm Gateway server cho mạng nội bộ.*

Một vài điểm cần lưu ý:

Bạn cần lưu ý các điểm sau nếu bạn muốn kết nối một cách an toàn hệ thống mạng của bạn với mạng Internet:

- Gateway của bạn không nên chạy bất kỳ ứng dụng nào ngoài những ứng dụng thật sự cần thiết.
- Gateway của bạn cần giới hạn chính xác loại và số lượng các giao thức cho phép xuyên qua nó (những giao thức tiềm tàng cung cấp những lỗ bảo mật, chẳng hạn như FTP và telnet).
- Bất kỳ hệ thống nào chứa đựng thông tin bí mật hoặc nhạy cảm thì không nên cho phép tiếp cận trực tiếp từ Internet.

Sự cấu hình tập tin script "/etc/rc.d/init.d/firewall" cho Gateway Server

Đây là cấu hình tập tin cho Gateway server của chúng ta. Cấu hình này cho phép không giới hạn trên giao diện Loopback, ICMP, DNS Server và Client(port 53), SSH Server và Client(port 22), HTTP Server và Client (port 80), HTTPS Server và Client (port 443), POP Server (port 110), NNTP News Client (port 119), SMTP Server và Client (port 25), IMAP Server (port 143), IRC Client (port 6667), ICQ Client (port 4000), FTP Client (port: 20, 21), RealAudio / QuickTime Client, và OUTGOING TRACEROUTE.

Bây giờ đến phần cấu hình tập tin script"/etc/rc.d/init.d/firewall". Nếu bạn không muốn những dịch vụ nào được liệt kê bên trong tập tin này cần cho Gateway Server hãy thêm dấu "#" đầu dòng dịch vụ đó. Tương tự, hãy loại bỏ dấu "#" ở đầu dòng dịch vụ nào bạn muốn sử dụng. Nếu bạn đã cấu hình Masquerading trên server của bạn, đừng quên loại bỏ dấu "#" ở những môđun cần thiết để giả mạo các dịch vụ tương ứng của chúng như ip_masq_irc.o, ip_masq_raudio.o,... Bạn sẽ tìm thấy những dòng mô tả này trong phần "MODULES MASQUERADING" trong tập tin script này.

Để cấu hình, bạn dùng lệnh sau:

```
[root@deep /]#touch /etc/rc.d/init.d/firewall
```

Sau khi tập tin firewall được tạo, chúng ta hãy thêm vào tập tin này với nội dung sau:

```
#!/bin/sh
#
# ------------------------------------------------------------------
# Last modified by Gerhard Mourani: 04-25-2000
# ------------------------------------------------------------------
# Copyright (C) 1997, 1998, 1999 Robert L. Ziegler
#
# Permission to use, copy, modify, and distribute this software and its
# documentation for educational, research, private and non-profit purposes,
# without fee, and without a written agreement is hereby granted.
# This software is provided as an example and basis for individual firewall
# development. This software is provided without warranty.
#
# Any material furnished by Robert L. Ziegler is furnished on an
# "as is" basis. He makes no warranties of any kind, either expressed
# or implied as to any matter including, but not limited to, warranty
# of fitness for a particular purpose, exclusivity or results obtained
# from use of the material.
```

```
# ------------------------------------------------------------------------
#
# Invoked from /etc/rc.d/init.d/firewall.
# chkconfig: - 60 95
# description: Starts and stops the IPCHAINS Firewall \
# used to provide Firewall network services.
# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network
# Check that networking is up.
if [ ${NETWORKING} = "no" ]
then
exit 0
fi
if [ ! -x /sbin/ipchains ]; then
exit 0
fi
# See how we were called.
case "$1" in
start)
echo -n "Starting Firewalling Services: "
# Some definitions for easy maintenance.
# ------------------------------------------------------------------------
# EDIT THESE TO SUIT YOUR SYSTEM AND ISP.
EXTERNAL_INTERFACE="eth0" # Internet connected interface
LOCAL_INTERFACE_1="eth1" # Internal LAN interface
LOOPBACK_INTERFACE="lo" # Your local naming convention
IPADDR="my.ip.address" # địa chỉ IP của bạn
LOCALNET_1="192.168.1.0/24" # Whatever private range you use
IPSECSG="my.ipsecsg.address" # Space separated list of remote VPN gateways
FREESWANVI="ipsec0" # Space separated list of virtual interfaces
ANYWHERE="any/0" # Match any IP address
NAMESERVER_1="my.name.server.1" # Everyone must have at least one
NAMESERVER_2="my.name.server.2" # Your secondary name server
MY_ISP="my.isp.address.range/24" # ISP & NOC address range
SMTP_SERVER="my.smtp.server" # Your Mail Hub Server.
POP_SERVER="my.pop.server" # External pop server, if any
NEWS_SERVER="my.news.server" # External news server, if any
SYSLOG_SERVER="syslog.internal.server" # Your syslog internal server
```

```
LOOPBACK="127.0.0.0/8" # Reserved loopback address range
CLASS_A="10.0.0.0/8" # Class A private networks
CLASS_B="172.16.0.0/12" # Class B private networks
CLASS_C="192.168.0.0/16" # Class C private networks
CLASS_D_MULTICAST="224.0.0.0/4" # Class D multicast addresses
CLASS_E_RESERVED_NET="240.0.0.0/5" # Class E reserved addresses
BROADCAST_SRC="0.0.0.0" # Broadcast source address
BROADCAST_DEST="255.255.255.255" # Broadcast destination address
PRIVPORTS="0:1023" # Well known, privileged port range
UNPRIVPORTS="1024:65535" # Unprivileged port range
# ------------------------------------------------------------------
# SSH starts at 1023 and works down to 513 for
# each additional simultaneous incoming connection.
SSH_PORTS="1022:1023" # range for SSH privileged ports
# traceroute usually uses -S 32769:65535 -D 33434:33523
TRACEROUTE_SRC_PORTS="32769:65535"
TRACEROUTE_DEST_PORTS="33434:33523"

# ------------------------------------------------------------------
# Default policy is DENY
# Explicitly accept desired INCOMING & OUTGOING connections
# Remove all existing rules belonging to this filter
ipchains –F
# Clearing all current rules and user defined chains
ipchains –X
# Set the default policy of the filter to deny.
# Don't even bother sending an error message back.
ipchains -P input DENY
ipchains -P output DENY
ipchains -P forward DENY
# set masquerade timeout to 10 hours for tcp connections
ipchains -M -S 36000 0 0
# Don't forward fragments. Assemble before forwarding.
ipchains -A output -f -i $LOCAL_INTERFACE_1 -j DENY
# ------------------------------------------------------------------
# MODULES MASQUERADING
# Uncomment bellow all modules lines that you need
# These modules are necessary to masquerade their respective services.
/sbin/modprobe ip_masq_ftp
/sbin/modprobe ip_masq_raudio ports=554,7070,7071,6970,6971
/sbin/modprobe ip_masq_irc
```

```
#/sbin/modprobe ip_masq_vdolive
#/sbin/modprobe ip_masq_cuseeme
#/sbin/modprobe ip_masq_quake
# ------------------------------------------------------------------
# LOOPBACK
# Unlimited traffic on the loopback interface.
ipchains -A input -i $LOOPBACK_INTERFACE -j ACCEPT
ipchains -A output -i $LOOPBACK_INTERFACE -j ACCEPT
# ------------------------------------------------------------------
# Network Ghouls
# Deny access to jerks
# /etc/rc.d/rc.firewall.blocked contains a list of
# ipchains -A input -i $EXTERNAL_INTERFACE -s address -j DENY
# rules to block from any access.
# Refuse any connection from problem sites
# if [ -f /etc/rc.d/rc.firewall.blocked ]; then
# . /etc/rc.d/rc.firewall.blocked
#fi
# ------------------------------------------------------------------
# SPOOFING & BAD ADDRESSES
# Refuse spoofed packets.
# Ignore blatantly illegal source addresses.
# Protect yourself from sending to bad addresses.
# Refuse spoofed packets pretending to be from the external address.
ipchains -A input -i $EXTERNAL_INTERFACE -s $IPADDR -j DENY -l

# Refuse packets claiming to be to or from a Class A private network
ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_A -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -d $CLASS_A -j DENY -l
ipchains -A output -i $EXTERNAL_INTERFACE -s $CLASS_A -j REJECT -l
ipchains -A output -i $EXTERNAL_INTERFACE -d $CLASS_A -j REJECT -l
# Refuse packets claiming to be to or from a Class B private network
ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_B -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -d $CLASS_B -j DENY -l
ipchains -A output -i $EXTERNAL_INTERFACE -s $CLASS_B -j REJECT -l
ipchains -A output -i $EXTERNAL_INTERFACE -d $CLASS_B -j REJECT -l
# Refuse packets claiming to be to or from a Class C private network
# ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_C -j DENY -l
# ipchains -A input -i $EXTERNAL_INTERFACE -d $CLASS_C -j DENY -l
# ipchains -A output -i $EXTERNAL_INTERFACE -s $CLASS_C -j REJECT -l
# ipchains -A output -i $EXTERNAL_INTERFACE -d $CLASS_C -j REJECT -l
```

```
# Refuse packets claiming to be from the loopback interface
ipchains -A input -i $EXTERNAL_INTERFACE -s $LOOPBACK -j DENY -l
ipchains -A output -i $EXTERNAL_INTERFACE -s $LOOPBACK -j REJECT -l
# Refuse broadcast address SOURCE packets
ipchains -A input -i $EXTERNAL_INTERFACE -s $BROADCAST_DEST -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -d $BROADCAST_SRC -j DENY -l
# Refuse Class D multicast addresses (in.h) (NET-3-HOWTO)
# Multicast is illegal as a source address.
# Multicast uses UDP.
ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_D_MULTICAST -j DENY -l
# Refuse Class E reserved IP addresses
ipchains -A input -i $EXTERNAL_INTERFACE -s $CLASS_E_RESERVED_NET -j DENY -l
# refuse addresses defined as reserved by the IANA
# 0.*.*.*, 1.*.*.*, 2.*.*.*, 5.*.*.*, 7.*.*.*, 23.*.*.*, 27.*.*.*
# 31.*.*.*, 37.*.*.*, 39.*.*.*, 41.*.*.*, 42.*.*.*, 58-60.*.*.*
# 65-95.*.*.*, 96-126.*.*.*, 197.*.*.*, 201.*.*.* (?), 217-223.*.*.*

ipchains -A input -i $EXTERNAL_INTERFACE -s 1.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 2.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 5.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 7.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 23.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 27.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 31.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 37.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 39.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 41.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 42.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 58.0.0.0/7 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 60.0.0.0/8 -j DENY -l
#65: 01000001 - /3 includes 64 - need 65-79 spelled out
ipchains -A input -i $EXTERNAL_INTERFACE -s 65.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 66.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 67.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 68.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 69.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 70.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 71.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 72.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 73.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 74.0.0.0/8 -j DENY -l
```

```
ipchains -A input -i $EXTERNAL_INTERFACE -s 75.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 76.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 77.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 78.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 79.0.0.0/8 -j DENY -l
#80: 01010000 - /4 masks 80-95
ipchains -A input -i $EXTERNAL_INTERFACE -s 80.0.0.0/4 -j DENY -l
# 96: 01100000 - /4 makses 96-111
ipchains -A input -i $EXTERNAL_INTERFACE -s 96.0.0.0/4 -j DENY -l
#126: 01111110 - /3 includes 127 - need 112-126 spelled out
ipchains -A input -i $EXTERNAL_INTERFACE -s 112.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 113.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 114.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 115.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 116.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 117.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 118.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 119.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 120.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 121.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 122.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 123.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 124.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 125.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 126.0.0.0/8 -j DENY -l
#217: 11011001 - /5 includes 216 - need 217-219 spelled out
ipchains -A input -i $EXTERNAL_INTERFACE -s 217.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 218.0.0.0/8 -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -s 219.0.0.0/8 -j DENY -l
#223: 11011111 - /6 masks 220-223
ipchains -A input -i $EXTERNAL_INTERFACE -s 220.0.0.0/6 -j DENY -l
# ------------------------------------------------------------------
# ICMP

# To prevent denial of service attacks based on ICMP bombs, filter
# incoming Redirect (5) and outgoing Destination Unreachable (3).
# Note, however, disabling Destination Unreachable (3) is not
# advisable, as it is used to negotiate packet fragment size.
# For bi-directional ping.
# Message Types: Echo_Reply (0), Echo_Request (8)
# To prevent attacks, limit the src addresses to your ISP range.
#
# For outgoing traceroute.
```

```
# Message Types: INCOMING Dest_Unreachable (3), Time_Exceeded (11)
# default UDP base: 33434 to base+nhops-1
#
# For incoming traceroute.
# Message Types: OUTGOING Dest_Unreachable (3), Time_Exceeded (11)
# To block this, deny OUTGOING 3 and 11
# 0: echo-reply (pong)
# 3: destination-unreachable, port-unreachable, fragmentation-needed, etc.
# 4: source-quench

# 5: redirect
# 8: echo-request (ping)
# 11: time-exceeded
# 12: parameter-problem
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
-s $ANYWHERE 0 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
-s $ANYWHERE 3 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
-s $ANYWHERE 4 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
-s $ANYWHERE 11 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
-s $ANYWHERE 12 -d $IPADDR -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
-s $MY_ISP 8 -d $IPADDR -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
-s $IPADDR 0 -d $MY_ISP -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
-s $IPADDR 3 -d $MY_ISP -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
-s $IPADDR 4 -d $ANYWHERE -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
-s $IPADDR 8 -d $ANYWHERE -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
-s $IPADDR 12 -d $ANYWHERE -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p icmp \
-s $IPADDR 11 -d $MY_ISP -j ACCEPT
# ------------------------------------------------------------------
# UDP INCOMING TRACEROUTE
# traceroute usually uses -S 32769:65535 -D 33434:33523
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-s $MY_ISP $TRACEROUTE_SRC_PORTS \
-d $IPADDR $TRACEROUTE_DEST_PORTS -j ACCEPT -l
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-s $ANYWHERE $TRACEROUTE_SRC_PORTS \
-d $IPADDR $TRACEROUTE_DEST_PORTS -j DENY -l
# ------------------------------------------------------------------
```

```
# DNS server
# ---------
# DNS: full server
# server/client to server query or response
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-s $ANYWHERE $UNPRIVPORTS \
-d $IPADDR 53 -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
-s $IPADDR 53 \
-d $ANYWHERE $UNPRIVPORTS -j ACCEPT
# DNS client (53)
# ---------------
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-s $NAMESERVER_1 53 \

-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
-s $IPADDR $UNPRIVPORTS \
-d $NAMESERVER_1 53 -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-s $NAMESERVER_2 53 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
-s $IPADDR $UNPRIVPORTS \
-d $NAMESERVER_2 53 -j ACCEPT
# TCP client to server requests are allowed by the protocol
# if UDP requests fail. This is rarely seen. Usually, clients
# use TCP as a secondary nameserver for zone transfers from
# their primary nameservers, and as hackers.
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $NAMESERVER_1 53 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $NAMESERVER_1 53 -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $NAMESERVER_2 53 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $NAMESERVER_2 53 -j ACCEPT
# ------------------------------------------------------------
# TCP accept only on selected ports
# ------------------------------------------------------------
# ------------------------------------------------------------
# SSH server (22)
# ------------------------------------------------------------
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
```

```
-s $ANYWHERE $UNPRIVPORTS \
-d $IPADDR 22 -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $IPADDR 22 \
-d $ANYWHERE $UNPRIVPORTS -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
-s $ANYWHERE $SSH_PORTS \
-d $IPADDR 22 -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $IPADDR 22 \
-d $ANYWHERE $SSH_PORTS -j ACCEPT
# SSH client (22)
# ----------
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \

-s $ANYWHERE 22 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 22 -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE 22 \
-d $IPADDR $SSH_PORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $SSH_PORTS \
-d $ANYWHERE 22 -j ACCEPT
# ----------
# HTTP client (80)
# ----------
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE 80 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 80 -j ACCEPT
# ----------
# HTTPS client (443)
# ----------
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE 443 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 443 -j ACCEPT
# ----------
# POP client (110)
# ----------
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
```

```
-s $POP_SERVER 110 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $POP_SERVER 110 -j ACCEPT
# ------------------------------------------------------------
# NNTP NEWS client (119)
# ----------------------
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $NEWS_SERVER 119 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $NEWS_SERVER 119 -j ACCEPT

# ------------------------------------------------------------
# FINGER client (79)
# ------------------
# ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
# -s $ANYWHERE 79 \
# -d $IPADDR $UNPRIVPORTS -j ACCEPT
# ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
# -s $IPADDR $UNPRIVPORTS \
# -d $ANYWHERE 79 -j ACCEPT
# ------------------------------------------------------------
# SYSLOG client (514)
# -------------------
# ipchains -A output -i $LOCAL_INTERFACE_1 -p udp \
# -s SIPADDR 514 \
# -d SSYSLOG_SERVER 514 -j ACCEPT
# ------------------------------------------------------------
# AUTH server (113)
# -----------------
# Reject, rather than deny, the incoming auth port. (NET-3-HOWTO)
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
-s $ANYWHERE \
-d $IPADDR 113 -j REJECT
# AUTH client (113)
# -----------------
# ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
# -s $ANYWHERE 113 \
# -d $IPADDR $UNPRIVPORTS -j ACCEPT
# ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
# -s $IPADDR $UNPRIVPORTS \
# -d $ANYWHERE 113 -j ACCEPT
# ------------------------------------------------------------
# SMTP client (25)
# ---------------
```

```
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE 25 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 25 -j ACCEPT
# ------------------------------------------------------------------
# IRC client (6667)
# ----------------
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE 6667 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 6667 -j ACCEPT
# ------------------------------------------------------------------
# ICQ client (4000)
# ----------------
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE 2000:4000 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 2000:4000 -j ACCEPT
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-s $ANYWHERE 4000 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 4000 -j ACCEPT
# ------------------------------------------------------------------
# FTP client (20, 21)
# -------------------
# outgoing request
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE 21 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 21 -j ACCEPT
# NORMAL mode data channel
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
-s $ANYWHERE 20 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
# NORMAL mode data channel responses
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $IPADDR $UNPRIVPORTS \
```

```
-d $ANYWHERE 20 -j ACCEPT
# PASSIVE mode data channel creation
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE $UNPRIVPORTS -j ACCEPT
# PASSIVE mode data channel responses
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE $UNPRIVPORTS \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
# ------------------------------------------------------------
# RealAudio / QuickTime client


# ---------------------------
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE 554 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 554 -j ACCEPT
# TCP is a more secure method: 7070:7071
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
-s $ANYWHERE 7070:7071 \
-d $IPADDR $UNPRIVPORTS -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE 7070:7071 -j ACCEPT
# UDP is the preferred method: 6970:6999
# For LAN machines, UDP requires the RealAudio masquerading module and
# the ipmasqadm third-party software.
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-s $ANYWHERE $UNPRIVPORTS \
-d $IPADDR 6970:6999 -j ACCEPT
ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
-s $IPADDR $UNPRIVPORTS \
-d $ANYWHERE $UNPRIVPORTS -j ACCEPT
# ------------------------------------------------------------
# WHOIS client (43)
# ----------------
# ipchains -A input -i $EXTERNAL_INTERFACE -p tcp ! -y \
# -s $ANYWHERE 43 \
# -d $IPADDR $UNPRIVPORTS -j ACCEPT
# ipchains -A output -i $EXTERNAL_INTERFACE -p tcp \
# -s $IPADDR $UNPRIVPORTS \
# -d $ANYWHERE 43 -j ACCEPT
# ------------------------------------------------------------
# OUTGOING TRACEROUTE
# ------------------
```

```
ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
-s $IPADDR $TRACEROUTE_SRC_PORTS \
-d $ANYWHERE $TRACEROUTE_DEST_PORTS -j ACCEPT
# ------------------------------------------------------------------
# Unlimited traffic within the local network.
# All internal machines have access to the firewall machine.
ipchains -A input -i $LOCAL_INTERFACE_1 -s $LOCALNET_1 -j ACCEPT
ipchains -A output -i $LOCAL_INTERFACE_1 -d $LOCALNET_1 -j ACCEPT


# ------------------------------------------------------------------
# FreeS/WAN IPSec VPN
# ------------------
# If you are using the FreeSWAN IPSec VPN, you will need to fill in the
# addresses of the gateways in the IPSECSG and the virtual interfaces for
# FreeS/Wan IPSEC in the FREESWANVI parameters. Look at the beginning of
# this firewall script rules tập tin to set the parameters.
# IPSECSG is a Space separated list of remote gateways. FREESWANVI is a
# Space separated list of virtual interfaces for FreeS/Wan IPSEC
# implementation. Only include those that are actually used.
# Allow IPSEC protocol from remote gateways on external interface
# IPSEC uses three main types of packet:
# IKE uses the UDP protocol and port 500,
# ESP use the protocol number 50, and
# AH use the protocol number 51
# ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
# -s $IPSECSG -j ACCEPT
# ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
# -d $IPSECSG -j ACCEPT
# ipchains -A input -i $EXTERNAL_INTERFACE -p 50 \
# -s $IPSECSG -j ACCEPT
# ipchains -A output -i $EXTERNAL_INTERFACE -p 50 \
# -d $IPSECSG -j ACCEPT
# ipchains -A input -i $EXTERNAL_INTERFACE -p 51 \
# -s $IPSECSG -j ACCEPT
# ipchains -A output -i $EXTERNAL_INTERFACE -p 51 \
# -d $IPSECSG -j ACCEPT
# Allow all traffic to FreeS/WAN Virtual Interface
# ipchains -A input -i $FREESWANVI \
# -s $ANYWHERE \
# -d $ANYWHERE -j ACCEPT
# ipchains -A output -i $FREESWANVI \
# -s $ANYWHERE \
# -d $ANYWHERE -j ACCEPT
# Forward anything from the FreeS/WAN virtual interface IPSEC tunnel
# ipchains -A forward -i $FREESWANVI \
# -s $ANYWHERE \
# -d $ANYWHERE -j ACCEPT
```

```
# Disable IP spoofing protection to allow IPSEC to work properly
# echo 0 > /proc/sys/net/ipv4/conf/ipsec0/rp_filter
# echo 0 > /proc/sys/net/ipv4/conf/eth0/rp_filter
# ------------------------------------------------------------------
# Masquerade internal traffic.
# All internal traffic is masqueraded externally.
ipchains -A forward -i $EXTERNAL_INTERFACE -s $LOCALNET_1 -j MASQ


# ------------------------------------------------------------------
# Enable logging for selected denied packets
ipchains -A input -i $EXTERNAL_INTERFACE -p tcp \
-d $IPADDR -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-d $IPADDR $PRIVPORTS -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-d $IPADDR $UNPRIVPORTS -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
-s $ANYWHERE 5 -d $IPADDR -j DENY -l
ipchains -A input -i $EXTERNAL_INTERFACE -p icmp \
-s $ANYWHERE 13:255 -d $IPADDR -j DENY -l
# ------------------------------------------------------------------
;;
stop)
echo -n "Shutting Firewalling Services: "
# Remove all existing rules belonging to this filter
ipchains -F
# Delete all user-defined chain to this filter
ipchains -X
# Reset the default policy of the filter to accept.
ipchains -P input ACCEPT
ipchains -P output ACCEPT
ipchains -P forward ACCEPT
;;
status)
status firewall
;;
restart|reload)
$0 stop
$0 start
;;
*)
echo "Usage: firewall {start|stop|status|restart|reload}"
exit 1
esac
exit 0
```

Bây giờ, bạn dùng các lệnh sau để làm cho tập tin script trên có khả năng thực thi và đổi quyền mặc định trên tập tin này:

```
[root@deep /]# chmod 700 /etc/rc.d/init.d/firewall
[root@deep /]# chown 0.0 /etc/rc.d/init.d/firewall
```

Tạo biểu tượng trong rc.d liên kết với Firewall của bạn bằng lệnh:

```
[root@deep /]# chkconfig --add firewall
[root@deep /]# chkconfig --level 345 firewall on
```

Bây giờ, các rule firewall của bạn đã được cấu hình để làm việc với hệ thống System V init (Hệ thống System V init chịu trách nhiệm khởi động tất cả các tiến trình thông thường (normal processes) cần thiết chạy lúc khởi động máy. và nó sẽ tự động được khởi động lại mỗi lần server bạn reboot.

- Để dừng hệ thống firewall, bạn sử dụng lệnh sau :

```
[root@deep /] /etc/rc.d/init.d/firewall stop
Shutting Firewalling Services : [ Ok ]
```

- Để khởi động hệ thống firewall, bạn sử dụng lệnh sau :

```
[ root@deep /] /etc/rc.d/init.d/firewall start
starting Firewalling Services : [ Ok ]
```

Từ chối truy nhập vào hệ thống từ một vài địa chỉ IP

Đôi khi bạn muốn một số địa chỉ IP không được phép truy nhập vào bất kỳ tài nguyên của server bạn. Bạn có thể làm việc đó bằng cách tạo ra tập tin rc.firewall.blocked trong thư mục "/etc/rc.d/" và uncomment (bỏ dấu "#") trước các dòng sau trong tập tin script của firewall của bạn:

- Để thay đổi cấu hình tập tin scriptfirewall, bạn dùng lệnh sau:

```
[ root@deep /] vi /etc/rc.d/init.d/firewall
```

Sau đó uncomment (bỏ dấu "#") trước các dòng sau:

```
if [ -f /etc/rc.d/rc.firewall.blocked ]; then
. /etc/rc.d/rc.firewall.blocked
fi
```

Tạo tập tin "rc.firewall.blocked" bằng lệnh:

```
[root@deep /] touch /etc/rc.d/rc.firewall.blocked
```

Sau đó bạn chỉ việc liệt kê tất cả các địa chỉ IP mà bạn muốn cấm truy nhập vào trong tập tin này, mỗi một dòng là một địa chỉ IP.

Ví dụ: Những địa chỉ sau sẽ bị cấm truy nhập:

```
204.254.45.9
187.231.11.5
```

Các tài liệu tham khảo thêm để biết thêm chi tiết, mời bạn tham khảo thêm các trang:

$ ipchains (8) - Quản trị IP firewall

$ ipchains-restore (8) - Phục hồi các chain IP firewall từ stdin

$ ipchains-save (8) - Lưu các chain IP firewall ra stdout

Công cụ quản trị IPCHAINS

Các lệnh được liệt kê bên dưới là một số công cụ rất thường dùng đã có sẵn trên máy của bạn. Bạn nên kiểm tra lại các trang man page và đọc thêm các tài liệu để biết thêm thông tin chi tiết.

ipchains

Công cụ ipchains được sử dụng để quản trị hệ thống firewall trên Linux. Chúng ta có thể sử dụng nó để tạo lập các tập tin chứa các rules của firewall, như là chúng ta đang làm trong sách này. Một khi những rules của firewall đã được tạo ra, chúng ta có thể bảo trì, theo dõi với nhiều lệnh có trong nhân Linux (Linux kernel).

Để liệt kê tất cả các rules trong chain được chọn, bạn dùng lệnh:

```
[root@deep /]# ipchains -L
```

Lệnh trên sẽ liệt kê tất c các rules của chain được chọn. Nếu không có chain nào được chọn, thì tất cả các chains đều được liệt kê.

```
[root@deep /]# ipchains -L input
```

Lệnh trên sẽ liệt kê tất cả các input rules mà chúng ta đã cấu hình trong chain được chọn.

```
[root@deep /]# ipchains -L output
```

Lệnh trên sẽ liệt kê tất cả các output rules mà chúng ta đã cấu hình trong chain được chọn.

```
[root@deep /]# ipchains -L forward
```

Lệnh trên sẽ liệt kê tất cả các rules chuyển tiếp trong chain được chọn. Tất nhiên việc này chỉ làm khi chúng ta đã cấu hình giả mạo IP trên server (Thông thường cho Gateway Server).

```
[root@deep /]# ipchains -ML
```

Lệnh với tùy chọn trên cho phép xem xét các kết nối được giả mạo hiện tại. Bạn phải cấu hình masquerading trên server của bạn thì lệnh này mới hoạt động (một lần nữa, chỉ dành cho gateway servers).

```
[root@deep /]# ipchains -nL
```

Lệnh trên sẽ liệt kê tất cả các rules ở dạng numeric output. Tất cả các địa chỉ IP và port numbers sẽ được liệt kê ở dạng numeric.

PHẦN 4

SỰ THAM KHẢO CÁC PHẦN MỀM LIÊN QUAN

Trong phần này chúng ta sẽ đề cập đến các chương sau:

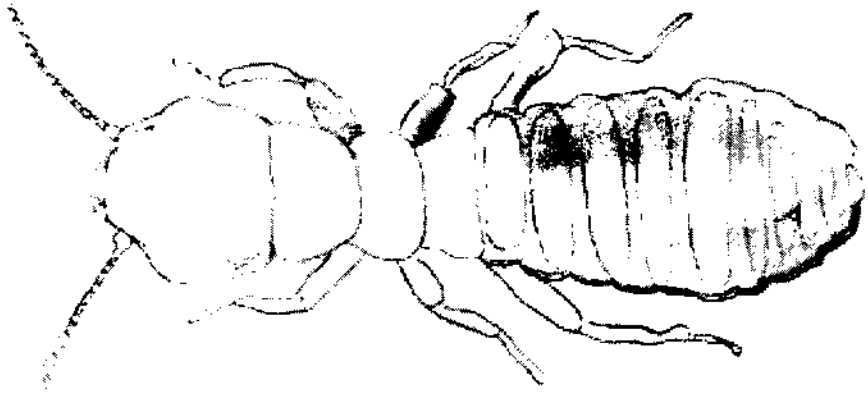
- **Chức năng biên dịch.**
- **Phần mềm bảo mật (Công cụ giám sát).**
- **Phần mềm bảo mật (Các dịch vụ mạng).**
- **Phần mềm bảo mật (Tính toàn vẹn hệ thống).**
- **Phần mềm bảo mật (Sự quản lý và sự giới hạn).**
- **Phần mềm server (Dịch vụ mạng BIND/DNS).**
- **Phần mềm server (Dịch vụ Mail).**
- **Phần mềm server (Dịch vụ mã hóa).**
- **Phần mềm server (Dịch vụ cơ sở dữ liệu).**
- **Phần mềm server (Dịch vụ Proxy).**
- **Phần mềm server (Dịch vụ Web).**
- **Cài đặt Apache với các thành phần tùy chọn.**
- **Phần mềm server (Dịch vụ chia sẻ tập tin).**

Chương 9:

CHỨC NĂNG BIÊN DỊCH
(Compiler Functionality)

Trong chương này chúng ta sẽ đề cập đến các vấn đề sau:

- Các gói dữ liệu (packages) cần thiết.
- Tại sao chúng ta chọn sử dụng tarball?
- Biên dịch các phần mềm trên hệ thống của bạn.
- Xây dựng và cài đặt phần mềm trên hệ thống của bạn.
- Soạn thảo các tập tin với trình soạn thảo vi.
- Một vài ghi chú sau cùng.



Chức năng biên dịch trong Linux

Tổng quan

Chúng ta bây giờ đang ở một trong những điểm thú vị nhất – Ở đây chúng ta sẽ biên dịch và cài đặt tất cả các dịch vụ mà chúng ta muốn cung cấp trong Linux Server. Trước khi chúng ta bắt đầu giải thích bằng cách nào để biên dịch và cài đặt phần mềm Server với tất cả sự bảo mật và tối ưu hóa cần thiết, những phần mà chúng ta sẽ cần trên Linux Server của chúng ta, một điều rất quan trọng là biết những lệnh và các chương trình chúng ta sẽ thường sử dụng để làm công việc này. Trước hết, chúng ta phải đảm bảo rằng chúng ta có những packages cần thiết dùng để làm trình biên dịch (compiler) trên hệ thống chúng ta. Những packages này phải được cài đặt trên Server của bạn nếu không bạn sẽ không thể biên dịch các chương trình trên Server.

Các gói dữ liệu cần thiết

Những gói dữ liệu (packages) được liệt kê sau là cần thiết để hệ thống Linux của bạn có thể biên dịch được sau khi bạn biên dịch lại Kernel của hệ điều hành. Những packages này đã có trên đĩa cài đặt Red Hat Linux 6.1 hoặc 6.2 trong thư mục RPMS. Nếu những phần mềm chưa được cài đặt, bạn hãy cài đặt chúng như sau:

Đưa đĩa cài đặt Linux thứ 1 vào ổ CD-ROM và chuyển vào thư mục Red Hat/RPMS:

```
[root@deep /]# mount /dev/cdrom  /mnt/cdrom/
[root@deep /]# cd /mnt/cdrom/Red Hat/RPMS/
```

Version 6.1	Version 6.2
autoconf-2.13-5.noarch.rpm	autoconf-2.13-5.noarch.rpm
m4-1.4-12.i386.rpm	m4-1.4-12.i386.rpm
automake-1.4-5.noarch.rpm	automake-1.4-6.noarch.rpm
dev86-0.14.9-1.i386.rpm	dev86-0.15.0-2.i386.rpm
bison-1.28-1.i386.rpm	bison-1.28-2.i386.rpm
byacc-1.9-11.i386.rpm	byacc-1.9-12.i386.rpm
cdecl-2.5-9.i386.rpm	cdecl-2.5-10.i386.rpm
cpp-1.1.2-24.i386.rpm	cpp-1.1.2-30.i386.rpm
cproto-4.6-2.i386.rpm	cproto-4.6-3.i386.rpm
ctags-3.2-1.i386.rpm	ctags-3.4-1.i386.rpm
egcs-1.1.2-24.i386.rpm	egcs-1.1.2-30.i386.rpm

ElectricFence-2.1-1.i386.rpm	ElectricFence-2.1-3.i386.rpm
flex-2.5.4a-7.i386.rpm	flex-2.5.4a-9.i386.rpm
gdb-4.18-4.i386.rpm	gdb-4.18-11.i386.rpm
kernel-headers-2.2.12-20.i386.rpm	kernel-headers-2.2.14-5.0.i386.rpm
glibc-devel-2.1.2-11.i386.rpm	glibc-devel-2.1.3-15.i386.rpm
make-3.77-6.i386.rpm	make-3.78.1-4.i386.rpm
patch-2.5-9.i386.rpm	patch-2.5-10.i386.rpm

***Ghi chú**: Cách tốt nhất là cài đặt (install) các phần mềm được diễn giải ở trên cùng một lần với nhau để tránh nhận những thông báo lỗi về sự phụ thuộc của các package trong quá trình install. Nếu bạn đã tuân theo tất cả những bước trong chương 2 "Installation of your Linux Server", thì tất cả các packages này đã được cài đặt trên Linux Server của bạn rồi và bạn không cần thiết phải cài đặt chúng lại lần nữa. Tuy nhiên bạn có thể sử dụng các lệnh sau để cài đặt các packages này vào hệ thống của bạn.*

- Lệnh rpm để cài đặt 1 package vào hệ thống là:

```
[root@deep /]# rpm -Uvh name-package-file.rpm
```

- Lệnh sau để kiểm tra xem package có cài đặt vào server hay chưa:
[root@deep /]# rpm -q tên-package-file

Một lần nữa, sau khi cài đặt và biên dịch tất cả các chương trình mà bạn cần trên server của bạn, điều quan trọng mà bạn cần biết là làm thế nào để bạn gỡ bỏ các trình biên dịch được diễn giải ở trên. Điều này sẽ bảo vệ hệ thống của bạn từ những user bất hợp pháp thâm nhập vào hệ thống và biên dịch các chương trình trên server của bạn mà không có sự cho phép.

Một việc khác cần làm là bạn có thể dời chương trình "rpm" ở dạng nhị phân tới một nơi an toàn chẳng hạn như đĩa mềm do một vài nguyên nhân được liệt kê ở trên. Tưởng tượng rằng có vài người bất hợp pháp cố gắng biên dịch các chương trình trên server của bạn và nhận thấy rằng những trình biên dịch đã không còn có sẵn. Chúng sẽ chuyển sang đưa vào server của bạn những chương trình RPM và cài đặt các chương trình này với những lệnh "rpm". Thật ngạc nhiên! Những lệnh "rpm" cũng không sẵn có. Tất nhiên, trong tương lai nếu bạn cần cài đặt phần mềm mới trên server, thì tất cả cái bạn phải làm là đặt các chương trình rpm vào vị trí cũ từ chiếc đĩa mềm ở trên.

- Để chuyển tập tin nhị phân "rpm" vào đĩa mềm, bạn dùng lệnh:

```
[root@deep /]# mount /dev/fd0 /mnt/floppy/
[root@deep /]# mv /bin/rpm /mnt/floppy
```

```
[root@deep /]# umount /mnt/floppy/
```

- Đặt chương trình RPM ở dạng nhị phân vào thư mục nguyên thủy của nó với lệnh sau:

```
[root@deep /]# mount /dev/fd0 /mnt/floppy/
[root@deep /]# cp /mnt/floppy/rpm /bin/
[root@deep /]# umount /mnt/floppy/
```

Ghi chú: *Không nên gỡ bỏ hoàn toàn chương trình RPM khỏi hệ thống của bạn hoặc bạn sẽ không thể cài đặt lại nó lần nữa vì bạn còn cần những lệnh RPM để cài đặt phần mềm khác trên server của bạn.*

Tại sao chúng ta chọn sử dụng tarballs?

Red Hat Linux được phân bổ dưới dạng các tập tin RPM. Một tập tin RPM, cũng hiểu như là một "package", là một cách để phân phối phần mềm sao cho nó có thể dễ dàng được cài đặt, nâng cấp, truy vấn, và xóa bỏ. Tuy nhiên, trong thế giới Unix, chuẩn defacto phân phối các package dưới dạng gọi là "tarballs". Tarballs đơn giản là những tập tin nén có thể đọc và giải nén được với trình tiện ích "tar". Tiến trình cài đặt từ tiện ích tar thông thường không tiện lợi hơn sử dụng RPM. Như vậy vì sao chúng ta chọn để dùng?

1. Không may thay, phải có ít nhất một vài tuần để các nhà phát triển phần mềm làm ra phiên bản mới nhất của gói phần mềm ở dạng RPM, vì nhiều nhà phát triển phần mềm phân phối phiên bản đầu tiên của các package là dạng tarballs.

2. Khi các nhà phát triển và các nhà cung cấp đưa ra các phiên bản mới RPM, họ thường kèm theo nhiều tùy chọn không cần thiết. Nhiều tổ chức hay các công ty đó không biết những tùy chọn nào bạn sẽ cần và tùy chọn nào bạn sẽ không cần. Như vậy họ sẽ kèm theo tất cả các tùy chọn để phù hợp với nhu cầu của mọi người.

3. Thông thường dạng tập tin RPMs chưa được tối ưu hóa cho những bộ xử lý cụ thể của bạn; Các công ty phát triển Hệ điều Hành Linux như Red Hat xây dựng cấu trúc RPMs dựa trên chuẩn PC, điều này cho phép các packages dạng RPM được cài đặt trên tất cả các loại máy tính i386.

4. Thỉnh thoảng bạn tải xuống và cài đặt RPM, việc mà những người khác trên thế giới đang xây dựng và làm sẵn cho những mục đích của bạn. Việc này có thể xảy ra những xung đột trong những trường hợp nào đó phụ thuộc những package riêng lẻ này được xây dựng như thế nào chẳng hạn như những lỗi, sự an toàn và những tất cả những vấn đề khác được diễn giải ở trên.

Biên dịch các phần mềm trên hệ thống của bạn

Một chương trình là cái gì đó mà máy tính có thể thực thi được. Nguyên thủy, các nhà lập trình viết "mã nguồn" trong một ngôn ngữ lập trình mà anh ta hay cô ấy có thể hiểu (ví dụ: C, C ++). Chương trình "Mã nguồn" cũng có ý nghĩa cần tới một trình biên dịch để chuyển đổi "mã nguồn" đó sang tập tin dạng nhị

phân thích hợp để bộ xử lý có thể hiểu được (ví dụ: bộ vi xử lý i386 hoặc tương tự). Một dạng tập tin hiện đại cho những chương trình "có thể thực thi" là elf. Người lập trình đưa mã nguồn cho trình biên dịch và lấy kết quả từ trình biên dịch đó. Điều bạn thường gặp phải khi biên dịch là biên dịch bị lỗi, hoặc đã biên dịch rồi, nhưng chương trình không hoạt động như mong đợi. Một nữa các chương trình phải kiểm tra theo dõi lại và phải sửa lỗi.

Đối với những người mới bắt đầu biên dịch trên hệ thống Linux, có nhiều khía cạnh và những từ mới liên quan đến sự biên dịch mã nguồn mà bạn phải biết, những điều đó bao gồm:

Multiple Files

Thông thường có một số files (gọi là *.c, *.cpp, ...) những files này được biên dịch vào trong những files objects (*.o) và sau đó liên kết vào trong một tập tin có thể thực thi được. Trình biên dịch thông thường thực hiện liên kết và gọi chương trình 'ld' phía sau.

Makefiles

Makefiles giúp đỡ bạn trong việc xây dựng chương trình. Chúng cũng giúp đỡ xây dựng chương trình nhanh chóng. Chương trình "make" sử dụng "quan hệ phụ thuộc" trong Makefile để quyết định những phần của chương trình cần biên dịch lại.

Libraries

Những chương trình có thể được liên kết không chỉ là các files object(*.o) mà còn tới các thư viện hay tập hợp của những tập tin objects đó. Có hai dạng của sự liên kết tới các thư viện : static, dạng này mã nguồn sẽ biên dịch vào tập tin thực thi, và dynamic, dạng này mã nguồn được lựa chọn khi chương trình bắt đầu chạy.

Patches

Ngày nay, các nhà phát triển phần mềm khi thay đổi một tỉ lệ nhỏ của toàn bộ mã nguồn, họ thường đặt những thay đổi này vào một tập tin gọi là "patch". Những phiên bản khác của sản phẩm, những thay đổi mã nguồn, các phiên bản sửa lỗi đều sử dụng "patch" files này.

Errors in Compilation and Linking

Những lỗi trong biên dịch và liên kết thường là do lỗi gõ bàn phím, do thiếu sót, và sự dùng sai ngôn ngữ. Bạn cần kiểm tra chính xác những tập tin được sử dụng cho những hàm mà bạn đang gọi. Những ký hiệu Unreferenced là dấu hiệu của một bước liên kết không hoàn thành. Bạn cũng cần kiểm tra xem những thư viện cần thiết (GLIBC) hoặc những công cụ (GCC, DEV86, AUTOMAKE,...) đã được cài đặt trên hệ thống của bạn chưa.

Debugging

Debugging là một chủ đề lớn. Nó là những khai báo trên các dòng code trong chương trình nhằm báo cho bạn biết những gì đang xảy ra. Để tránh bị sa lầy trong debug, đôi khi bạn có thể in kết quả ra, kiểm tra giá trị các biến có

chính xác giữa các module trong chương trình, làm quen với các công cụ debugging của bạn.

Xây dựng và cài đặt phần mềm trên hệ thống của bạn

Bạn sẽ thấy từ chương 10 đến chương 20, trong các chương này chúng ta sử dụng nhiều lệnh biên dịch khác nhau để xây dựng và cài đặt các chương trình trên server. Những lệnh này đều tương thích với hệ thống UNIX và được sử dụng trên tất cả các máy unix để biên dịch và cài đặt phần mềm.

Những thủ tục để biên dịch và cài đặt phần mềm tarballs trên trên server của bạn:

1. Trước hết, bạn phải tải xuống những tập tin dạng tarball từ những nơi lưu trữ phần mềm mà bạn có thể tin cẩn được. Thông thường được tải xuống từ nơi có phần mềm bạn muốn cài đặt.
2. Sau khi tải xuống tập tin tarball, bạn chuyển thư mục hiện hành sang thư mục "/var/tmp/" (chú ý rằng các thư mục khác đều được, tùy theo ý của bạn) và giải nén tập tin đó bằng lệnh như ví dụ sau (bạn cần có quyền là "root" khi thực hiện lệnh này):

```
[root@deep /]# tar xzpf foo.tar.gz
```

Lệnh trên sẽ giải nén tất cả các tập tin được nén trong tập tin "foo.tar.gz" và tạo một thư mục mới có tên là tên của phần mềm mà bạn muốn cài đặt.

- Tham số "x" Để báo tar rút (extract) tất cả các files có trong tập tin đang thực hiện.
- Tham số "z" để báo tar biết rằng tập tin được nén với tiện ích gzip.
- Tham số "p" Để bảo đảm cấu trúc các tập tin nguyên thủy và các quyền trên tập tin như được nén trong tập tin gốc.
- Tham số "f" Để chỉ tham số kế tiếp là tập tin name.

Khi tập tin dạng tarball đã được giải nén vào trong thư mục thích hợp, bạn sẽ gần như chắc chắn tìm thấy một tập tin "README" và/hoặc một tập tin "INSTALL" ở trong thư mục đó, bạn nên đọc những chỉ dẫn trong tập tin này để biết cách cài đặt gói phần mềm này trên server của bạn. Thông thường, bạn sẽ cần gõ vào những lệnh giống như các lệnh được liệt kê sau:

```
./configure
make
make install
```

Lệnh "./configure" trên sẽ cấu hình software cần cài đặt để bảo đảm rằng hệ thống của bạn có những chức năng và các thư viện cần thiết để quá trình biên dịch thành công. Lệnh "make" sẽ biên dịch tất cả các tập tin mã nguồn vào trong các tập tin dạng nhị phân có thể thi hành được. Cuối cùng, lệnh "make install" sẽ cài đặt các tập tin nhị phân trên và các tập tin phụ kèm theo vào trong những thư mục thích hợp. Những lệnh đặc biệt khác mà bạn sẽ nhìn thấy trong sách này để biên dịch và cài đặt là như sau:

```
make depend
strip
chown
```

Lệnh "make depend" sẽ xây dựng và làm những sự phụ thuộc cần thiết cho những tập tin khác nhau. Lệnh "strip" sẽ loại bỏ tất cả các biểu tượng từ những tập tin đối tượng (object). Điều này có nghĩa rằng các tập tin nhị phân của chúng ta sẽ có kích thước nhỏ hơn. Việc này sẽ cải thiện được tốc độ của hệ thống khi thực thi chương trình vì chỉ có một vài dòng được đọc từ hệ thống khi thực thi. Lệnh "chown" sẽ thiết đặt các tập tin nhị phân với quyền owner (làm chủ file) và quyền nhóm (group) cho các tập tin nhị phân.

Ghi chú: *Nhiều lệnh sẽ được giải thích trong những phần cài đặt liên quan.*

Soạn thảo các tập tin với trình soạn thảo vi

Chương trình vi là trình biên soạn tập tin dạng văn bản mà bạn có thể sử dụng để soạn thảo bất kỳ văn bản nào và các tập tin chương trình. Trong suốt quá trình cài đặt phần mềm, người dùng sẽ thường phải soạn thảo các tập tin văn bản như "Makefiles" hay các tập tin cấu hình để tạo và thay đổi chúng cho phù hợp với ý muốn. Sau đây là vài lệnh phím nóng quan trọng nhất của lệnh vi:

Lệnh	Giải thích
i	Chèn ký tự trước dấu con trỏ.
a	Chèn ký tự sau dấu cursor.
dd	Xóa dòng hiện hành.
x	Xóa ký tự ngay tại dấu cursor.
Esc	Kết thúc chế độ chèn ký tự.
u	Phục hồi sự thay đổi trước đó.
Ctrl + f	Cuộn lên một trang.
Ctrl + b	Cuộn xuống một trang.
/string	Tìm kiếm chuỗi về phía sau dấu cursor.
:f	Hiển thị tên tập tin và số thứ tự của dòng hiện tại.
:q	Thoát lệnh vi.
:q!	Thoát lệnh vi và không lưu lại các thay đổi đã biên soạn.
:wq	Lưu lại và thoát khỏi lệnh vi.

Một vài ghi chú sau cùng

Trước khi theo đọc các phần còn lại của quyển sách này, bạn nên chú ý đến những tập tin nào đó bạn đang quan tâm được đặt trong những thư mục nào đó. Ở

đâu chúng đã được chỉ rõ, những quy ước mà chúng ta chấp nhận ở đây xác định những tập tin này được phân phối bởi Red Hat Linux. Nếu bạn đang sử dụng một sự phân bố của Linux (hoặc của hệ điều hành khác nào đó), có thể chọn phân phối những tập tin này theo một cách khác, bạn cần phải cẩn thận khi sao lại những ví dụ trực tiếp từ văn bản.

Tại phần này của quyển sách, tất cả các phần mềm được liệt kê từ chương 10 đến chương 21 là tùy chọn và phụ thuộc vào những gì bạn muốn cài đặt hoặc làm trên server của bạn (ví dụ: Bạn muốn server bạn làm các tác vụ gì, và mạng của bạn là Intranet hay Internet?). Trong các phần khác, có thể quan trọng cho bạn, sẽ trình bày việc thay thế chương trình Telnet với hỗ trợ SSH để bảo mật việc quản trị hệ thống từ xa một cách an toàn. Một chương trình thú vị khác là Tripwire để giúp cho những người quản trị hệ thống và người dùng trong việc theo dõi các files khi có bất kỳ những sự thay đổi nào trên file.

Chương 10:

PHẦN MỀM BẢO MẬT
(Công cụ giám sát)

Trong chương này chúng ta sẽ đề cập đến các phần mềm sau:

- Linux sXid.

  Cấu hình.

  Các công cụ quản trị sXid.

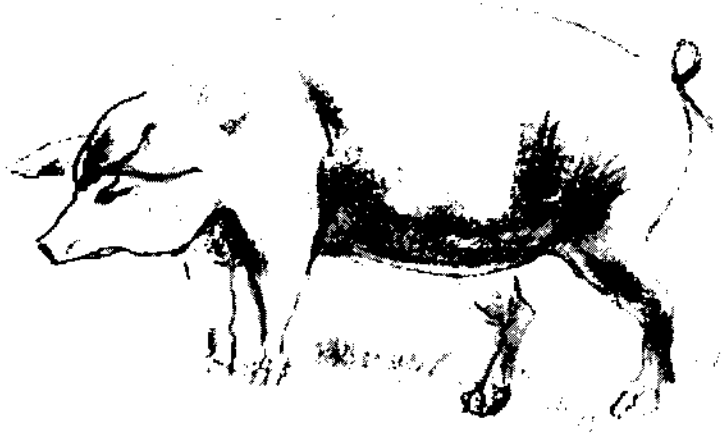
- Linux Logcheck.

  Cấu hình.

- Linux Portsentry.

  Cấu hình.

  Khởi động PortSentry.



Linux sXid

Giới thiệu tổng quan

Các tập tin SUID/SGID có thể trở thành một mối nguy cho vấn đề bảo mật và an toàn hệ thống. Để giảm các rủi ro này, trước đây chúng ta đã remove các bit 's' từ các chương trình được sở hữu bởi root mà sẽ không yêu cầu nhiều quyền sử dụng, nhưng tương lai các tập tin tồn tại khác có thể được đặt với 's' bits được bật lên không có sự thông báo của bạn. sXid là một trong những chương trình theo dõi hệ thống suid/sgid được thiết kế để chạy từ cron trên một nguyên lý cơ bản.

Cơ bản là nó theo dõi bất kỳ sự thay đổi nào trong các thư mục và các tập tin s[ug]id của bạn. Nếu có bất kỳ một điều gì mới trong các thư mục hay tập tin, các thư mục và tập tin này thay đổi các bit hoặc các mode khác sau đó nó sẽ báo về các thay đổi thông qua email hoặc trên dòng lệnh. sXid sẽ tự động thực hiện việc tìm kiếm tất cả SUID/SGID trên máy server của bạn và thông báo về chúng cho bạn. Một khi cài đặt xong bạn có thể quên nó và nó sẽ làm công việc cho bạn.

Tóm tắt các chỉ thị cài đặt

Các dòng lệnh là tương thích với Unix.

Đường dẫn nguồn là "/var/tmp" (các đường dẫn khác thì cũng có thể).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Số phiên bản sXid là 4.0.1

Các gói cần thiết (Packages)

sXid FTP Site: **ftp://marcus.seva.net/pub/sxid/**

Bạn phải bảo đảm là tải về gói: sxid_4.0.1.tar.gz

Dạng lưu trữ Tarballs

Đó là một ý tưởng hay trong việc tạo một danh sách các tập tin trên hệ thống trước khi bạn cài đặt sXid, và một danh sách các tập tin sau khi cài đặt sXid. Sau đó so sánh hai tập tin này dùng lệnh 'diff' để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh "find /* > sXid1" trước và "find /* > sXid2" sau khi bạn cài đặt phần mềm, và sử dụng lệnh "diff sXid1 sXid2 > sXid-Installed" để lấy một danh sách các thay đổi nào đó.

Sự biên dịch

Giải nén dạng lưu trữ tarball (tar.gz)

```
[root@deep /]# cp sxid_version.tar.gz /var/tmp/
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf sxid_version.tar.gz
```

Biên dịch và tối ưu

Chuyển tới các thư mục sXid mới và gõ vào các lệnh sau trên máy trạm (terminal) của bạn:

```
[root@deep tmp]# cd sxid-4.0.1
[root@deep sxid-4.0.1]# make install
```

Các lệnh bên trên sẽ cấu hình phần mềm để bảo đảm rằng hệ thống của bạn có chức năng cần thiết và các thư viện cần thiết để biên dịch tốt các gói (package), biên dịch tất cả các tập tin nguồn sang các tập tin dạng nhị phân có thể thi hành được, và sau đó cài đặt các tập tin nhị phân và các tập tin hỗ trợ bất kỳ vào thư mục các thích hợp.

Xoá sạch sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf sxid-version/ sxid_version.tar.gz
```

Lệnh "rm" được sử dụng ở trên sẽ xóa tất cả các tập tin nguồn mà chúng ta dùng để biên dịch và cài đặt sXid. Nó cũng sẽ xóa các dạng lưu trữ được nén từ thư mục "/var/tmp".

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

Để chạy sXid, tập tin sau được yêu cầu và phải được tạo hoặc được sao chép tới thư mục thích hợp trên máy server của bạn.

Sao chép tập tin **sxid.conf** tới thư mục "/etc/".

Cấu hình tập tin "/etc/sxid.conf"

Tập tin cấu hình cho sXid ("/etc/sxid.conf") cho phép bạn đặt các tùy chọn mà hiệu chỉnh các thao tác của chương trình. Điều đó được chú thích kỹ lưỡng và rất cơ bản.

Bước 1

Soạn thảo tập tin **sxid.conf** (vi /etc/sxid.conf) và đặt lại theo các yêu cầu của bạn:

```
# Configuration file for sXid
# Note that all directories must be absolute with no trailing /'s

# Where to begin our taäp tin search
SEARCH = "/"

# Which subdirectories to exclude from searching
EXCLUDE = "/proc /mnt /cdrom /floppy"
```

```
# Who to send reports to
EMAIL = "root"

# Always send reports, even when there are no changes?
ALWAYS_NOTIFY = "no"

# Where to keep interim logs. This will rotate 'x' number of
# times based on KEEP_LOGS below
LOG_FILE = "/var/log/sxid.log"

# How many logs to keep
KEEP_LOGS = "5"

# Rotate the logs even when there are no changes?
ALWAYS_ROTATE = "no"

# Directories where +s is forbidden (these are searched
# even if not explicitly in SEARCH). EXCLUDE rules apply
FORBIDDEN = "/home /tmp"

# Remove (-s) files found in forbidden directories?
ENFORCE = "yes"

# This implies ALWAYS_NOTIFY. It will send a full list of
# entries along with the changes
LISTALL = "no"

# Ignore entries for directories in these paths
# (this means that only files will be recorded, you
# can effectively ignore all directory entries by
# setting this to "/"). The default is /home since
# some systems have /home g+s.
IGNORE_DIRS = "/home"

# file that contains a list of (each on it's own line)
# of other files that sxid should monitor. This is useful
# for files that aren't +s, but relate to system
# integrity (tcpd, inetd, apache...).
# EXTRA_LIST = "/etc/sxid.list"
```

```
# Mail program. This changes the default compiled in
# mailer for reports. You only need this if you have changed
# it's location and don't want to recompile sxid.
# MAIL_PROG = "/usr/bin/mail"
```

Step 2

Đặt một mục nhập (entry) vào trong tập tin crontab của root để tạo sXid chạy như một cronjob:

Sxid sẽ chạy từ crond; cơ bản nó theo dõi bất kỳ sự thay đổi nào trong các thư mục và các tập tin s[ug]id của bạn. Nếu có bất kỳ một điều gì mới trong các thư mục hay tập tin, các thư mục và tập tin này thay đổi các bit hoặc các mode khác và nó sẽ báo cáo về các thay đổi. Để thêm sxid vào trong cronjob của bạn, phải hiệu chỉnh tập tin crontab (nếu không có sẵn tập tin này trên server của bạn) và thêm vào các dòng sau:

- Để soạn thảo crontab, dùng lệnh sau (có quyền là root):

```
[root@deep /]# crontab -e
# Sample crontab entry to run every day at 4am
0 4 * * * /usr/bin/sxid
```

Tài liệu kỹ thuật chi tiết

Để biết thêm chi tiết, có một số trang hướng dẫn (man) bạn có thể đọc :

```
$ man sxid.conf (5) - configuration settings for sxid
$ man sxid (1) - check for changes in s[ug]id files and directories
```

Các công cụ quản trị sXid

Ý nghĩa của chương trình này là chạy như một cronjob. Nó phải được chạy một lần một ngày, nhưng đối với busy shell boxes có thể cần chạy nó hai lần một ngày. Bạn cũng có thể chạy chương trình này bằng tay cho việc kiểm tra dấu vết (spot-checking).

- Để Sxid được thực hiện bằng tay, sử dụng lệnh:

```
[root@deep /]# sxid -k
sXid Vers : 4.0.1
Check run : Wed Dec 29 12:40:32 1999
This host : mail.openna.com
Spotcheck : /home/admin
Excluding : /proc /mnt /cdrom /floppy
Ignore Dirs: /home
Forbidden : /home /tmp
```

```
No changes found
```

Điều này kiểm tra các thay đổi bằng cách đệ qui trong thư mục làm việc hiện hành. Các tập tin xuất các kết quả theo dõi (log file) sẽ không được đề cập đến và cũng không được email gửi đi. Tất cả các kết quả xuất ra thẳng stdout.

Các tập tin cài đặt

> /etc/sxid.conf

> /usr/bin/sxid

> /usr/man/man1/sxid.1

> /usr/man/man5/sxid.conf.5

Linux Logcheck

Giới thiệu

Một công việc quan trọng trong thế giới bảo mật và an toàn là phải kiểm tra thường xuyên các tập tin xuất ra các kết quả theo dõi hệ thống (log files). Thông thường các hoạt động hàng ngày của một người quản trị hệ thống không cho phép anh ta có thời gian để thực hiện công việc này và có thể mang đến nhiều vấn đề.

Giải thích tính trừu tượng của Logcheck:

Kiểm tra theo dõi và ghi nhận các sự kiện xảy ra trong hệ thống thì rất quan trọng! Quan trọng hơn ở đây là gì? Những người quản trị hệ thống nhận biết được các sự kiện này do vậy họ có thể ngăn chặn các vấn đề sẽ chắc chắn xảy ra nếu bạn có một hệ thống được kết nối tới Internet. Thật không may cho hầu hết logfile là nó không giải quyết được vấn đề bạn phải thực hiện công việc log như thế nào khi không có ai kiểm tra các log đó, mà nó chỉ thường được kiểm tra khi có sự kiện nào đó xảy ra. Điều này logcheck sẽ giúp đỡ cho bạn. Logcheck tự động kiểm tra tiến trình, loại bỏ thông tin log "bình thường", cung cấp cho bạn một cái nhìn được cô đọng lại về các vấn đề và những tác nhân tiềm ẩn gây ra rắc rối, và sẽ được gửi đến bất cứ nơi đâu mà bạn muốn. Logcheck là một phần mềm đóng gói, nó được thiết kế để chạy một cách tự động và kiểm tra các tập tin log của hệ thống cho các phá rối về tính an toàn bảo mật và các công việc không bình thường được thực hiện trên server của bạn. Tiện ích Logcheck là một chương trình được gọi là theo dõi log (logtail) mà nên nhớ rằng vị trí sau cùng nó đọc từ trong một tập tin log và nó sử dụng vị trí này để tiếp tục chạy để xử lý các thông tin mới.

Tóm tắt các chỉ thị cài đặt này

Các dòng lệnh tương thích với Unix.

Đường dẫn nguồn là "/var/tmp" (các đường dẫn khác là cũng có thể).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Số phiên bản Logcheck là 1.1.1

Các gói cần thiết (Packages)

Logcheck Homepage Site: http://www.psionic.com/abacus/logcheck/

Bạn phải bảo đảm là tải về gói : logcheck-1.1.1.tar.gz

Dạng lưu trữ Tarballs

Đó là một ý tưởng hay trong việc tạo một danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt Logcheck, và sau đó so sánh chúng dùng lệnh "diff" để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh "find /* > Logcheck1" trước và "find /* > Logcheck2" sau khi bạn cài đặt phần mềm, và sử dụng lệnh "'diff Logcheck1 Logcheck2 > Logcheck-Installed" để lấy một danh sách các thay đổi nào đó.

Sự biên dịch

Giải nén dạng lưu trữ tarball (tar.gz)

```
[root@deep /]# cp logcheck-version.tar.gz /var/tmp/
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf logcheck-version.tar.gz
```

Biên dịch và tối ưu

Bạn phải hiệu chỉnh tập tin "Makefile" của Logcheck để chỉ rõ các đường dẫn cài đặt, các cờ hiệu biên dịch, và các vấn đề tối ưu hóa cho hệ thống. Chúng ta phải hiệu chỉnh tập tin này cho tương thích với cấu trúc hệ thống tập tin của Red Hat và các tập tin script Logcheck trong biến môi trường PATH của chúng ta.

Bước 1

Chuyển tới các thư mục Logcheck mới và gõ vào các lệnh sau trên máy trạm (terminal) của bạn:

Soạn thảo tập tin **Makefle** (vi Makefile) và thay đổi các dòng sau:

CC = cc

Trở thành:

CC = egcs

CFLAGS = -O

Trở thành:

CFLAGS = -O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -march=pentiumpro -fomit-frame-pointer -fno-exceptions

INSTALLDIR = /usr/local/etc

Trở thành:

INSTALLDIR = /etc/logcheck

INSTALLDIR_BIN = /usr/local/bin

Trở thành:

INSTALLDIR_BIN = /usr/bin

INSTALLDIR_SH = /usr/local/etc

Trở thành:

INSTALLDIR_SH = /usr/bin

TMPDIR = /usr/local/etc/tmp

Trở thành:

TMPDIR = /etc/logcheck/tmp

Các thay đổi phía trên sẽ cấu hình phần mềm để sử dụng trình biên dịch "egcs", các cờ hiệu tối ưu hoá được chỉ rõ ra trong hệ thống của bạn, các định vị tất cả các tập tin liên quan tới phần mềm Logcheck tới các thư mục đích sau cùng mà chúng ta chọn để tương thích với cấu trúc hệ thống tập tin Red Hat.

Bước 2

Soạn thảo tập tin Makefile (vi +67 Makefile) và thay đổi dòng sau:

@if [! -d $(TMPDIR)]; then /bin/mkdir $(TMPDIR); fi

Trở thành:

@if [! -d $(TMPDIR)]; then /bin/mkdir -p $(TMPDIR); fi

Thay đổi bên trên (-p) sẽ cho phép chương trình cài đặt để tạo các thư mục cha khi cần thiết.

Bước 3

Cài đặt Logcheck trên hệ thống.

[root@deep logcheck-1.1.1]# make linux

Lệnh ở trên sẽ cấu hình phần mềm cho hệ điều hành Linux, biên dịch tất cả các tập tin nguồn sang dạng nhị phân có thể thi hành được, và sau đó cài đặt các tập tin nhị phân này và các tập tin hỗ trợ vào trong các vị trí thích hợp.

Xoá sạch sau khi làm việc

[root@deep /]# cd /var/tmp

[root@deep tmp]# rm -rf logcheck-version/ logcheck-version.tar.gz

Lệnh "rm" như được sử dụng bên trên sẽ xóa tất cả các tập tin nguồn mà chúng ta dùng để biên dịch và cài đặt Logcheck. Nó cũng sẽ xóa các dạng lưu trữ được nén từ thư mục "/var/tmp".

Cấu hình

Cấu hình tập tin script "/usr/bin/logcheck.sh"

Bởi vì chúng ta đang sử dụng một đường dẫn xen kẽ cho các tập tin (nghĩa là không phải /usr/local/etc), chúng ta cần thay đổi đường dẫn tới các điểm vào cho "logcheck.hacking", "logcheck.ignore", "logcheck.violations.ignore", và "logtail" trong tập tin script chính logcheck.sh. Tập tin script của Logcheck "/usr/bin/logcheck.sh" cho phép bạn cài đặt các tùy chọn này nhằm hiệu chỉnh toàn bộ đường dẫn và sự hoạt động của chương trình. Điều này sẽ được ghi chú kỹ và rất cơ bản.

Bước 1

Soạn thảo tập tin **logcheck.sh** (vi /usr/bin/logcheck.sh) thay đổi như sau:

LOGTAIL=/usr/local/bin/logtail

Trở thành:

LOGTAIL=/usr/bin/logtail

TMPDIR=/usr/local/etc/tmp

Trở thành:

TMPDIR=/etc/logcheck/tmp

HACKING_FILE=/usr/local/etc/logcheck.hacking

Trở thành:

HACKING_FILE=/etc/logcheck/logcheck.hacking

VIOLATIONS_FILE=/usr/local/etc/logcheck.violations

Trở thành:
VIOLATIONS_FILE=/etc/logcheck/logcheck.violations

VIOLATIONS_IGNORE_FILE=/usr/local/etc/logcheck.violations.ignore

Trở thành:
VIOLATIONS_IGNORE_FILE=/etc/logcheck/logcheck.violations.ignore

IGNORE_FILE=/usr/local/etc/logcheck.ignore

Trở thành:
IGNORE_FILE=/etc/logcheck/logcheck.ignore

Bước 2

Đặt một mục nhập (entry) vào trong tập tin crontab của root để tạo Logcheck chạy như một cronjob:

Sau khi cài đặt Logcheck, bạn nên soạn thảo tập tin cục bộ crontab cho root và cài đặt Logcheck để chạy một lần trên giờ (tất nhiên là bạn có thể thực hiện điều này thường xuyên hơn, hoặc ít thường xuyên). Để thêm Logcheck vào trong cronjob của bạn, bạn phải hiệu chỉnh tập tin crontab và thêm vào dòng sau:

- Để hiệu chỉnh crontab, dùng lệnh sau (như root):

```
[root@deep /]# crontab -e
# Hourly check Log files for security violations and unusual activity.
00 * * * * /usr/bin/logcheck.sh
```

Ghi chú: *Hãy nhớ rằng, Logcheck không báo về bất cứ cái gì thông qua email nếu nó không có vấn đề gì xảy ra.*

Các tập tin cài đặt

> /etc/logcheck
> /etc/logcheck/tmp
> /etc/logcheck/logcheck.hacking
> /etc/logcheck/logcheck.violations
> /etc/logcheck/logcheck.violations.ignore
> /etc/logcheck/logcheck.ignore
> /usr/bin/logcheck.sh
> /usr/bin/logtail
> /var/log/messages.offset
> /var/log/secure.offset
> /var/log/maillog.offset

Linux PortSentry

Tổng quan

Bức tường lửa (firewall) giúp đỡ chúng ta bảo vệ mạng khỏi những xâm nhập bất hợp pháp từ bên ngoài. Với firewall chúng ta có thể chọn những ports nào chúng ta muốn mở và những port nào chúng ta sẽ đóng. Thông tin trên được giữ một cách bí mật bởi những người chịu trách nhiệm đến firewall. Tuyệt đối không người nào từ bên ngoài biết thông tin này, tuy nhiên các hackers (tin tặc) cũng như các spammers biết một vài cách tấn công bạn, họ có thể sử dụng một chương trình đặc biệt để quét tất cả các ports trên server của bạn nhặt thông tin quí giá này (port nào mở và port nào đóng).

Như được giải thích trong lời giới thiệu của phần PortSentry:

Một chương trình quét port là một dấu hiệu của một vấn đề lớn đang đến với bạn. Nó thường là tiền thân cho một sự tấn công và là một bộ phận nguy hiểm trong việc bảo vệ hữu hiệu tài nguyên thông tin của bạn. PortSentry là một chương trình được thiết kế để phát hiện ra và phản hồi tới các port quét nhằm chống lại một host đích trong thời gian chúng đang thực hiện quét port và có một số tùy chọn để phát hiện ra các port quét. Khi nó tìm thấy một port quét nó có thể phản ứng lại trong những cách sau:

- Một logfile lưu các sự việc xảy ra thông qua syslog().
- Tên host mục tiêu tự động được bỏ vào trong tập tin "/etc/hosts.deny" cho những trình bao bọc TCP.
- Host nội bộ tự động cấu hình lại để hướng tất cả các lưu thông tới host mục tiêu trỏ tới một host không hoạt động (deal host) làm hệ thống mục tiêu biến mất.
- Local host tự động cấu hình lại để loại bỏ tất cả các gói thông tin từ host mục tiêu thông qua bộ lọc local host.

Mục đích của Portsentry là để giúp người quản trị mạng có được công cụ khảo sát kỹ lưỡng hệ thống của mình.

Các chỉ dẫn cài đặt Portsentry

- Những lệnh dùng trong phần này đều tương thích với Unix.
- Đường dẫn nguồn là "/var/tmp" (Các đường dẫn khác đều có thể được).
- Quá trình cài đặt đã được kiểm tra trên Red Hat Linux 6.1 and 6.2.
- Tất cả các bước trong quá trình cài đặt đều thực hiện với quyền super-user "root".
- Phiên bản của Portsentry là 1.0

Gói dữ liệu (Package)

Bạn có thể tìm thấy Portsentry từ trang chủ:

http://www.psionic.com/abacus/portsentry/

Bạn phải đảm bảo rằng đã tải về file: portsentry-1.0.tar.gz

Tarballs

Bạn nên liệt kê các files trên hệ thống trước và sau khi cài đặt Portsentry, sau đó bạn so sánh chúng bằng cách dùng lệnh "diff" để tìm ra nơi lưu trữ các tập tin cài đặt. Việc này đơn giản bằng cách dùng lệnh "find /* > Portsentry1" trước và lệnh "find /* > Portsentry2" sau khi cài đặt Portsentry, sau đó dùng lệnh "diff Portsentry1 Portsentry2 > PortSentry-Installed" để có danh sách các thay đổi.

Biên dịch

Giải nén tập tin tarball (tar.gz).

```
[root@deep /]# cp portsentry-version.tar.gz /var/tmp/
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf portsentry-version.tar.gz
```

Biên dịch và tối ưu hóa

Bạn nên sửa tập tin "Makefile" để xác định đường dẫn cài đặt Portsentry, các cờ hiệu biên dịch, và tối ưu hoá hệ thống của bạn. Bạn cũng nên sửa lại tập tin này để tương thích với cấu trúc tập tin hệ thống của Red Hat.

Bước 1

Chuyển vào thư mục Portsentry và gõ vào những lệnh sau trên màn hình terminal của bạn:

Mở tập tin Makefile (dùng lệnh vi Makefile) và thay đổi các dòng sau:

```
#CC = cc
```

Trở thành:

```
CC = egcs
```

```
#CFLAGS = -O –Wall
```

Trở thành:

```
CFLAGS = -O9 -funroll-loops -ffast-math -malign-double - cpu=pentiumpro -
march=pentiumpro - fomit-frame-pointer -fno-exceptions –Wall
#INSTALLDIR = /usr/local/psionic
```

Trở thành:

```
INSTALLDIR = /usr/psionic
```

Những sự thay đổi ở trên sẽ cấu hình phần mềm sử dụng trình biên dịch "egcs", tối ưu hóa các cờ hiệu cụ thể cho hệ thống, và xác định vị trí tất cả các tập tin liên quan đến phần mềm Portsentry đặt vào những thư mục mà chúng ta đã chọn.

Bước 2

Trước đây chúng ta sử dụng một đường dẫn không thích hợp (ví dụ /usr/local/psionic). Chúng ta cần thay đổi đường dẫn tập tin cấu hình PortSentry trong tập tin header chính "portsentry_config.h".

Chuyển vào trong thư mục PortSentry và soạn thảo tập tin portsentry_config.h (vi portsentry_config.h) và thay đổi các dòng sau:

```
#define CONFIG_FILE "/usr/local/psionic/portsentry/portsentry.conf"
```

Trở thành:

```
#define CONFIG_FILE "/usr/psionic/portsentry/portsentry.conf"
```

Bước 3

Cài Đặt Portsentry vào hệ thống

```
[root@deep portsentry-1.0]# make linux
[root@deep portsentry-1.0]# make install
```

Các lệnh trên sẽ cấu hình Portsentry cho hệ điều hành Linux, biên dịch, xây dựng, và cuối cùng cài đặt các tập tin vào trong thư mục thích hợp.

Xóa các tập tin không cần thiết sau khi cài đặt

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf portsentry-version/ portsentry-version.tar.gz
```

Lệnh "rm" sẽ xóa tất cả các tập tin nguồn mà chúng ta đã dùng để biên dịch và cài đặt PortSentry. Nó cũng xóa tập tin nén Portsentry trong thư mục "/var/tmp".

Cấu hình

Cấu hình tập tin "/usr/psionic/portsentry/portsentry.conf"

File "/usr/psionic/portsentry/portsentry.conf " là tập tin cấu hình chính của phần mềm PortSentry; bạn có thể định rõ những port nào bạn muốn mở, những địa chỉ IP nào bị từ chối, theo dõi, bỏ qua, và vô hiệu hóa chức năng phản hồi tự động và một số tính năng khác .... để biết thêm chi tiết hãy đọc tập tin "README.install" trong thư mục nguồn của PortSentry.

Soạn thảo tập tin portsentry.conf (vi /usr/psionic/portsentry.conf), sau đó kiểm tra và thay đổi các chọn lựa mà bạn cần:

```
# PortSentry Configuration
#
# $Id: portsentry.conf,v 1.13 1999/11/09 02:45:42 crowland Exp crowland $
#
# IMPORTANT NOTE: You CAN NOT put spaces between your port arguments.
#
# The default ports will catch a large number of common probes
#
# All entries must be in quotes.
########################
# Port Configurations #
########################
```

```
#
#
# Some example port configs for classic and basic Stealth modes
#
# I like to always keep some ports at the "low" end of the spectrum.
# This will detect a sequential port sweep really quickly and usually
# these ports are not in use (i.e. tcpmux port 1)
#
# ** X-Windows Users **: If you are running X on your box, you need to be sure
# you are not binding PortSentry to port 6000 (or port 2000 for OpenWindows users).
# Doing so will prevent the X-client from starting properly.
#
# These port bindings are *ignored* for Advanced Stealth Scan Detection Mode.
#
# Un-comment these if you are really anal:
#TCP_PORTS="1,7,9,11,15,70,79,80,109,110,111,119,138,139,143,512,513,514,515,5
40,635,1080,1524,2000,2001,4000,4001,5742,6000,6001,6667,12345,12346,20034,303
03,32771,32772,32773,32774,31337,40421,40425,49724,54320"
#UDP_PORTS="1,7,9,66,67,68,69,111,137,138,161,162,474,513,517,518,635,640,641,
666,700,2049,32770,32771,32772,32773,32774,31337,54321"
#
# Use these if you just want to be aware:
TCP_PORTS="1,11,15,79,111,119,143,540,635,1080,1524,2000,5742,6667,12345,123
46,20034,31337,32771,32772,32773,32774,40421,49724,54320"
UDP_PORTS="1,7,9,69,161,162,513,635,640,641,700,32770,32771,32772,32773,3277
4,31337,54321"
#
# Use these for just bare-bones
#TCP_PORTS="1,11,15,110,111,143,540,635,1080,524,2000,12345,12346,20034,3277
1,32772,32773,32774,49724,54320"
#UDP_PORTS="1,7,9,69,161,162,513,640,700,32770,32771,32772,32773,32774,31337
,54321"

##########################################
# Advanced Stealth Scan Detection Options #
##########################################
#
# This is the number of ports you want PortSentry to monitor in Advanced mode.
# Any port *below* this number will be monitored. Right now it watches
# everything below 1023.
#
# On many Linux systems you cannot bind above port 61000. This is because
# these ports are used as part of IP masquerading. I don't recommend you
```

```
# bind over this number of ports. Realistically: I DON'T RECOMMEND YOU MONITOR
# OVER 1023 PORTS AS YOUR FALSE ALARM RATE WILL ALMOST CERTAINLY RISE. You've been
# warned! Don't write me if you have have a problem because I'll only tell
# you to RTFM and don't run above the first 1023 ports.
#
#
ADVANCED_PORTS_TCP="1023"
ADVANCED_PORTS_UDP="1023"
#
# This field tells PortSentry what ports (besides listening daemons) to
# ignore. This is helpful for services like ident that services such
# as FTP, SMTP, and wrappers look for but you may not run (and probably
# *shouldn't* IMHO).
#
# By specifying ports here PortSentry will simply not respond to
# incoming requests, in effect PortSentry treats them as if they are
# actual bound daemons. The default ports are ones reported as
# problematic false alarms and should probably be left alone for
# all but the most isolated systems/networks.
#
# Default TCP ident and NetBIOS service
ADVANCED_EXCLUDE_TCP="113,139"
# Default UDP route (RIP), NetBIOS, bootp broadcasts.
ADVANCED_EXCLUDE_UDP="520,138,137,67"
######################
# Configuration Files#
######################
#
# Hosts to ignore
IGNORE_FILE="/usr/psionic/portsentry/portsentry.ignore"
# Hosts that have been denied (running history)
HISTORY_FILE="/usr/psionic/portsentry/portsentry.history"
# Hosts that have been denied this session only (temporary until next restart)
BLOCKED_FILE="/usr/psionic/portsentry/portsentry.blocked"
###################
# Response Options#
###################
# Options to dispose of attacker. Each is an action that will
# be run if an attack is detected. If you don't want a particular
# option then comment it out and it will be skipped.
#
```

```
# The variable $TARGET$ will be substituted with the target attacking
# host when an attack is detected. The variable $PORT$ will be substituted
# with the port that was scanned.
#
##################
# Ignore Options #
##################

# These options allow you to enable automatic response
# options for UDP/TCP. This is useful if you just want
# warnings for connections, but don't want to react for
# a particular protocol (i.e. you want to block TCP, but
# not UDP). To prevent a possible Denial of service attack
# against UDP and stealth scan detection for TCP, you may
# want to disable blocking, but leave the warning enabled.
# I personally would wait for this to become a problem before
# doing though as most attackers really aren't doing this.
# The third option allows you to run just the external command
# in case of a scan to have a pager script or such execute
# but not drop the route. This may be useful for some admins
# who want to block TCP, but only want pager/e-mail warnings
# on UDP, etc.
#
#
# 0 = Do not block UDP/TCP scans.
# 1 = Block UDP/TCP scans.
# 2 = Run external command only (KILL_RUN_CMD)
BLOCK_UDP="1"
BLOCK_TCP="1"
####################
# Dropping Routes:#
####################
# This command is used to drop the route or add the host into
# a local filter table.
#
# The gateway (333.444.555.666) should ideally be a dead host on
# the *local* subnet. On some hosts you can also point this at
# localhost (127.0.0.1) and get the same effect. NOTE THAT
# 333.444.555.66 WILL *NOT* WORK. YOU NEED TO CHANGE IT!!
#
# All KILL ROUTE OPTIONS ARE COMMENTED OUT INITIALLY. Make sure you
```

```
# uncomment the correct line for your OS. If you OS is not listed
# here and you have a route drop command that works then please
# mail it to me so I can include it. ONLY ONE KILL_ROUTE OPTION
# CAN BE USED AT A TIME SO DON'T UNCOMMENT MULTIPLE LINES.
#
# NOTE: The route commands are the least optimal way of blocking
# and do not provide complete protection against UDP attacks and
# will still generate alarms for both UDP and stealth scans. I
# always recommend you use a packet filter because they are made
# for this purpose.
#
# Generic
#KILL_ROUTE="/sbin/route add $TARGET$ 333.444.555.666"
# Generic Linux
#KILL_ROUTE="/sbin/route add -host $TARGET$ gw 333.444.555.666"
# Newer versions of Linux support the reject flag now. This
# is cleaner than the above option.
KILL_ROUTE="/sbin/route add -host $TARGET$ reject"
# Generic BSD (BSDI, OpenBSD, NetBSD, FreeBSD)
#KILL_ROUTE="/sbin/route add $TARGET$ 333.444.555.666"
# Generic Sun
#KILL_ROUTE="/usr/sbin/route add $TARGET$ 333.444.555.666 1"
# NEXTSTEP
#KILL_ROUTE="/usr/etc/route add $TARGET$ 127.0.0.1 1"
# FreeBSD (Not well tested.)
#KILL_ROUTE="route add -net $TARGET$ -netmask 255.255.255.255 127.0.0.1 -
blackhole"
# Digital UNIX 4.0D (OSF/1 / Compaq Tru64 UNIX)
#KILL_ROUTE="/sbin/route add -host -blackhole $TARGET$ 127.0.0.1"
# Generic HP-UX
#KILL_ROUTE="/usr/sbin/route add net $TARGET$ netmask 255.255.255.0 127.0.0.1"
##
# Using a packet filter is the preferred method. The below lines
# work well on many OS's. Remember, you can only uncomment *one*
# KILL_ROUTE option.
##
################
# TCP Wrappers#
################
# This text will be dropped into the hosts.deny taäp tin for wrappers
# to use. There are two formats for TCP wrappers:
#
```

```
# Format One: Old Style - The default when extended host processing
# options are not enabled.
#
KILL_HOSTS_DENY="ALL: $TARGET$"
#
# Format Two: New Style - The format used when extended option
# processing is enabled. You can drop in extended processing
# options, but be sure you escape all '%' symbols with a backslash
# to prevent problems writing out (i.e. \%c \%h )
#
#KILL_HOSTS_DENY="ALL: $TARGET$ : DENY"
####################
# External Command#
####################
# This is a command that is run when a host connects, it can be whatever
# you want it to be (pager, etc.). This command is executed before the
# route is dropped. I NEVER RECOMMEND YOU PUT IN RETALIATORY ACTIONS
# AGAINST THE HOST SCANNING YOU. TCP/IP is an *unauthenticated protocol*
# and people can make scans appear out of thin air. The only time it
# is reasonably safe (and I *never* think it is reasonable) to run
# reverse probe scripts is when using the "classic" -tcp mode. This
# mode requires a full connect and is very hard to spoof.
#
#KILL_RUN_CMD="/some/path/here/script $TARGET$ $PORT$"
######################
# Scan trigger value#
######################
# Enter in the number of port connects you will allow before an
# alarm is given. The default is 0 which will react immediately.
# A value of 1 or 2 will reduce false alarms. Anything higher is
# probably not necessary. This value must always be specified, but
# generally can be left at 0.
#
# NOTE: If you are using the advanced detection option you need to
# be careful that you don't make a hair trigger situation. Because
# Advanced mode will react for *any* host connecting to a non-used
# below your specified range, you have the opportunity to really
# break things. (i.e someone innocently tries to connect to you via
# SSL [TCP port 443] and you immediately block them). Some of you
# may even want this though. Just be careful.
#
```

```
SCAN_TRIGGER="0"
#######################
# Port Banner Section#
#######################
#
# Enter text in here you want displayed to a person tripping the PortSentry.
# I *don't* recommend taunting the person as this will aggravate them.
# Leave this commented out to disable the feature
#
# Stealth scan detection modes don't use this feature
#
PORT_BANNER="*** UNAUTHORIZED ACCESS PROHIBITED *** YOUR CONNECTION ATTEMPT HAS BEEN LOGGED. GO AWAY."
# EOF
```

Bây giờ, chúng ta kiểm tra/thay đổi quyền mặc định của tập tin portsentry.conf cho mục đích an toàn:

```
[root@deep /]# chmod 600 /usr/psionic/portsentry/portsentry.conf
```

Cấu hình tập tin "/usr/psionic/portsentry/portsentry.ignore"

Tập tin "/usr/psionic/portsentry/portsentry.ignore" là nơi mà bạn có thể thêm bất kỳ host nào bạn muốn lờ đi nếu nó kết nối tới một tripwired port. Trong tập tin này phải luôn luôn chứa tối thiểu một địa chỉ của localhost (127.0.0.1) và IP của giao tiếp nội bộ (local interface) (lo). Không khuyến cáo rằng bạn đặt mọi địa chỉ IP trong mạng nội bộ của bạn.

Soạn thảo tập tin portsentry.ignore (vi /usr/psionic/portsentry.ignore) và thêm vào địa chỉ IP của bất kỳ host nào bạn muốn lờ đi nếu chúng kết nối tới một tripwired port:

```
# Put hosts in here you never want blocked. This includes the IP addresses
# of all local interfaces on the protected host (i.e virtual host, mult-home)
# Keep 127.0.0.1 and 0.0.0.0 to keep people from playing games.
127.0.0.1
```

Bây giờ, để an toàn cho vấn đề bảo mật bạn cần đổi quyền trên tập tin này:

```
[root@deep /]# chmod 600 /usr/psionic/portsentry/portsentry.ignore
```

Khởi động PortSentry

Chương trình PortSentry có thể được cấu hình trong sáu kiểu hoạt động khác nhau, nhưng nhận thấy rằng chỉ một kiểu giao thức có thể được khởi động tại một thời điểm. Chính xác hơn, bạn có thể khởi động một kiểu TCP và một kiểu UDP, như vậy có hai kiểu: kiểu TCP và UDP. Những kiểu sẵn có là:

- portsentry -tcp (basic port-bound TCP mode)
- portsentry -udp (basic port-bound UDP mode)

- portsentry -stcp (Stealth TCP scan detection)
- portsentry -atcp (Advanced TCP stealth scan detection)
- portsentry -sudp ("Stealth" UDP scan detection)
- portsentry -audp (Advanced "Stealth" UDP scan detection)

Trong phần dưới đây, chúng ta sẽ khởi động kiểu TCP theo cơ chế "Advanced TCP stealth scan detection" và kiểu UDP theo cơ chế "Stealth UDP scan detection". Để có thông tin thêm về những kiểu giao thức khác, mời bạn tham khảo thêm trong tập tin "README.install" và "README.stealth" trong thư mục nguồn của PortSentry.

Khởi động kiểu TCP theo cơ chế -atcp:

-atcp - Advanced TCP stealth scan detection mode

Với kiểu Advanced TCP stealth scan detection (kiểu giao thức "-atcp"). Đầu tiên PortSentry sẽ kiểm tra xem các ports nào bạn đang chạy trên server, sau đó sẽ loại bỏ việc giám sát những port này và sẽ bắt đầu xem những port còn lại. Điều này rất hữu ích và có tác động chống lại một cách cực kỳ nhanh đối với các chương trình quét ports. Cơ chế này sử dụng rất ít thời gian của CPU.

Khởi động kiểu UDP theo cơ chế -sudp:

-sudp - "Stealth" UDP scan detection mode

Với kiểu giao thức Stealth UDP scan detection mode "-sudp", Các ports UDP sẽ được liệt kê và sau đó được giám sát.

- Để khởi động PortSentry theo hai mode trên, bạn dùng lệnh sau:

```
[root@deep /]# /usr/psionic/portsentry/portsentry –atcp
[root@deep /]# /usr/psionic/portsentry/portsentry –sudp
```

***Ghi chú** : Bạn có thể thêm các dòng lệnh ở trên vào trong tập tin script"/etc/rc.d/rc.local" và phần mềm PortSentry sẽ tự động được khởi động mỗi khi bạn khởi động hệ thống.*

Các files của PortSentry được cài đặt trong các thư mục sau:

- /usr/psionic
- /usr/psionic/portsentry
- /usr/psionic/portsentry/portsentry.conf
- /usr/psionic/portsentry/portsentry.ignore
- /usr/psionic/portsentry/portsentry

Chương 11:

PHẦN MỀM BẢO MẬT
(Các dịch vụ mạng)

Trong chương này chúng ta sẽ đề cập đến các phần mềm sau:

- Linux OpenSSH Client/Server.

  Cấu hình.

  Cấu hình OpenSSH để dùng TCP-Wrappers inetd super server.

  Cấu hình OpenSSH cho mỗi người sử dụng.

  Các công cụ OpenSSH cho người sử dụng.

- Linux SSH2 Client/Server.

  Cấu hình.

  Cấu hình sshd2 để dùng TCP-Wrappers inetd super server.

  Cấu hình ssh2 cho mỗi người sử dụng.

  Các công cụ OpenSSH cho người sử dụng.

Linux OpenSSH Client/Server

Giới thiệu tổng quan

Như được minh hoạ trong chương 2, "Sự cài đặt Linux server", rất nhiều dịch vụ mạng được đưa vào, nhưng không có hạn chế các dịch vụ như rsh, rlogin, hoặc rexec không bị xâm nhập với kiểu mà các tin tặc thường dùng như nghe trộm điện tử. Như một hệ quả, bất kỳ ai, người mà đã truy cập tới bất kỳ máy tính nào đã được kết nối vào mạng đều có thể lắng nghe trên đường truyền giao tiếp của họ và lấy về mật khẩu của bạn, và cũng giống như việc lấy bất kỳ thông tin riêng tư nào khác thông qua đường mạng ở dạng văn bản. Hiện tại Telnet là chương trình rất cần thiết cho công việc quản trị hàng ngày, nhưng nó không an toàn khi nó truyền mật khẩu của bạn ở dạng văn bản (plain text) thông qua mạng và cho phép bất kỳ trình lắng nghe nào (listener), theo cách này tin tặc sử dụng tài khoản của bạn để làm bất kỳ công việc phá hoại nào mà hắn ta muốn. Để giải quyết vấn đề này chúng ta phải tìm ra một cách khác, hoặc một chương trình để thay thế nó. Thật may mắn OpenSSH là một dịch vụ thật sự vững chắc và bảo mật có thể thay thế cho cách cũ, các chương trình login từ xa không an toàn và cổ xưa chẳng hạn như telnet, rlogin, rsh, rdist hay rcp.

Thông qua tập tin README chính thức của OpenSSH:

Ssh (Secure Shell) là một chương trình để log vào một máy tính khác thông qua một hệ thống mạng, để thi hành các lệnh trong một máy tính ở xa, và để chuyển các tập tin từ một máy này tới một máy khác. Nó cung cấp tính năng xác nhận hợp lệ "authentication" và bảo mật sự trao đổi thông tin qua các kênh truyền dẫn không an toàn. Nó cũng được dự trù để thay thế cho các chương trình rlogin, rsh, và rdist.

Trong việc cấu hình, chúng ta phải cấu hình OpenSSH hỗ trợ tcp-wrappers (inetd super server) để cải tiến việc bảo mật cho chương trình bảo mật sẵn có và luôn luôn tránh việc phải chạy trình daemon của nó theo kiểu background trên máy server. Theo cách này, chương trình sẽ chỉ chạy khi máy khách (client) kết nối đến và sẽ tái thiết lập lại chúng thông qua trình daemon TCP-WRAPPERS cho việc xác minh tính đúng đắn và cho phép trước khi được phép kết nối tới máy server. OpenSSH thì miễn phí, một sự thay thế và cải tiến của SSH1 với tất cả các cản trở của các giải thuật sáng tạo được công nhận bị xoá bỏ (và trở thành các thư viện được mở rộng ra bên ngoài), tất cả các lỗi được nhận biết đã được sữa chữa, các đặc trưng mới được giới thiệu và rất nhiều trình dọn dẹp rác (clean-up) khác. Điều được khuyên là bạn dùng phiên bản SSH (miễn phí và các lỗi đã được sửa) thay cho bản SSH1 (miễn phí, còn lỗi, và lỗi thời) hay SSH2 mà có nguồn gốc là được miễn phí nhưng hiện nay đã trở thành một phiên bản thương mại. Đối với tất cả mọi người mà dùng SSH2 như công ty Datafellows, chúng tôi sẽ cung cấp trong quyển sách này cả hai phiên bản, và bắt đầu với OpenSSH, và xem nó như là một chương trình SSH mới mà mọi người sẽ phải chuyển sang sử dụng nó trong tương lai.

Hướng dẫn cài đặt

Các dòng lệnh tương thích với Unix.

Đường dẫn nguồn sẽ là "/var/tmp"(cũng có thể chọn những đường dẫn khác).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Số phiên bản OpenSSH là 1.2.3

Các gói cần thiết (Packages)

OpenSSH Homepage Site: http://violet.ibs.com.au/openssh/

Bạn phải bảo đảm là tải về gói : openssh-1.2.3.tar.gz

Những điều kiện cần thiết

OpenSSH yêu cầu là phải có gói zlib-devel, gói dữ liệu này chứa đựng các tập tin header và thư viện cần thiết cho việc phát triển các chương trình cần dùng thư viện nén và giải nén zlib, và đã được cài đặt trên hệ thống của bạn. Nếu trong trường hợp chưa có, thì bạn phải cài đặt nó từ Red Hat Linux 6.1 hay 6.2 CD-ROM.

- Để kiểm tra rằng gói zlib-devel là được cài đặt trên hệ thống Linux của bạn, dùng lệnh sau:

```
[root@deep /]# rpm -qi zlib-devel
package zlib-devel is not installed
```

- Để cài đặt gói zlib-devel trên hệ thống Linux của bạn, dùng lệnh sau:

```
[root@deep /]# mount /dev/cdrom /mnt/cdrom/
[root@deep /]# cd /mnt/cdrom/Red Hat/RPMS/
[root@deep RPMS]# rpm -Uvh zlib-devel-version.i386.rpm
gd
##################################################
[root@deep RPMS]# rpm -Uvh gd-devel-version.i386.rpm
zlib-devel
##################################################
[root@deep RPMS]# cd /; umount /mnt/cdrom/
```

- OpenSSL cho phép hỗ trợ cho các chức năng SSL, nếu bạn muốn dùng tính năng này thì các phần mềm về SSL phải được cài đặt sẵn trên hệ thống của bạn để có thể dùng phần mềm OpenSSH.

Ghi chú: *Để biết thêm thông tin về OpenSSL Server, xem các chương liên quan của nó trong quyển sách này. Thậm chí nếu bạn không cần dùng phần*

mềm OpenSSL để tạo hay lưu giữ các tập tin khoá được mã hoá (encrypted key files), điều quan trọng đáng lưu ý là chương trình OpenSSH yêu cầu các tập tin thư viện của nó có thể làm việc một cách chính xác trên hệ thống của bạn.

Dạng lưu trữ Tarballs

Bạn nên tạo ra một danh sách các tập tin trên hệ thống trước và sau khi cài đặt OpenSSH, và sau đó so sánh chúng dùng lệnh **"diff"** để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh **"find /* > OpenSSH1"** trước và **"find /* > OpenSSH2"** sau khi bạn cài đặt phần mềm, và sử dụng lệnh **"diff OpenSSH1 OpenSSH2 > OpenSSH-Installed"** để lấy một danh sách các thay đổi nào đó.

Sự biên dịch

Giải nén dạng lưu trữ tarball (tar.gz)

```
[[root@deep /]# cp openssh-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf openssh-version.tar.gz
```

Biên dịch và tối ưu

Bước 1

Chuyển vào trong thư mục OpenSSH mới và gõ các lệnh sau trên terminal của bạn:

```
CC="egcs" \
CFLAGS="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomit-frame-pointer -fno-exceptions" \
./configure \
--prefix=/usr \
--sysconfdir=/etc/ssh \
--with-tcp-wrappers \
--with-ipv4-default \
--with-ssl-dir=/usr/include/openssl
```

Việc này bảo OpenSSH tự thiết lập cấu hình cho việc cài đặt các phần cứng riêng biệt với:

- Hỗ trợ biên dịch libwrap và cho phép tính năng TCP Wrappers (/etc/hosts.allow I deny).
- Không cho phép việc trì hoãn lâu trong việc nhận biết tên dưới Linux/glibc-2.1.2 để cải tiến thời gian kết nối.
- Chỉ rõ vị trí các thư viện của OpenSSL được yêu cầu bởi chương trình OpenSSH để làm việc.

Bước 2

Bây giờ, chúng ta phải biên dịch và cài đặt OpenSSH trên máy server:

```
[root@deep openssh-1.2.3]# make
[root@deep openssh-1.2.3]# make install
[root@deep openssh-1.2.3]# make host-key
[root@deep openssh-1.2.3]# install -m644 contrib/Red Hat/sshd.pam /etc/pam.d/sshd
```

Lệnh "make" biên dịch tất cả các tập tin nguồn sang dạng nhị phân có thể thi hành được, và lệnh "make install" sẽ cài đặt các tập tin nhị phân và bất kỳ các tập tin hỗ trợ vào các nơi thích hợp. Lệnh "make host-key" sẽ sinh ra một khoá chính "host key", và sau cùng lệnh "install" sẽ cài đặt hỗ trợ PAM cho Red Hat Linux, mà hiện nay có nhiều chức năng hơn các package của bản thương mại ssh-1.2.x quen thuộc.

Dọn dẹp sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf openssh-version/ openssh-version.tar.gz
```

Lệnh "**rm**" sẽ xoá tất cả các tập tin nguồn mà chúng ta dùng để biên dịch và cài đặt OpenSSH. Nó cũng sẽ xoá tập tin dạng lưu trữ nén OpenSSH từ thư mục "/var/tmp".

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy OpenSSH Client/Server, các tập tin sau được yêu cầu và phải được tạo hay được sao chép vào các thư mục thích hợp trên máy server của bạn:

  Sao chép tập tin ssh_config vào trong thư mục "/etc/ssh/".

  Sao chép tập tin sshd_config vào trong thư mục "/etc/ssh/".

  Sao chép tập tin sshd vào trong thư mục "/etc/pam.h/".

Cấu hình cho tập tin "/etc/ssh/ssh_config"

Tập tin "/etc/ssh/ssh_config" là tập tin cấu hình tầm ảnh hưởng rộng của hệ thống (system-wide) cho OpenSSH mà cho phép bạn cài đặt các tùy chọn nhằm thay đổi sự hoạt động của các chương trình client. Tập tin chứa đựng các cặp giá trị keyword-value, một cặp trên một dòng, với các từ khóa (keyword) là các ký tự thường. Đây là hầu hết các từ khóa quan trọng cấu hình "ssh" của bạn cho việc bảo mật cao nhất; một danh sách đầy đủ và/hoặc các yêu cầu đặc biệt là để có sẵn trong trang man cho ssh (1).

Soạn thảo tập tin cấu hình ssh_config (vi /etc/ssh/ssh_config) và thêm/hay thay đổi, khi các thông số sau cần thiết:

```
# Site-wide defaults for various options
```

```
Host *
ForwardAgent no
ForwardX11 no
RhostsAuthentication no
RhostsRSAAuthentication no
RSAAuthentication yes
PasswordAuthentication yes
FallBackToRsh no
UseRsh no
BatchMode no
CheckHostIP yes
StrictHostKeyChecking no
IdentityFile ~/.ssh/identity
Port 22
Cipher blowfish
EscapeChar ~
```

Việc này bảo ssh_config tự thiết lập cấu hình riêng biệt với:

```
Host*
```

Tùy chọn "Host" hạn chế tất cả các khai báo được chuyển tới (forward) và các tùy chọn trong tập tin cấu hình là chỉ dành cho các host mà nó phải thích hợp với một trong các mẫu (pattern) được cung cấp sẵn sau từ khoá. Kiểu mẫu "*" nghĩa là dành cho tất cả các host và nằm cạnh bên từ khóa Host. Với tùy chọn này bạn có thể đặt các khai báo khác nhau cho các host khác nhau nằm cùng trong tập tin "ssh_config".

ForwardAgent no

Tùy chọn "ForwardAgent" chỉ rõ tác nhân kết nối xác thực (connection authentication) nào (nếu có) sẽ được chuyển tới máy ở xa.

ForwardX11 no

Tùy chọn "ForwardX11" là dùng cho những người mà sử dụng giao diện Xwindow (Xwindow GUI) và muốn tự động gửi X11 session một lần nữa tới máy ở xa. Bởi vì chúng ta cài đặt một server và không có cài đặt giao diện người dùng (GUI) lên nó, chúng ta có thể tắt tùy chọn này một cách an toàn.

RhostsAuthentication no

Tùy chọn "RhostsAuthentication" định rõ hoặc chúng ta có thể thử sử dụng việc kiểm tra tính xác thực dựa trên rhosts. Bởi vì rhosts authentication thì không bảo mật nên bạn không nên dùng tùy chọn này.

RhostsRSAAuthentication no

Tùy chọn "RhostsRSAAuthentication" xác định rõ hoặc có thể hoặc không thử dùng rhosts authentication trong sự phối hợp với RSA rhosts authentication.

RSAAuthentication yes

Tùy chọn "RSAAuthentication" định rõ hoặc chỉ thử dùng RSA authentication. Tùy chọn này phải được đặt là "yes" để việc bảo mật được tốt hơn trên các phiên (session) của bạn. RSA dùng cặp khoá public và private được tạo với tiện ích ssh-keygen cho các mục đích kiểm tra tính xác thực.

PasswordAuthentication yes

Tùy chọn "PasswordAuthentication" chỉ rõ hoặc là chúng ta nên dùng tính xác thực dựa trên mật mã. Để bảo mật tốt, tùy chọn này phải luôn được đặt là "yes".

FallBackToRsh no

Tùy chọn "FallBackToRsh" chỉ rõ rằng nếu một kết nối với một ssh daemon không thành công thì rsh nên được dùng thay vào đó một cách tự động. Nhắc lại là dịch vụ rsh thì không được bảo mật, tùy chọn này phải luôn được đặt là "no".

UseRsh no

Tùy chọn "UseRsh" chỉ rõ rằng các dịch vụ rlogin/rsh nên được dùng trên host này. Cũng giống như tùy chọn "FallBackToRsh", nó phải được đặt là "no" vì các lý do hiển nhiên.

BatchMode no

Tùy chọn "BatchMode" chỉ rõ việc truy vấn một user-name hay password trên kết nối sẽ bị vô hiệu hóa. Tùy chọn này là tiện ích khi bạn tạo các script và không cần phải cung cấp một password. (Nghĩa là các script là được dùng lệnh scp để tạo bản dự phòng thông qua mạng).

CheckHostIP yes

Tùy chọn "CheckHostIP" chỉ rõ là có hay không ssh sẽ kiểm tra thêm địa chỉ IP của host kết nối tới máy server để nhận biết DNS spoofing. Một điều được khuyên là bạn nên đặt tùy chọn này là "yes".

StrictHostKeyChecking no

Tùy chọn "StrictHostKeyChecking" chỉ rõ là có hay không ssh sẽ tự động thêm các khóa chính (host key) mới vào tập tin $HOME/.ssh/known_hosts, hay là không bao giờ thêm khóa chính mới một cách tự động vào tập tin host. Tùy chọn này, khi được đặt "yes", sẽ cung cấp sự bảo vệ tối đa chống lại kiểu tấn công con ngựa thành Troa (Trojan horse). Một thủ tục được ưa dùng với tùy chọn này là ban đầu đặt nó là "no", cho phép ssh thêm một cách tự động tất cả các host chung vào

tập tin host khi chúng đã được kết nối, và sau đó quay trở lại và đặt nó là "yes" để tận dụng được các đặc trưng tối ưu của nó.

IdentityFile ~/.ssh/identity

Tùy chọn "IdentityFile" chỉ rõ một tập tin nhận dạng kiểm tra tính xác thực RSA (RSA authentication) luân phiên để đọc. Ngoài ra, nhiều tập tin nhận dạng có thể được định rõ trong tập tin cấu hình (ssh_config).

Port 22

Tùy chọn "port" xác định rõ số cổng nào mà ssh kết nối đến một host ở xa. Mặc nhiên của nó là 22.

Cipher blowfish

Tùy chọn "Cipher" xác định rõ là mật mã (cipher) nào sẽ được dùng cho việc mã hoá các phiên (session). Blowfish dùng các khối (block) 64 bit và các khoá (key) lên đến 448 bit.

EscapeChar ~

Tùy chọn "EscapeChar" xác định rõ ký tự escape dùng cho việc dừng lại.

Cấu hình cho tập tin "/etc/ssh/sshd_config"

Tập tin "/etc/ssh/sshd_config" là tập tin cấu hình hệ thống lớn (system-wide) cho OpenSSH, tập tin này cho phép bạn cài đặt các tùy chọn mà thay đổi sự hoạt động của chương trình daemon. Tập tin này chứa đựng các giá trị cặp từ khoá "keyword-value", một cặp trên một dòng, với các từ khóa (keyword) ở dạng ký tự thường. Đây là các từ khóa quan trọng nhất để cấu hình "sshd" của bạn với sự bảo mật nhất; một danh sách đầy đủ và/hoặc các yêu cầu đặc biệt đã có sẵn trong trang man cho sshd (8).

Hiệu chỉnh tập tin cấu hình sshd_config (vi /etc/ssh/sshd_config), thêm hoặc thay đổi khi cần thiết các thông số sau:

```
# This is ssh server systemwide configuration file.

Port 22
ListenAddress 192.168.1.1
HostKey /etc/ssh/ssh_host_key
ServerKeyBits 1024
LoginGraceTime 600
KeyRegenerationInterval 3600
PermitRootLogin no
IgnoreRhosts yes
```

```
IgnoreUserKnownHosts yes
StrictModes yes
X11Forwarding no
PrintMotd yes
SyslogFacility AUTH
LogLevel INFO
RhostsAuthentication no
RhostsRSAAuthentication no
RSAAuthentication yes
PasswordAuthentication yes
PermitEmptyPasswords no
AllowUsers admin
```

Việc này bảo sshd_config tự thiết lập cấu hình cho việc cài đặt cấu hình riêng biệt với:

Port 22

Tùy chọn "port" xác định rõ trên cổng nào mà ssh daemon lắng nghe các kết nối đến. Mặc nhiên của nó là 22.

ListenAddress 192.168.1.1

Tùy chọn "ListenAddress" chỉ rõ địa chỉ IP của giao tiếp mạng trên socket của ssh daemon server nào được ràng buộc. Mặc định là "0.0.0.0"; để cải tiến bảo mật bạn có thể định rõ chỉ một số các yêu cầu nào đó cho giới hạn các địa chỉ có thể nào đó.

HostKey /etc/ssh/ssh_host_key

Tùy chọn "HostKey" chỉ rõ vị trí chứa đựng khóa chính (private host key).

ServerKeyBits 1024

Tùy chọn "ServerKeyBits" chỉ rõ có bao nhiêu bit được dùng trong server key. Các bit này được dùng khi trình daemon khởi động để sinh ra các RSA key.

LoginGraceTime 600

Tùy chọn "LoginGraceTime" chỉ rõ bao nhiêu giây sau khi một sự kết nối yêu cầu server sẽ chờ trước khi nó ngừng kết nối (disconnect) nếu người dùng log in không thành công.

KeyRegenerationInterval 3600

Tùy chọn "KeyRegenerationInterval" chỉ rõ thời gian bao nhiêu giây server sẽ chờ trước khi tự động sinh ra các key của nó. Đây là một đặc trưng bảo mật để ngăn chặn các session giải mã được ghi lại.

PermitRootLogin no

Tùy chọn "PermitRootLogin" chỉ rõ có hay không root có thể login dùng ssh. Không bao giờ đặt "yes" cho tùy chọn này.

IgnoreRhosts yes

Tùy chọn "IgnoreUserKnownHosts" chỉ rõ có hay không các tập tin rhost hay shosts sẽ không được dùng trong authentication. Để phục vụ cho các lý do bảo mật có một điều khuyên cho bạn là không dùng các tập tin rhosts hay shosts cho authentication.

IgnoreUserKnownHosts yes

Tùy chọn "IgnoreUserKnownHosts" chỉ rõ có hay không chương trình ssh daemon sẽ bỏ qua "$HOME/.ssh/known_hosts" của user trong lúc kiểm tra tính xác thực RhostsRSAAuthentication.

StrictModes yes

Tùy chọn "StrictModes" chỉ rõ có hay không ssh sẽ kiểm tra các quyền của người sử dụng trong thư mục home của họ và các tập tin rhosts trước khi chấp nhận login. Tùy chọn này phải luôn được đặt là "yes" bởi vì đôi khi người sử dụng có thể tình cờ bỏ quên các thư mục hay tập tin có thể ghi của họ.

X11Forwarding no

Tùy chọn "X11Forwarding" chỉ rõ có hay không chức năng X11 forwarding sẽ được hỗ trợ trên server này. Vì chúng ta cài đặt một server không có GUI được cài đặt trên nó nên chúng ta có thể an tâm tắt tùy chọn này.

PrintMotd yes

Tùy chọn "PrintMotd" chỉ rõ có hay không ssh daemon sẽ in ra nội dung của tập tin "/etc/motd" khi một người sử dụng bắt đầu tương tác. Tập tin "/etc/motd" thì cũng được biết đến như "thông điệp trong ngày".

SyslogFacility AUTH

Tùy chọn "SyslogFacility" chỉ rõ mã được dùng khi ghi lại thông điệp từ sshd. Điều kiện định rõ hệ thống con nào mà nó sinh ra thông điệp – trong trường hợp của chúng ta, AUTH.

LogLevel INFO

Tùy chọn "*LogLevel*" chỉ rõ mức độ được dùng khi ghi lại thông điệp từ sshd. INFO là một chọn lựa tốt. Xem trang man về sshd để biết thêm thông tin.

RhostsAuthentication no

Tùy chọn "RhostsAuthentication" định rõ có hay không sshd có thể dùng việc kiểm tra tính xác thực dựa trên rhosts. Bởi vì rhosts authentication thì không an toàn, bạn không nên dùng tùy chọn này.

RhostsRSAAuthentication no

Tùy chọn "RhostsRSAAuthentication" định rõ có hay thể dùng rhosts authentication trong sự phối hợp với RSA host authentication.

RSAAuthentication yes

Tùy chọn "RSAAuthentication" chỉ rõ có hay không thử dùng RSA authentication. Tùy chọn này phải được đặt là "yes" để việc bảo mật được tốt hơn trên các phiên (session) của bạn. RSA dùng cặp khoá public và private được tạo với trình tiện ích ssh-keygen cho các mục đích kiểm tra tính xác thực.

PasswordAuthentication yes

Tùy chọn "PasswordAuthentication" chỉ rõ có hay không chúng ta sẽ sử dụng việc kiểm tra tính xác thực dựa trên mật khẩu. Để bảo mật tốt, tùy chọn này phải luôn được đặt là "yes".

PermitEmptyPasswords no

Tùy chọn "PermitEmptyPasswords" chỉ rõ có hay không server cho phép log in vào với tài khoản có password là null. Nếu bạn dự định dùng tiện ích "scp" để tạo bản dự phòng tự động thông qua mạng, bạn phải đặt tùy chọn này là "yes".

AllowUsers admin

Tùy chọn "AllowUsers" chỉ rõ và điều khiển những người dùng nào có thể truy cập các dịch vụ ssh. Nhiều người sử dụng có thể được chỉ định, được tách biệt bằng các khoảng trống.

Cấu hình OpenSSH để dùng TCP-Wrappers inetd super server

TCP-Wrappers nên được hỗ trợ để khởi động và dừng server OpenSSH của chúng ta. Vào lúc thi hành, inetd đọc thông tin cấu hình của nó từ một tập tin cấu hình mặc nhiên là "/etc/inetd.conf". Có một entry cho mỗi trường (field) của tập tin cấu hình, với các entry cho mỗi trường được tách riêng biệt bởi một tab hoặc một khoảng trống (space).

Bước 1

Soạn thảo tập tin **inetd.conf** (vi /etc/inetd.conf) và thêm vào dòng sau:

ssh stream tcp nowait root /usr/sbin/tcpd sshd -i

Ghi chú: *Tham số "-i" thì quan trọng khi nó chỉ rõ là sshd được chạy từ inetd. Ngoài ra, để việc cập nhật tập tin "inetd.conf" có hiệu lực bạn gửi một tín hiệu SIGHUP (killall -HUP inetd) sau khi thêm dòng trên vào tập tin.*

- Để cập nhật tập tin "inetd.conf", dùng lệnh sau:

```
[root@deep /]# killall -HUP inetd
```

Bước 2

Soạn thảo tập tin hosts.allow (vi /etc/hosts.allow) và thêm vào dòng sau:

sshd: 192.168.1.4 win.openna.com

Có nghĩa là một địa chỉ IP của máy client "192.168.1.4" với tên host là "win.openna.com" thì được cho phép dùng dịch vụ ssh trên máy server.

Các chương trình "daemon" (cho tcp-wrappers) được dùng bởi sshd:

sshdfwd-X11 (Nếu bạn muốn cho phép/từ chối X11-forwarding).

sshdfwd-<port-number> (cho tcp-forwarding).

sshdfwd-<port-name> (port-name được định nghĩa trong tập tin /etc/services. Được dùng trong tcp-forwarding).

***Ghi chú**: Nếu bạn quyết định dùng ssh, hãy bảo đảm rằng bạn phải cài đặt và dùng nó trên tất cả máy server của bạn. Và bạn cũng nên nhớ rằng nếu như bạn có mười máy server bảo mật và chỉ có một máy server không bảo mật thì đó là một việc làm lãng phí thời gian.*

Tài liệu bổ sung

Muốn biết thêm chi tiết, bạn có thể tham khảo thêm nhiều trang man sau:

$ man ssh (1)	- OpenSSH secure shell client (remote login program)
$ man ssh [slogin] (1)	- OpenSSH secure shell client (remote login program)
$ man ssh-add (1)	- adds identities for the authentication agent
$ man ssh-agent (1)	- authentication agent
$ man ssh-keygen (1)	- authentication key generation
$ man sshd (8)	- secure shell daemon

Cấu hình OpenSSH cho mỗi người sử dụng

Bước 1

Tạo các khóa cục bộ private & public của bạn, bằng cách thi hành:

```
[root@deep /]# su admin
[admin@deep /]$ ssh-keygen
```

Kết quả nên giống như ví dụ sau:

```
Initializing random number generator...
Generating p: .....++ (distance 430)
Generating q: .....++ (distance 456)
Computing the keys...
```

Testing the keys...

Key generation complete.

Enter file in which to save the key (/home/admin/.ssh/identity): **[Press Enter]**

Enter passphrase:

Enter the same passphrase again:

Your identification has been saved in /home/admin/.ssh/identity.

Your public key is:

1024 37

14937757511251955533691120318477293862290049394715136511145806
10887000176437849467683129757784315853227236120610062314604405
36487184367748423324091941848098890786099717524446977589647127
75703072877997370856999301704314156353633306888894403817846160
85924838445902021541027569030558465340633656355848997654021 81
admin@deep.openna.com

Your public key has been saved in /home/admin/.ssh/identity.pub

Ghi chú: *Nếu bạn có nhiều tài khoản bạn có thể muốn tạo một khóa riêng biệt trên mỗi tài khoản sử dụng. Bạn có thể có các khóa riêng biệt cho:*

- Mail server của bạn.
- Web server của bạn.
- Gateway (GW) server của bạn.

Điều này cho phép bạn giới hạn việc truy cập giữa các server này, nghĩa là không cho phép dùng tài khoản Mail truy cập tới tài khoản Web của bạn hay các máy trong GW. Điều này tăng thêm tính bảo mật trên toàn bộ hệ thống trong trường hợp bất kỳ các khoá xác thực nào đã được thỏa hiệp cho bất kỳ lý do nào.

Bước 2

Sao chép các public key cục bộ của bạn (**identity.pub**) tới thư mục của admin trên một máy ở xa "/home/admin/.ssh" với tên được gọi là "authorized_keys".

Ghi chú: *Một cách để sao chép tập tin là dùng lệnh ftp hay bạn có thể gửi public key của bạn bằng email tới người quản trị hệ thống. Chỉ bao gồm nội dung của tập tin "~/.ssh/identity.pub" trong thông điệp.*

Thay đổi pass-phrase của bạn

Bạn có thể thay đổi pass-phrase vào bất kỳ thời điểm nào bằng cách dùng tùy chọn –p của ssh-keygen.

- Để thay đổi pass-phrase, dùng lệnh:

[root@deep /]# **su admin**

[admin@deep /]$ **ssh-keygen -p**

Enter file key is in (/home/admin/.ssh/identity): [Press ENTER]

Enter old passphrase:

Key has comment 'admin@deep.openna.com'

Enter new passphrase:

Enter the same passphrase again:

Your identification has been saved with the new passphrase.

Các công cụ OpenSSH cho người sử dụng

Các lệnh được liệt kê bên dưới là một số lệnh mà chúng ta thường dùng, nhưng hiện nay nó đã được bổ sung thêm nhiều lệnh hơn, và bạn nên kiểm tra trang man và tài liệu để biết thêm chi tiết.

ssh

Lệnh "ssh" (Secure Shell) cung cấp dịch vụ bảo mật sự trao đổi thông tin giữa hai host không được tin cậy thông qua một mạng không bảo mật. Đó là một chương trình dùng cho việc đăng nhập (login) một cách bảo mật vào trong một máy ở xa và thi hành một số lệnh ở đó. Đó là một sự thay thế tuyệt vời cho các chương trình không bảo mật như telnet, rlogin, rcp, rdist và rsh.

- Để login vào một máy ở xa, dùng lệnh:

  [root@deep /]# **ssh -l <login_name> <hostname>**

  Lấy ví dụ:

  [root@deep /]# **ssh -l admin www.openna.com**

  admin@deep.openna.com's password:

  Last login: Tue Oct 19 1999 18:13:00 -0400 from deep.openna.com

  No mail.

  [admin@www admin]$

Trong đó <login_name> là tên mà bạn dùng để kết nối tới ssh server và <hostname> là địa chỉ của ssh server của bạn ở xa.

scp

Tiện ích "scp" (Secure Copy) sao chép các tập tin từ hệ thống cục bộ tới một một hệ thống ở xa hoặc ngược lại hay thậm chí giữa hai hệ thống từ xa dùng lệnh scp:

- Để sao chép các tập tin từ máy ở xa về hệ thống cục bộ, dùng lệnh sau:

  [root@deep /]# **su admin**

  [admin@deep /]$ **scp -p <login_name@hostname>:/dir/for/file**

localdir/to/filelocation

For example:

[admin@deep /]$ scp1 -p admin@mail:/etc/test1 /tmp

Enter passphrase for RSA key 'admin@mail.openna.com':

test1 | 2 KB | 2.0 kB/s | ETA: 00:00:00 | 100%

- Để sao chép các tập tin từ máy cục bộ tới hệ thống ở xa, dùng lệnh sau:

[root@deep /]# **su admin**

[admin@deep /]$ **scp -p localdir/to/filelocation**

<username@hostname>:/dir/for/file

For example:

[admin@deep /]$ scp1 -p /usr/bin/test2 admin@mail:/var/tmp

admin@mail's password:

test2 | 7 KB | 7.9 kB/s | ETA: 00:00:00 | 100%

Ghi chú: *Tùy chọn "-p" chỉ ra rằng việc thay đổi và thời gian truy cập, cũng giống như các mode của tập tin nguồn, nên giữ lại bản sao chép. Đây là mong muốn thường xuyên.*

Một số cách dùng phần mềm OpenSSH

OpenSSH có thể được dùng để :

1. Thay thế telnet, rlogin, rsh, rdist, và rcp.
2. Thực hiện việc sao chép bảo mật thông qua mạng.
3. Thi hành các lệnh từ xa
4. Truy cập các tài nguyên tổng hợp thông qua internet.

Các tập tin được cài đặt

> /etc/ssh
> /etc/ssh/ssh_config
> /etc/ssh/sshd_config
> /etc/ssh_host_key
> /etc/ssh_host_key.pub
> /usr/bin/ssh
> /usr/bin/slogin
> /usr/man/man1/ssh.1
> /usr/man/man1/scp.1
> /usr/man/man1/ssh-add.1
> /usr/man/man1/ssh-agent.1
> /usr/man/man1/ssh-keygen.1

> /usr/bin/scp
> /usr/bin/ssh-add
> /usr/bin/ssh-agent
> /usr/bin/ssh-keygen
> /usr/man/man1/slogin.1
> /usr/man/man8/sshd.8
> /usr/sbin/sshd

Các chương trình SSH client miễn phí chạy trên Windows

Putty

Putty Homepage: http://www.chiark.greenend.org.uk/~sgtatham/putty.html

Tera Term Pro and TTSSH

Tera Term Pro Homepage:
http://hp.vector.co.jp/authors/VA002416/teraterm.html

TTSSH Homepage: http://www.zip.com.au/~roca/download.html

Linux SSH2 Client/Server

Giới thiệu tổng quan

Chương trước đã cung cấp cho bạn những thông tin khá rõ ràng để những ai dùng SSH2 trên Linux sẽ được thay bằng OPENSSH như trường hợp dùng OpenSSH thay cho SSH2 từ công ty Datafellows Company. Tuy nhiên, với các cá nhân hay các tổ chức muốn dùng phiên bản không miễn phí của phần mềm này, chúng tôi cung cấp ở đây các bước như sau. Đây là phiên bản phần mềm SSH2 thương mại. Trong việc cấu hình, chúng ta cũng phải cấu hình sshd2 để hỗ trợ tcp-wrappers (là inetd super server) và phục vụ cho các lý do bảo mật.

Hướng dẫn cài đặt

Các dòng lệnh là tương thích với Unix.

Đường dẫn nguồn sẽ là "/var/tmp"(cũng có thể chọn những đường dẫn khác).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Số phiên bản SSH2 là 2.0.13.

Các gói cần thiết (Packages)

SSH2 Homepage Site: http://www.ssh.org

Bạn phải bảo đảm là tải về gói : ssh-2.0.13.tar.gz

Dạng lưu trữ Tarballs

Tốt nhất bạn nên tạo ra danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt ssh2, và sau đó so sánh chúng dùng lệnh **"diff"** để tìm ra các tập tin nào được cài đặt ở đâu. Đơn giản chạy lệnh **"find /* > SSH1"** trước và **"find /* > SSH2"** sau khi bạn cài đặt phần mềm, và sử dụng lệnh **"diff SSH1 SSH2 > SSH –Installed"** để lấy một danh sách các thay đổi nào đó.

Sự biên dịch

Giải nén dạng lưu trữ tarball (tar.gz)

```
[root@deep /]# cp ssh-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf ssh-version.tar.gz
```

Biên dịch và tối ưu

Chuyển vào trong thư mục SSH2 mới và gõ các lệnh sau trên terminal của bạn:

```
CC="egcs" \
```

```
CFLAGS="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomit-frame-pointer -fno-exceptions" \
./configure \
--prefix=/usr \
--without-ssh-agent1-compat \
--disable-suid-ssh-signer \
--disable-tcp-port-forwarding \
--disable-X11-forwarding \
--enable-tcp-nodelay \
--with-libwrap
```

Việc này bảo SSH2 tự thiết lập cấu hình cho việc cài đặt các phần cứng riêng biệt với:

- Bỏ đi khả năng ssh-agent1.
- Cài đặt ssh-signer không có bit suid.
- Không cho phép hỗ trợ port forwarding.
- Không cho phép hỗ trợ X11 forwarding.
- Cho phép tùy chọn TCP_NODELAY socket.
- Biên dịch hỗ trợ libwrap (tcp_wrappers).

```
[root@deep ssh-2.0.13]# make clean
[root@deep ssh-2.0.13]# make
[root@deep ssh-2.0.13]# make install
[root@deep ssh-2.0.13]# rm -f /usr/bin/ssh-askpass
```

Lệnh "**make clean**" xoá tất cả các dấu vết trước đó của một sự biên dịch, thực hiện như vậy để tránh bất kỳ lỗi nào. Lệnh "**make**" biên dịch tất cả các tập tin nguồn sang dạng nhị phân có thể thi hành. Sau cùng là lệnh "**make install**" cài đặt các tập tin nhị phân này và mọi tập tin hỗ trợ vào các nơi thích hợp.

Dọn dẹp sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf ssh-version/ ssh-version.tar.gz
```

Lệnh "**rm**" sẽ xoá tất cả các tập tin nguồn mà chúng ta dùng để biên dịch và cài đặt SSH2. Nó cũng sẽ xoá dạng lưu trữ nén SSH2 từ thư mục "/var/tmp".

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy SSH2 Client/Server, các tập tin sau được yêu cầu và phải được tạo hay được sao chép vào các thư mục thích hợp trên máy server của bạn:

  Sao chép tập tin sshd2_config vào trong thư mục "/etc/ssh2/".

  Sao chép tập tin ssh2_config vào trong thư mục "/etc/ssh2/".

  Sao chép tập tin ssh vào trong thư mục "/etc/pam.h/".

Bạn có thể lấy các tập tin cấu hình được liệt kê bên dưới dựa trên tập tin lưu trữ dạng nén floppy.tgz trên đĩa mềm kèm theo sách. Sao chép các tập tin sau từ dạng lưu trữ được giải nén floppy.tgz vào các nơi thích hợp hay sao chép và dán chúng trực tiếp từ quyển sách này vào tập tin có liên quan.

Cấu hình cho tập tin "/etc/ssh2/ssh2_config"

Tập tin cấu hình cho ssh2 "/etc/ssh2/ssh2_config" cho phép bạn thiết đặt nhiều tùy chọn, các tùy chọn này thay đổi sự hoạt động của các chương trình client. Tập tin chứa đựng các cặp keyword-value, một cặp trên một dòng, các từ khóa (keyword) ở dạng các ký tự thường. Đây là hầu hết các từ khóa quan trọng để cấu hình "ssh" của bạn cho bảo mật nhất; một danh sách đầy đủ và các yêu cầu đặc biệt đã có sẵn trong trang man cho ssh (2).

Soạn thảo tập tin cấu hình ssh2_config (vi /etc/ssh/ssh2_config) và thêm/hay thay đổi các thông số sau nếu cần thiết:

```
# ssh2_config
# SSH 2.0 Client Configuration File

*:
Port                    22
Ciphers                 blowfish
Compression             yes
IdentityFile            identification
AuthorizationFile       authorization
RandomSeedFile          random_seed
VerboseMode             no
ForwardAgent            no
ForwardX11              no
PasswordPrompt          "%U's password: "
Ssh1Compatibility       no
Ssh1AgentCompatibility  none
NoDelay                 yes
```

KeepAlive	yes
QuietMode	**no**

Việc này bảo ssh2_config tự thiết lập cấu hình cho việc cài đặt cấu hình riêng biệt với:

Port 22

Tùy chọn "Port" xác định rõ số cổng nào mà ssh kết nối đến một host ở xa. Port mặc định là 22.

Ciphers blowfish

Tùy chọn "Ciphers" xác định rõ mật mã (cipher) nào sẽ được sử dụng cho các phiên (session) mã hóa. Blowfish dùng các block 64 bit và các key lên đến 448 bit.

Compression yes

Tùy chọn "Compression" định rõ có hay không sử dụng kỹ thuật nén (compression) trong suốt các session. Việc nén sẽ cải tiến tốc độ truyền thông và việc truyền các tập tin.

IdentityFile identification

Tùy chọn "IdentityFile" chỉ rõ một tên luân phiên cho tập tin nhận diện (identification) của người sử dụng.

AuthorizationFile authorization

Tùy chọn "AuthorizationFile" chỉ rõ một tên luân phiên cho tập tin authorization của người sử dụng.

RandomSeedFile random_seed

Tùy chọn "RandomSeedFile" chỉ rõ một tên luân phiên cho tập tin khởi đầu ngẫu nhiên (random seed file) của người sử dụng.

VerboseMode no

Tùy chọn "VerboseMode" chỉ thị cho ssh2 in ra các thông điệp debug về sự tiến trình của nó. Tùy chọn này là giúp ích trong việc sửa lỗi sự kết nối, tính xác thực và các vấn đề cấu hình.

ForwardAgent no

Tùy chọn "ForwardAgent" chỉ rõ tác nhân tính xác thực của sự kết nối nào (nếu có) sẽ được chuyển tới máy ở xa.

ForwardX11 no

Tùy chọn "ForwardX11" dùng cho những người mà sử dụng giao diện Xwindow (Xwindow GUI) và muốn tự động gởi một lần nữa (redirect) các X11

session tới máy ở xa. Vì chúng ta cài đặt một server và không có cài đặt giao diện người dùng (GUI) lên nó, chúng ta có thể an toàn tắt tùy chọn này.

PasswordPrompt "%U's password: "

Tùy chọn "PasswordPrompt" chỉ rõ dấu nhắc mật khẩu mà nó sẽ được hiển thị cho người dùng khi kết nối tới một host. Các biến "%U" và "%H" đưa ra tên đăng nhập (login-name) và host của người dùng tương ứng.

Ssh1Compatibility no

Tùy chọn "Ssh1Compatibility" chỉ rõ có hoặc không dùng khả năng tương thích SSH1 với SSH2 cho những người sử dụng ssh1.

Ssh1Agent Compatibility none

Tùy chọn "Ssh1Agent Compatibility" chỉ rõ có hoặc không chuyển tiếp (forward) những kết nối tác nhân SSH1 (SSH1 agent) với kết nối với SSH2 cho những người sử dụng ssh1.

NoDeplay yes

Tùy chọn "NoDeplay" chỉ rõ nếu tùy chọn socket TCP_NODEPLAY được hỗ trợ. Có một điều khuyên là bạn nên đặt tùy chọn này là "yes" để cải tiến sự thực thi network.

KeepAlive yes

Tùy chọn "KeepAlive" chỉ rõ có hay không hệ thống sẽ gửi các thông điệp giữ kết nối (keep alive) tới máy server ở xa. Nếu đặt là "yes" thì khi đó việc mất kết nối tới các máy ở xa sẽ được thông báo.

QuietMode no

Tùy chọn "QuietMode" chỉ rõ có hay không hệ thống chạy trong chế độ yên lặng (quiet mode). Tùy chọn này phải được đặt là "no" bởi vì trong quiet mode này, không có gì được ghi lại trong hệ thống log ngoại trừ các lỗi nặng. Vì chúng ta cần phải có thông tin về các session của người dùng thì vô hiệu hoá tùy chọn này có thể được ưa chuộng hơn.

Cấu hình cho tập tin "/etc/ssh2/sshd2_config"

Tập tin "/etc/ssh2/sshd2_config" là tập tin cấu hình cho phép bạn cài đặt nhiều tùy chọn, các tùy chọn này thay đổi sự hoạt động của chương trình daemon. Tập tin chứa đựng các cặp keyword-value, một cặp trên một dòng, với các từ khóa (keyword) ở dạng các ký tự thường; Đây là các từ khoá quan trọng, một danh sách đầy đủ đã có sẵn trong trang man cho sshd2 (8).

Soạn thảo tập tin cấu hình sshd2_config (vi /etc/ssh2/sshd2_config) và thêm/hay thay đổi các thông số khi cần thiết:

```
# sshd2_config
# SSH 2.0 Server Configuration File
```

```
*
Port                    22
ListenAddress           192.168.1.1
Ciphers                 blowfish
IdentityFile            identification
AuthorizationFile       authorization
HostKeyFile             hostkey
PublicHostKeyFile       hostkey.pub
RandomSeedFile          random_seed
ForwardAgent            no
ForwardX11              no
PasswordGuesses         3
MaxConnections          5
PermitRootLogin         no
AllowedAuthentications  publickey,password
RequiredAuthentications publickey,password
VerboseMode             no
PrintMotd               yes
CheckMail               yes
UserConfigDirectory     "%D/.ssh2"
SyslogFacility          DAEMON
Ssh1Compatibility       no
NoDelay                 yes
KeepAlive               yes
UserKnownHosts          yes
AllowHosts              192.168.1.4
DenyHosts               *
QuietMode               no

# subsystem definitions

subsystem-sftp          sftp-server
```

Việc này bảo sshd_config tự thiết lập cấu hình cho việc cài đặt cấu hình riêng biệt với:

Port 22

Tùy chọn "port" xác định rõ trên số cổng nào mà ssh daemon lắng nghe các kết nối đến. Mặc nhiên của nó là 22.

ListenAddress 192.168.1.1

Tùy chọn "ListenAddress" chỉ rõ địa chỉ IP của giao tiếp mạng trên socket của ssh daemon server nào được ràng buộc. Mặc định là "0.0.0.0"; để cải tiến bảo mật bạn có thể định rõ chỉ một số các yêu cầu nào đó cho giới hạn các địa chỉ có thể nào đó.

Ciphers blowfish

Tùy chọn "Ciphers" xác định rõ là cipher nào nên được dùng cho việc mã hoá các phiên (session). Blowfish dùng các khối (block) 64 bit và các key lên đến 448 bit.

IdentityFile identification

Tùy chọn "IdentityFile" chỉ rõ một tên luân phiên cho tập tin xác nhận (identification) của người sử dụng.

AuthorizationFile authorization

Tùy chọn "AuthorizationFile" chỉ rõ một tên luân phiên cho tập tin authorization của người sử dụng.

HostKeyFile hosthey

Tùy chọn "HostKey" chỉ rõ một tên tập tin chứa đựng private host key. Mặc nhiên là "/etc/ssh2/hostkey".

PublicHostKeyFile hostkey.pub

Tùy chọn "PublicHostKeyFile" chỉ rõ một tên tập tin chứa đựng khoá công cộng (public host key). Mặc nhiên là "/etc/ssh2/hostkey.pub".

RandomSeedFile random_seed

Tùy chọn "RandomSeedFile" chỉ rõ một tên luân phiên cho tập tin khởi đầu ngẫu nhiên (random seed file) của người sử dụng.

ForwardAgent no

Tùy chọn "ForwardAgent" chỉ rõ tác nhân tính xác thực của sự kết nối nào (nếu có) sẽ được chuyển tới máy ở xa.

ForwardX11 no

Tùy chọn "ForwardX11" dùng cho những người mà sử dụng giao diện Xwindow (Xwindow GUI) và muốn tự động gởi một lần nữa (redirect) các X11 session tới máy ở xa. Vì chúng ta cài đặt một server và không có cài đặt giao diện người dùng (GUI) lên nó, chúng ta có thể an toàn tắt tùy chọn này.

PasswordGuesses 3

Tùy chọn "PasswordGuesses" chỉ rõ có bao nhiêu lần thử mật khẩu mà người dùng có khi sử dụng password authentication.

MaxConnections 5

Tùy chọn "MaxConnections" chỉ rõ rằng số lượng kết nối tối đa mà ssh2 daemon sẽ xử lý xảy ra đồng thời.

PermitRootLogin no

Tùy chọn "PermitRootLogin" chỉ rõ có hay không root có thể log in dùng ssh. Không bao giờ đặt "yes" cho tùy chọn này.

AllowedAuthentications publickey,password

Tùy chọn "AllowedAuthentications" chỉ rõ các phương thức kiểm tra tính xác thực (authentication) nào là được phép được dùng. Với tùy chọn này thì người quản trị hệ thống có thể buộc các người sử dụng phải hoàn thành nhiều authentication trước khi chúng được quan tâm đến.

RequiredAuthentications publickey,password

Tùy chọn "RequiredAuthentications" liên quan với "AllowedAuthentication", chỉ rõ các phương pháp authentication nào mà người sử dụng phải hoàn thành trước khi tiếp tục. Tham số này phải giống như tùy chọn "AllowedAuthentication" hay mọi thời gian server sẽ từ chối kết nối.

VerboseMode no

Tùy chọn "VerboseMode" chỉ thị cho ssh2 in ra các thông điệp debug về sự tiến trình của nó. Tùy chọn này là giúp ích trong việc sửa lỗi sự kết nối, tính xác thực và các vấn đề cấu hình.

PrintMotd yes

Tùy chọn "PrintMotd" chỉ rõ hoặc ssh daemon nên in ra nội dung của tập tin "/etc/motd" khi có một cách tương tác của người dùng. Tập tin "/etc/motd" thì cũng được biết đến như "thông điệp trong ngày".

CheckMail yes

Tùy chọn "CheckMail" chỉ rõ có hay không trình daemon ssh2 sẽ in ra thông tin về mail mới mà bạn có thể có.

UserConfigDirectory "%D/.ssh2"

Tùy chọn "UserConfigDirectory" chỉ rõ vị trí mặc định cho dữ liệu cấu hình của một người sử dụng nào đó.

SyslogFacility DAEMON

Tùy chọn "SyslogFacility" chỉ rõ mã điều kiện được dùng khi ghi lại thông điệp từ sshd. Điều kiện chỉ rõ hệ thống con nào mà nó sinh ra thông điệp – trong trường hợp của chúng ta, DAEMON.

Ssh1Compatibility no

Tùy chọn "Ssh1Compatibility" chỉ rõ có hoặc không dùng mã tương thích SSH1 với SSH2 cho những người sử dụng ssh1.

NoDeplay yes

Tùy chọn "NoDeplay" chỉ rõ nếu tùy chọn socket TCP_NODEPLAY được hỗ trợ. Có một điều khuyên là bạn nên đặt tùy chọn này là "yes" để cải tiến sự thực thi network.

KeepAlive yes

Tùy chọn "KeepAlive" chỉ rõ có hay không hệ thống sẽ gửi các thông điệp giữ kết nối (keep alive) tới máy server ở xa. Nếu đặt là "yes" thì khi đó việc mất kết nối tới các máy ở xa sẽ được thông báo.

UserKnownHosts yes

Tùy chọn "UserKnownHosts" chỉ rõ có hay không thư mục home mặc định của người sử dụng là "$HOME/.ssh2/knownhosts/" có thể được dùng để lấy các public key của host khi dùng "hostbased" authentication.

AllowHosts 192.168.1.4

Tùy chọn "AllowHosts" chỉ rõ và điều khiển các host nào mà có thể truy cập các dịch vụ ssh2. Nhiều host có thể được chỉ định riêng biệt bởi các khoảng trống.

DenyHosts *

Tùy chọn "DenyHosts" chỉ rõ và điều khiển các host nào mà không thể được truy cập các dịch vụ ssh2. Nhiều host có thể được chỉ định riêng biệt bởi các khoảng trống. Kiểu mẫu mặc định "*" có nghĩa là tất cả các host.

QuietMode no

Tùy chọn "QuietMode" chỉ rõ có hay không hệ thống chạy trong chế độ yên lặng (quiet mode). Tùy chọn này phải được đặt là "no" bởi vì trong quiet mode này, không có gì được ghi lại trong hệ thống log ngoại trừ các lỗi nặng. Vì chúng ta cần

phải có thông tin về các phiên (session) của người dùng thì vô hiệu hoá tùy chọn này có thể được ưa chuộng hơn.

Cấu hình sshd2 để dùng TCP-Wrappers inetd super server

TCP-Wrappers nên được hỗ trợ để khởi động và dừng máy server ssh2 của chúng ta. Vào lúc thi hành, inetd đọc thông tin cấu hình của nó từ một tập tin cấu hình mặc nhiên là "/etc/inetd.conf". Có một mục (entry) cho mỗi trường (field) của tập tin cấu hình, với các entry cho mỗi trường được tách riêng biệt bởi một tab và một khoảng trống (space).

Bước 1

Soạn thảo tập tin **inetd.conf** (vi /etc/inetd.conf) và thêm vào các dòng sau:

```
ssh stream  tcp  nowait  root  /usr/sbin/tcpd  sshd -i
```

Ghi chú: *Tham số "-i" là quan trọng khi nó chỉ rõ là sshd được chạy từ inetd. Cũng giống như vậy, cập nhật tập tin "inetd.conf" của bạn bằng cách gửi một tín hiệu SIGHUP (killall -HUP inetd) sau dòng trên được thêm vào tập tin.*

- Để cập nhật tập tin "inetd.conf", dùng lệnh sau:

```
[root@deep /]# killall -HUP inetd
```

Bước 2

Soạn thảo tập tin hosts.allow (vi /etc/hosts.allow) và thêm vào dòng sau:

```
sshd: 192.168.1.4 win.openna.com
```

Có nghĩa là một địa chỉ IP của máy client "192.168.1.4" với tên host name là "win.openna.com" thì được cho phép ssh trên máy server.

Những dịch vụ daemon (cho tcp-wappers) được sử dụng bởi sshd2:

sshd, sshd2	(Tên sshd2 thường được gọi với tên "sshd").
sshdfwd-X11	(Nếu bạn muốn cho phép/từ chối X11-forwarding).
sshdfwd-<port-number>	(cho tcp-forwarding).
sshdfwd-<port-name>	(port-name được định nghĩa trong /etc/services. được sử dụng trong tcp-forwarding).

Ghi chú: *Nếu bạn quyết định dùng ssh, hãy bảo đảm rằng bạn phải cài đặt và dùng nó trên tất cả máy server của bạn. Và bạn cũng nên nhớ rằng nếu như bạn có mười máy server bảo mật và chỉ có một máy server không bảo mật thì đó là một việc làm lãng phí thời gian.*

Cấu hình tập tin "/etc/pam.d/ssh"

Để bảo mật tốt hơn cho server ssh2, thì bạn có thể cấu hình nó để dùng pam authentication. Thực hiện điều đó, bạn phải tạo tập tin "/etc/pam.d/ssh".

Tạo tập tin ssh (touch /etc/pam.d/ssh) và thêm vào hay thay đổi, nếu cần thiết:

```
#%PAM-1.0
auth        required    /lib/security/pam_pwdb.so shadow
auth        required    /lib/security/pam_nologin.so
account     required    /lib/security/pam_pwdb.so
password    required    /lib/security/pam_cracklib.so
password    required    /lib/security/pam_pwdb.so use_authtok nullok md5 shadow
session     required    /lib/security/pam_pwdb.so
```

Tài liệu bổ sung

Muốn biết thêm chi tiết, bạn có thể tham khảo thêm nhiều trang man sau:

```
$ man ssh-add2 (1) - adds identities for the authentication agent
$ man ssh-agent2 (1) - authentication agent
$ man ssh-keygen2 (1) - authentication key pair generation
$ man ssh2 (1) - secure shell client (remote login program)
$ man sshd2 (8) - secure shell daemon
```

Cấu hình SSH2 cho mỗi người sử dụng

Bước 1

Tạo các khóa cục bộ private & public của bạn, bằng cách thi hành:

```
[root@deep /]# su admin
[admin@deep /]$ ssh-keygen2
Generating 1024-bit dsa key pair
6 Oo..oOo.oOo.
Key generated.
1024-bit dsa, admin@deep.openna.com, Sun Feb 13 2000 05:33:38 -0500
Passphrase :
Again :
Private key saved to /home/admin/.ssh2/id_dsa_1024_a
Public key saved to /home/admin/.ssh2/id_dsa_1024_a.pub
```

Bước 2

Tạo một tập tin "identification" trong thư mục home trên máy cục bộ:

```
[admin@deep]$ cd ~/.ssh2
[admin@deep .ssh2]$ echo "IdKey id_dsa_1024_a" > identification
```

Ghi chú: *Nó không bắt buộc để tạo một tập tin identification trên máy ở xa. Tập tin identification chứa đựng tên của private key mà nó được dùng trong authentification.*

Bước 3

Sao chép các public key cục bộ của bạn (**id_dsa_1024_a.pub**) tới thư mục home "~/ssh2" trên máy ở xa với tên được gọi là "**Local.pub**".

Bước 4

Tạo một tập tin "**authorizartion**" trong thư mục home trên máy ở xa:

```
[admin@remote .ssh2]$ touch authorization
```

Chú ý: *"~/" có ý nghĩa là home directory của người sử dụng*

Bước 5

Thêm một dòng sau vào tập tin "authorization" trên host ở xa:

```
[admin@remote .ssh2]$ vi authorization
key             Local.pub
```

Các công cụ SSH2 cho người sử dụng

Các lệnh được liệt kê bên dưới là một số lệnh mà chúng ta thường dùng, nhưng hiện nay nó đã được bổ sung thêm nhiều lệnh hơn, và bạn nên kiểm tra trang man và tài liệu để biết thêm chi tiết.

Ssh2

Lệnh "ssh2" (Secure Shell) cung cấp dịch vụ bảo mật được mã hóa việc trao đổi thông tin giữa hai host không được tin cậy thông qua một mạng không bảo mật. Đó là một chương trình dùng cho việc log in được bảo mật vào trong một máy ở xa và thi hành một số lệnh từ đó. Đó là một sự thay thế phù hợp cho các chương trình không bảo mật như telnet, rlogin, rcp, rdist và rsh.

- Để login vào một máy ở xa, dùng lệnh:

```
[root@deep /]# ssh2 -l <login_name> <hostname>
```

Lấy ví dụ:

```
[root@deep /]# ssh2 -l admin www.openna.com
Passphrase for key "/home/admin/.ssh2/id_dsa_1024_a" with comment
"1024-bit dsa,
```

admin@deep.openna.com, Tue Oct 19 1999 14:31:40 -0400":

admin's password:

Last login: Tue Oct 19 1999 18:13:00 -0400 from deep.openna.com

No mail.

[admin@www admin]$

Trong đó <login_name> là tên mà bạn dùng để kết nối tới ssh2 server ở xa và <hostname> là tên của ssh2 server của bạn ở xa.

sftp2

Tiện ích "sftp2" (Secure File Tranfer) là một client giống như ftp mà nó cung cấp việc truyền tập tin thông qua mạng. Bạn phải được kết nối với ssh2 trước khi dùng tiện ích sftp2.

- Để ftp trên ssh2, dùng lệnh sau:

[admin@deep /]$ **sftp2 <hostname>**

Cho ví dụ:

[admin@deep /]$ **sftp2 www.openna.com**

local path : /home/admin

Passphrase for key "/home/admin/.ssh2/id_dsa_1024_a" with comment "1024-bit dsa,

admin@deep.openna.com, Tue Oct 19 1999 14:31:40 -0400":

admin's password:

admin's password:

remote path : /home/admin

sftp>

Trong đó <hostname> là tên của máy server ở xa mà bạn muốn sftp.

Các tập tin được cài đặt

> /etc/pam.d/ssh
> /etc/ssh2
> /etc/ssh2/hostkey
> /etc/ssh2/hostkey.pub
> /etc/ssh2/sshd2_config
> /etc/ssh2/ssh2_config
> /root/.ssh2
> /usr/man/man8/sshd2.8
> /usr/man/man8/sshd.8
> /usr/bin/ssh2
> /usr/bin/scp2
> /usr/bin/sftp2
> /usr/bin/sftp-server2
> /usr/bin/ssh-agent2

> /root/.ssh2/random_seed
> /root/ssh2
> /usr/man/man1/ssh2.1
> /usr/man/man1/ssh-keygen2.1
> /usr/man/man1/ssh-add2.1
> /usr/man/man1/ssh-agent2.1
> /usr/man/man1/scp2.1
> /usr/man/man1/sftp2.1
> /usr/man/man1/ssh.1
> /usr/man/man1/ssh-add.1
> /usr/man/man1/ssh-agent.1
> /usr/man/man1/ssh-keygen.1
> /usr/man/man1/scp.1
> /usr/man/man1/sftp.1
> /usr/bin/ssh-keygen2
> /usr/bin/ssh-signer2
> /usr/bin/ssh-add2
> /usr/bin/ssh
> /usr/bin/ssh-agent
> /usr/bin/ssh-add
> /usr/bin/ssh-askpass
> /usr/bin/ssh-keygen
> /usr/bin/scp
> /usr/bin/sftp
> /usr/bin/sftp-server
> /usr/bin/ssh-signer
> /usr/sbin/sshd2
> /usr/sbin/sshd

Chương 12:

PHẦN MỀM BẢO MẬT
(Tính toàn vẹn hệ thống)

Trong chương này chúng ta sẽ đề cập đến các phần mềm sau:

- Linux Tripwire 2.2.1.
  - Cấu hình.
  - Bảo vệ Linux với Tripwire.
  - Các lệnh.
- Linux Tripwire ASR 1.3.1.
  - Cấu hình.
  - Sự An toàn của Tripwire.
  - Các lệnh.



Linux Tripwire 2.2.1

Giới thiệu tổng quan

Một tiến trình cài đặt Red Hat Linux Server tiêu biểu xử lý khoảng 30.400 tập tin. Vào thời điểm bận rộn nhất của chúng, các nhà quản trị hệ thống không thể kiểm tra tính toàn vẹn của tất cả các tập tin, và nếu như một kẻ tấn công nào đó truy cập máy server của bạn, thì anh ta hay cô ta có thể cài đặt hay hiệu chỉnh các tập tin mà bạn không dễ nhận biết việc này. Do khả năng của sự cố trên mà một số các chương trình đã được tạo để đáp ứng cho loại vấn đề này.

Y theo site chính thức của Tripwire:

Tripwire làm việc ở tầng cơ bản nhất, bảo vệ các máy server và các máy trạm làm việc (workstation) mà chúng được cấu tạo thành mạng hợp nhất. Tripwire làm việc bằng cách trước tiên là quét một máy tính và tạo một cơ sở dữ liệu của các tập tin hệ thống, một dạng số hóa "snapshot" của hệ thống trong một trạng thái bảo mật đã biết. Người sử dụng có thể cấu hình Tripwire một cách rất chính xác, chỉ rõ các tập tin và thư mục sở hữu riêng trên mỗi máy để theo dõi, hay tạo một dạng mẫu chuẩn mà nó có thể được sử dụng trên tất cả các máy trong mạng.

Một khi cơ sở dữ liệu này được tạo, một người quản trị hệ thống có thể dùng Tripwire để kiểm tra toàn vẹn của hệ thống ở bất kỳ thời điểm nào. Bằng cách quét một hệ thống hiện hành và so sánh thông tin với dữ liệu được lưu trữ trong cơ sở dữ liệu, Tripwire phát hiện và báo cáo bất kỳ việc thêm vào, xóa bớt, hay thay đổi tới hệ thống bên ngoài các ranh giới được chỉ định. Nếu việc thay đổi là hợp lệ, thì người quản trị hệ thống có thể cập nhật cơ sở dữ liệu biên với thông tin mới. Nếu các thay đổi cố tình làm hại được tìm thấy, thì người quản trị hệ thống sẽ biết ngay các phần nào của các thành phần của mạng đã bị ảnh hưởng.

Phiên bản Tripwire này là một sản phẩm có các phần được cải tiến đáng kể so với phiên bản Tripwire trước đó.
Một vài điểm cải tiến bao gồm:

- Nhiều mức độ của báo cáo cho phép bạn chọn các mức độ khác nhau của phần chi tiết báo cáo.
- Tùy chọn Syslog gửi thông tin về sự khởi tạo cơ sở dữ liệu, cập nhật cơ sở dữ liệu và sự kiểm tra toàn vẹn syslog.
- Sự thực thi của cơ sở dữ liệu đã được tối ưu hoá để tăng hiệu quả kiểm tra toàn vẹn.
- Những người nhận email riêng lẻ có thể được gửi các phần nào đó của một báo cáo.
- Hỗ trợ email báo cáo SMTP.
- Chế độ thử nghiệm email (Email test mode) cho phép bạn kiểm tra xem các thông số cài đặt email là đúng.

- Có khả năng tạo nhiều phần bên trong một tập tin chính sách để được thi hành một cách riêng biệt.

Tóm tắt các hướng dẫn cài đặt

Các dòng lệnh là tương thích với Unix.

Đường dẫn nguồn là "/var/tmp" (các đường dẫn khác cũng có thể).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Phiên bản Tripwire là 2.2.1

Các gói cần thiết (Packages)

Tripwire Homepage Site: http://www.psionic.com/abacus/logcheck/

Bạn phải bảo đảm là tải về gói : Tripwire_*221*_for_Linux_x86.tar.gz

Sự biên dịch Trip-2.2.1

Giải nén dạng lưu trữ tarball (tar.gz).

```
[root@deep /]# cp Tripwire_version_for_Linux_x86.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf Tripwire_version_for_Linux_x86.tar.gz
```

Ghi chú: *Sau khi giải nén Tripwire bạn sẽ thấy các tập tin sau đây trong thư mục "/var/tmp" của bạn liên quan tới phần mềm Tripwire: License.txt, README, Release_Notes, install.cfg, install.sh, thư mục package và tập tin Tripwire_version_for_Linux_x86.tar.gz.*

Cấu hình cho tập tin "/var/tmp/install.cfg"

Nhắc lại là do phiên bản Tripwire 2.2.1 không phải là một phiên bản có mã nguồn mở (open source), nên bạn không thể biên dịch và cài đặt nó giống các tập tin nguồn lưu trữ khác; thay vào đó bạn phải hiệu chỉnh tập tin "install.cfg" của Tripwire (tập tin sẽ giúp bạn cài đặt một cách tự động phần mềm Tripwire) để chỉ rõ các đường dẫn cài đặt cho hệ thống của bạn. Chúng ta phải hiệu chỉnh tập tin này để tương thích với cấu trúc hệ thống tập tin của Red Hat và cài đặt các dạng nhị phân Tripwire theo biến môi trường PATH của chúng ta.

Bước 1

Hiệu chỉnh tập tin install.cfg (vi install.cfg) và thay đổi tập tin này để giống như sau:

```
#
```

```
# install.cfg
#
# default install.cfg for:
# Tripwire(R) 2.2.1 for Unix
#
# NOTE: This is a Bourne shell script that stores installation
# parameters for your installation. The installer will
# execute this taäp tin to generate your config taäp tin and also to
# locate any special configuration needs for your install.
# Protect this file, because it is possible for
# malicious code to be inserted here
#
# To set your Root directory for install, set TWROOT= to something
# other than /usr/TSS as necessary.
#
#=============================================================

# If CLOBBER is true, then existing files are overwritten.
# If CLOBBER is false, existing files are not overwritten.
CLOBBER=false

# The root of the TSS directory tree.
TWROOT="/usr"

# Tripwire binaries are stored in TWBIN.
TWBIN="${TWROOT}/bin"

# Tripwire policy files are stored in TWPOLICY.
TWPOLICY="${TWROOT}/TSS/policy"

# Tripwire manual pages are stored in TWMAN.
TWMAN="${TWROOT}/man"

# Tripwire database files are stored in TWDB.
TWDB="${TWROOT}/TSS/db"

# The Tripwire site key files are stored in TWSITEKEYDIR.
TWSITEKEYDIR="${TWROOT}/TSS/key"

# The Tripwire local key files are stored in TWLOCALKEYDIR.
TWLOCALKEYDIR="${TWROOT}/TSS/key"
```

```
# Tripwire report files are stored in TWREPORT.
TWREPORT="${TWROOT}/TSS/report"

# This sets the default text editor for Tripwire.
TWEDITOR="/bin/vi"

# TWLATEPROMTING controls the point when tripwire asks for a password.
TWLATEPROMPTING=false

# TWLOOSEDIRCHK selects whether the directory should be monitored for
# properties that change when files in the directory are monitored.
TWLOOSEDIRCHK=false

# TWMAILNOVIOLATIONS determines whether Tripwire sends a no violation
# report when integrity check is run with --email-report but no rule
# violations are found. This lets the admin know that the integrity
# was run, as opposed to having failed for some reason.
TWMAILNOVIOLATIONS=true

# TWEMAILREPORTLEVEL determines the verbosity of e-mail reports.
TWEMAILREPORTLEVEL=3

# TWREPORTLEVEL determines the verbosity of report printouts.
TWREPORTLEVEL=3

# TWSYSLOG determines whether Tripwire will log events to the system log
TWSYSLOG=false

#####################################
# Mail Options - Choose the appropriate
# method and comment the other section
#####################################

#####################################
# SENDMAIL options - DEFAULT
#
# Either SENDMAIL or SMTP can be used to send reports via
# WMAILMETHOD.
# Specifies which sendmail program to use.
#####################################
```

```
TWMAILMETHOD=SENDMAIL
TWMAILPROGRAM="/usr/lib/sendmail -oi -t"

###################################
# SMTP options
#
# TWSMTPHOST selects the SMTP host to be used to send reports.
# SMTPPORT selects the SMTP port for the SMTP mail program to use.
###################################

# TWMAILMETHOD=SMTP
# TWSMTPHOST="mail.domain.com"
# TWSMTPPORT=25

###########################################################
# Copyright (C) 1998-2000 Tripwire (R) Security Systems, Inc. Tripwire (R) is
a
# registered trademark of the Purdue Research Foundation and is licensed
# exclusively to Tripwire (R) Security Systems, Inc.
###########################################################
```

Ghi chú: *Tập tin "install.cfg" là một script Bourne shell thường dùng bởi các trình cài đặt để đặt các biến cấu hình. Các biến này chỉ rõ các thư mục đích nơi mà các trình cài đặt sẽ sao chép các tập tin và các trình cài đặt nào sẽ thực hiện nếu quá trình cài đặt có thể chép chồng các tập tin phần mềm Tripwire đã tồn tại.*

Bước 2:

Bây giờ chúng ta phải chạy script cài đặt để cài đặt các tập tin nhị phân Tripwire và các tập tin có liên quan trên hệ thống của chúng ta tùy theo hoặc chúng ta sử dụng các giá trị cấu hình mặc định hoặc những giá trị cấu hình theo ý muốn.

- Để chạy các script cài đặt và cài đặt Tripwire, dùng lệnh sau:

```
[root@deep tmp]# ./install.sh
```

Ghi chú: *Tập tin "install.sh" là script cài đặt mà bạn chạy để bắt đầu của việc cài đặt Tripwire. Trong suốt thủ tục cài đặt, bạn sẽ:*

1. Trả lời một số các câu hỏi liên quan tới việc cài đặt.
2. Chỉ rõ hai mật khẩu (pass phrase) để được gán cho site của bạn và các khoá cục bộ (local key).

Bước 3

Khi Tripwire được cài đặt trong hệ thống của bạn nó sẽ sao chép các tập tin "License.txt", "README", và "Release_Notes" vào thư mục "/usr". Dĩ nhiên sau khi kết thúc việc đọc các tập tin đó bạn có thể xóa chúng một cách an toàn từ thư mục "/usr" với lệnh sau:

- Để xoá các tập tin này từ hệ thống của bạn, dùng lệnh sau:

```
[root@deep /usr]# cd /usr
[root@deep /usr]# rm -f License.txt README Release_Notes
```

Thu dọn sau khi làm việc:

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf License.txt README Release-Notes install.cfg
install.sh pkg/Tripwire_version_for_Linux_x86.tar.gz
```

Lệnh "**rm**" khi được dùng ở trên sẽ xóa tất cả các tập tin và các thư mục có liên quan mà chúng ta thường dùng cài đặt Tripwire cho Linux. Nó cũng sẽ xóa Tripwire cho Linux lưu trữ dạng nén từ thư mục "/var/tmp".

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy Tripwire cho Linux, các tập tin sau là được yêu cầu và phải được tạo hay sao chép tới thư mục phù hợp trên máy server của bạn.

Sao chép tập tin **twpol.txt** tới thư mục "/usr/TSS/policy" .

Cấu hình cho tập tin "/usr/TSS/policy/twpol.txt"

Tập tin "/usr/TSS/policy/twpol.txt" là một tập tin dạng văn bản của Tripwire ở đây bạn có thể chỉ rõ các tập tin và các thư mục nào sẽ cần kiểm tra. Ghi chú rằng bao quát việc kiểm tra và kinh nghiệm là cần thiết khi soạn thảo tập tin chính sách này trước khi bạn nhận một tập tin báo cáo đang làm việc. Sau đây là một ví dụ đang làm việc từ nơi mà bạn có thể bắt đầu sự tùy biến của riêng bạn.

Bước 1

Bạn phải hiệu chỉnh tập tin chính sách mặc nhiên, hay tạo riêng của bạn. Tập tin "policyguide.txt" trong thư mục "/usr/TSS/policy" có thể giúp bạn. Mở tập tin "twpol.txt" với một trình soạn thảo văn bản (vi /usr/TSS/policy/twpol.txt) và thay đổi nó để thích hợp với các yêu cầu của bạn:

```
@@section GLOBAL
TWROOT="/usr";
TWBIN="/usr/bin";
TWPOL="/usr/TSS/policy";
TWDB="/usr/TSS/db";
```

```
TWSKEY="/usr/TSS/key";
TWLKEY="/usr/TSS/key";
TWREPORT="/usr/TSS/report";
HOSTNAME=deep.openna.com;

@@section FS
SEC_CRIT = $(IgnoreNone)-SHa; # Critical files - we can't afford to miss any
changes.
SEC_SUID = $(IgnoreNone)-SHa; # Binaries with the SUID or SGID flags set.
SEC_TCB = $(ReadOnly);        # Members of the Trusted Computing Base.
SEC_BIN = $(ReadOnly);        # Binaries that shouldn't change
SEC_CONFIG = $(Dynamic);      # Config files that are changed infrequently but
                              #accessed often.
SEC_LOG = $(Growing);         # Files that grow, but that should never change
                              #ownership.
SEC_INVARIANT = +pug;         # Directories that should never change
                              permission #or ownership.
SIG_LOW = 33;                 # Non-critical files that are of minimal security
impact
SIG_MED = 66;                 # Non-critical files that are of significant
                              #security impact
SIG_HI = 100;                 # Critical files that are significant  points of
                              #vulnerability
# Tripwire Binaries
(emailto = admin@openna.com, rulename = "Tripwire Binaries", severity =
$(SIG_HI))
{
   $(TWBIN)/siggen -> $(ReadOnly);
   $(TWBIN)/tripwire -> $(ReadOnly);
   $(TWBIN)/twadmin -> $(ReadOnly);
   $(TWBIN)/twprint -> $(ReadOnly);
}

# Tripwire Data Files - Configuration Files, Policy Files, Keys, Reports, Databases
(emailto = admin@openna.com, rulename = "Tripwire Data Files", severity =
$(SIG_HI))
{
  # NOTE: Removing the inode attribute because when Tripwire creates a backup
  # it does so by renaming the old file and creating a new one (which will
  # have a new inode number). Leaving inode turned on for keys, which shouldn't
  # ever change.

  # NOTE: this rule will trigger on the first integrity check after database
  # initialization, and each integrity check afterward until a database update
```

```
  # is run, since the database file will not exist before that point.
  $(TWDB)                                -> $(Dynamic) -i;
  $(TWPOL)/tw.pol                        -> $(SEC_BIN) -i;
  $(TWBIN)/tw.cfg                        -> $(SEC_BIN) -i;
  $(TWLKEY)/$(HOSTNAME)-local.key        -> $(SEC_BIN) ;
  $(TWSKEY)/site.key                     -> $(SEC_BIN) ;

  #don't scan the individual reports
  $(TWREPORT)                            -> $(Dynamic) (recurse=0);
}

# These files are critical to a correct system boot.
(emailto = admin@openna.com, rulename = "Critical system boot files", severity =
100)
{
  /boot                                  -> $(SEC_CRIT) ;
  !/boot/System.map                             ;
  !/boot/module-info                            ;
}

# These files change the behavior of the root account
(emailto = admin@openna.com, rulename = "Root config files", severity = 100)
{
  /root                                  -> $(SEC_CRIT) ;
  /root/.bash_history                    -> $(SEC_LOG) ;
}

# Commonly accessed directories that should remain static with regards to owner
and #group
(emailto = admin@openna.com, rulename = "Invariant Directories", severity =
$(SIG_MED))
{
  /                              -> $(SEC_INVARIANT) (recurse = 0);
  /home                          -> $(SEC_INVARIANT) (recurse = 0);
  /etc                           -> $(SEC_INVARIANT) (recurse = 0);
  /chroot                        -> $(SEC_INVARIANT) (recurse = 0);
  /cache                         -> $(SEC_INVARIANT) (recurse = 0);
}

(emailto = admin@openna.com, rulename = "Shell Binaries")
{
  /bin/bsh                       -> $(SEC_BIN);
  /bin/csh                       -> $(SEC_BIN);
```

```
  /bin/sh                          -> $(SEC_BIN);
}

# Rest of critical system binaries
(emailto = admin@openna.com, rulename = "OS executables and libraries",
severity = $(SIG_HI))
{
  /bin                             -> $(ReadOnly) ;
  /lib                             -> $(ReadOnly) ;
}

# Local files
(emailto = admin@openna.com, rulename = "User binaries", severity =
$(SIG_MED))
{
  /sbin                            -> $(SEC_BIN) (recurse = 1);
  /usr/sbin                        -> $(SEC_BIN) (recurse = 1);
  /usr/bin                         -> $(SEC_BIN) (recurse = 1);
}

# Temporary directories
(emailto = admin@openna.com, rulename = "Temporary directories", recurse =
false, severity =$(SIG_LOW))
{
  /usr/tmp                         -> $(SEC_INVARIANT);
  /var/tmp                         -> $(SEC_INVARIANT);
  /tmp                             -> $(SEC_INVARIANT);
}

# Libraries
(emailto = admin@openna.com, rulename = "Libraries", severity = $(SIG_MED))
{
  /usr/lib                         -> $(SEC_BIN);
}

# Include
(emailto = admin@openna.com, rulename = "OS Development Files", severity =
$(SIG_MED))
{
  /usr/include                     -> $(SEC_BIN);
}

# Shared
```

```
(emailto = admin@openna.com, rulename = "OS Shared Files", severity =
$(SIG_MED))
{
  /usr/share                    -> $(SEC_BIN);
}

# Kernel headers files
(emailto = admin@openna.com, rulename = "Kernel Headers Files", severity = $(
SIG_HI))
{
  /usr/src/linux-2.2.14         -> $(SEC_BIN);
}

# setuid/setgid root programs
(emailto = admin@openna.com, rulename = "setuid/setgid", severity = $(SIG_HI))
{
  /bin/su                       -> $(SEC_SUID);
  /sbin/pwdb_chkpwd             -> $(SEC_SUID);
  /sbin/dump                    -> $(SEC_SUID);
  /sbin/restore                 -> $(SEC_SUID);
  /usr/bin/at                   -> $(SEC_SUID);
  /usr/bin/passwd               -> $(SEC_SUID);
  /usr/bin/suidperl             -> $(SEC_SUID);
  /usr/bin/crontab              -> $(SEC_SUID);
  /usr/sbin/sendmail            -> $(SEC_SUID);
  /usr/bin/man                  -> $(SEC_SUID);
  /usr/bin/sperl5.00503         -> $(SEC_SUID);
  /usr/bin/slocate              -> $(SEC_SUID);
  /usr/sbin/utempter            -> $(SEC_SUID);
  /sbin/netreport               -> $(SEC_SUID);
}

(emailto = admin@openna.com, rulename = "Configuration Files")
{
  /etc/hosts                    -> $(SEC_CONFIG);
  /etc/inetd.conf               -> $(SEC_CONFIG);
  /etc/initlog.conf             -> $(SEC_CONFIG);
  /etc/inittab                  -> $(SEC_CONFIG);
  /etc/resolv.conf              -> $(SEC_CONFIG);
  /etc/syslog.conf              -> $(SEC_CONFIG);
}

(emailto = admin@openna.com, rulename = "Security Control")
```

```
{
  /etc/group                    -> $(SEC_CRIT);
  /etc/security/                -> $(SEC_CRIT);
  /lib/security/                -> $(SEC_CRIT);
  /var/spool/cron               -> $(SEC_CRIT);
}

(emailto = admin@openna.com, rulename = "Login Scripts")
{
  /etc/csh.login                -> $(SEC_CONFIG);
  /etc/profile                  -> $(SEC_CONFIG);
}

# These files change every time the system boots
(emailto = admin@openna.com, rulename = "System boot changes", severity =
$(SIG_HI))
{
  /dev/log                      -> $(Dynamic) ;
  /dev/cua0                     -> $(Dynamic) ;
  /dev/console                  -> $(Dynamic) ;
  /dev/tty2                     -> $(Dynamic) ; # tty devices
  /dev/tty3                     -> $(Dynamic) ; # are extremely
  /dev/tty4                     -> $(Dynamic) ; # variable
  /dev/tty5                     -> $(Dynamic) ;
  /dev/tty6                     -> $(Dynamic) ;
  /dev/urandom                  -> $(Dynamic) ;
  /dev/initctl                  -> $(Dynamic) ;
  /var/lock/subsys              -> $(Dynamic) ;
  /var/run                      -> $(Dynamic) ; # daemon PIDs
  /var/log                      -> $(Dynamic) ;
  /etc/ioctl.save               -> $(Dynamic) ;
  /etc/.pwd.lock                -> $(Dynamic) ;
  /etc/mtab                     -> $(Dynamic) ;
  /lib/modules                  -> $(Dynamic) ;
}

# Critical configuration files
(emailto = admin@openna.com, rulename = "Critical configuration files", severity
= $(SIG_HI))
{
  /etc/conf.modules             -> $(ReadOnly) ;
  /etc/crontab                  -> $(ReadOnly) ;
  /etc/cron.hourly              -> $(ReadOnly) ;
```

```
  /etc/cron.daily                -> $(ReadOnly) ;
  /etc/cron.weekly               -> $(ReadOnly) ;
  /etc/cron.monthly              -> $(ReadOnly) ;
  /etc/default                   -> $(ReadOnly) ;
  /etc/fstab                     -> $(ReadOnly) ;
  /etc/group                     -> $(ReadOnly) ; # changes should be infrequent
  /etc/host.conf                 -> $(ReadOnly) ;
  /etc/hosts.allow               -> $(ReadOnly) ;
  /etc/hosts.deny                -> $(ReadOnly) ;
  /etc/lilo.conf                 -> $(ReadOnly) ;
  /etc/logrotate.conf            -> $(ReadOnly) ;
  /etc/pwdb.conf                 -> $(ReadOnly) ;
  /etc/securetty                 -> $(ReadOnly) ;
  /etc/sendmail.cf               -> $(ReadOnly) ;
  /etc/protocols                 -> $(ReadOnly) ;
  /etc/services                  -> $(ReadOnly) ;
  /etc/rc.d/init.d               -> $(ReadOnly) ;
  /etc/rc.d                      -> $(ReadOnly) ;
  /etc/motd                      -> $(ReadOnly) ;
  /etc/passwd                    -> $(ReadOnly) ;
  /etc/passwd-                   -> $(ReadOnly) ;
  /etc/profile.d                 -> $(ReadOnly) ;
  /etc/rpc                       -> $(ReadOnly) ;
  /etc/sysconfig                 -> $(ReadOnly) ;
  /etc/shells                    -> $(ReadOnly) ;
  /etc/nsswitch.conf             -> $(ReadOnly) ;
}

# Critical devices
(emailto = admin@openna.com, rulename = "Critical devices", severity =
$(SIG_HI), recurse = false)
{
  /dev/kmem                      -> $(Device) ;
  /dev/mem                       -> $(Device) ;
  /dev/null                      -> $(Device) ;
  /dev/zero                      -> $(Device) ;
  /proc/devices                  -> $(Device) ;
  /proc/net                      -> $(Device) ;
  /proc/tty                      -> $(Device) ;
  /proc/sys                      -> $(Device) ;
  /proc/cpuinfo                  -> $(Device) ;
  /proc/modules                  -> $(Device) ;
  /proc/mounts                   -> $(Device) ;
```

```
    /proc/dma                -> $(Device) ;
    /proc/filesystems        -> $(Device) ;
    /proc/ide                -> $(Device) ;
    /proc/interrupts         -> $(Device) ;
    /proc/ioports            -> $(Device) ;
    /proc/scsi               -> $(Device) ;
    /proc/kcore              -> $(Device) ;
    /proc/self               -> $(Device) ;
    /proc/kmsg               -> $(Device) ;
    /proc/stat               -> $(Device) ;
    /proc/ksyms              -> $(Device) ;
    /proc/loadavg            -> $(Device) ;
    /proc/uptime             -> $(Device) ;
    /proc/locks              -> $(Device) ;
    /proc/version            -> $(Device) ;
    /proc/meminfo            -> $(Device) ;
    /proc/cmdline            -> $(Device) ;
    /proc/misc               -> $(Device) ;
}
```

Ghi chú: *Đây là một ví dụ tập tin chính sách mà chúng tôi cung cấp cho bạn; dĩ nhiên, bạn phải hiệu chỉnh tập tin này để thích hợp với các tập tin hệ thống và các yêu cầu cụ thể nào đó.*

Bước 2

Một khi bạn sẵn sàng để dùng tập tin chính sách của bạn trong lần đầu; cài đặt nó với lệnh sau:

```
[root@deep /]# twadmin --create-polfile /usr/TSS/policy/twpol.txt
Please enter your site passphrase:
Wrote policy file: /usr/TSS/policy/tw.pol
```

Bảo vệ Linux với Tripwire

Vấn đề bảo mật

Điều quan trọng là phải bảo đảm rằng tính toàn vẹn của hệ thống mà bạn đang chạy không bị tổn thương. Để bảo mật tối đa trong cơ sở dữ liệu của bạn, bạn nên sinh ra các tập tin ứng dụng và hệ thống từ một sự cài đặt rõ ràng và môi trường gốc.

Ngoài ra bạn nên xóa bản sao chép văn bản của tập tin cấu hình Tripwire đặt tên là "twcfg.txt" được đặt trong thư mục "/usr/bin" để ẩn vị trí của các tập tin Tripwire và ngăn chặn bất kỳ ai tạo một tập tin thứ hai, hoặc thay đổi tập tin cấu hình.

- Để xóa bản sao chép dạng văn bản của tập tin cấu hình Tripwire, dùng lệnh sau:

```
[root@deep /]# rm -f /usr/bin/twcfg.txt
```

Các tư liệu

Để biết thêm các chi tiết, có rất nhiều trang hướng dẫn (man pages) mà bạn có thể đọc:

$ siggen (8) - Thu nhặt những dấu hiệu thông thường cho Tripwire.

$ tripwire (8) - Kiểm tra tính nguyên vẹn một tập tin cho các hệ thống Unix.

$ twadmin (8) - Công cụ quản trị và tiện ích Tripwire.

$ twconfig (4) - Tham khảo tập tin cấu hình Tripwire.

$ twfiles (5) - Khái quát những tập tin được sử dụng bởi Tripwire và tiến trình sao lưu (backup).

$ twintro (8) - Giới thiệu phần mềm Tripwire.

$ twpolicy (4) - Tham khảo tập tin chính sách Tripwire.

$ twprint (8) - In báo cáo và cơ sở dữ liệu Tripwire.

Các lệnh

Các lệnh được liệt kê bên dưới là một số mà chúng ta thường dùng trong trong công việc hằng ngày, nhưng vẫn còn nhiều lệnh chúng ta cũng cần tham khảo.

Kiểm tra man page để biết thêm chi tiết.

Tạo cơ sở dữ liệu cho lần đầu tiên

Một khi tập tin chính sách của bạn đã được cài đặt, đã đến lúc phải xây dựng và khởi tạo cơ sở dữ liệu của các đối tượng tập tin hệ thống, dựa trên các nguyên tắc từ tập tin chính sách của bạn. Cơ sở dữ liệu này sẽ phục vụ như đường biên cho việc kiểm tra tính toàn vẹn sau này.

Cú pháp cho việc khởi tạo CSDL (Database Initialization mode)là :

```
[root@deep /]# tripwire { --init }
```

- Để khởi tạo tập tin cơ sở dữ liệu, dùng lệnh sau:

```
[root@deep /]# tripwire --init
Please enter your local passphrase:
Parsing policy file: /usr/TSS/policy/tw.pol
Generating the database...
*** Processing Unix file System ***
```

```
Wrote database file: /usr/TSS/db/deep.openna.com.twd
The database was successfully generated.
```

Ghi chú: *Khi lệnh này đã được thi hành, cơ sở dữ liệu sẵn sàng. Bạn có thể kiểm tra tính toàn vẹn và xem lại tập tin báo cáo.*

Chạy kiểm tra tính tương tác hoặc toàn vẹn của các file

Tripwire có một đặc tính được gọi là phương thức kiểm tra tính toàn vẹn "Integrity Check Mode". Bây giờ thì cơ sở dữ liệu của chúng ta đã được xây dựng, chúng ta có thể chạy đặc tính này để so sánh các đối tượng hệ thống tập tin hiện tại với các thuộc tính riêng của nó khi được lưu lại trong cơ sở dữ liệu. Tất cả sự vi phạm của các tập tin sẽ được in ra màn hình; tập tin báo cáo được sinh sẽ được lưu trữ và có thể được truy cập sau bởi tiện ích twprint.

Cú pháp cho chế độ kiểm tra tính toàn vẹn(integrity check mode)

```
[root@deep /]# tripwire { --check }
```

- Để chạy chế độ kiểm tra tính toàn vẹn (integrity check mode), dùng lệnh:

```
[root@deep /]# tripwire --check
```

Tripwire cũng có thể được chạy trong "Integrity Check Mode". Trong mode này bạn có thể cập nhật một cách tự động các thay đổi của bạn thông qua máy trạm (ternminal).

- Để chạy trong chế độ kiểm tra tương tác (interactive check mode), dùng lệnh:

```
[root@deep /]# tripwire --check –interactive
```

Một tùy chọn email sẵn có với Tripwire và cho phép bạn gửi email. Tùy chọn này sẽ chỉ rõ rằng các báo cáo được email tới các người nhận được định sẵn trong tập tin chính sách.

- Để chạy trong chế độ kiểm tra tính toàn vẹn (integrity check mode) và gửi email tới người nhận, dùng lệnh:

```
[root@deep /]# tripwire --check --email-report
```

Cập nhật cơ sở dữ liệu sau khi một sự kiểm tra toàn vẹn

Nếu bạn quyết định dùng "Integrity Check Mode" của Tripwire thay vì là "Interactive Check Mode", bạn phải cập nhật cơ sở dữ liệu Tripwire với đặc tính "Database Update Mode". Quá trình cập nhật này cho phép bạn tiết kiệm thời gian bằng cách cập nhật cơ sở dữ liệu không phải phục hồi lại nó, và nó cũng cho phép khả năng chọn cập nhật, điều mà không thể được thực hiện thông qua sự tái tạo lại.

Cú pháp cho Database update mode là :

```
[root@deep /]# tripwire { --update -r}
```

- Để cập nhật Database (update mode), dùng lệnh:

```
[root@deep /]# tripwire --update -r /usr/TSS/report/deep.openna.com-200001-021854.twr
```

Trong đó "-r" đọc tập tin báo cáo được chỉ rõ (deep.opena.com-200001-021854.twr). Tùy chọn này là được yêu cầu bởi vì biến REPORTFILE trong tập tin cấu hình hiện tại sử dụng $(DATE).

Ghi chú: *Trong Database update mode hay Interactive Check Mode, phần mềm Tripwire hiển thị báo cáo trong máy trạm của bạn với message tương ứng với mỗi vi phạm chính sách. Bạn có thể chấp nhận một thay đổi hệ thống tập tin bằng cách bỏ dấu "x" cạnh bên mỗi vi phạm chính sách hay xóa dấu "x" cạnh bên message, như vậy cơ sở dữ liệu sẽ không cập nhật với các giá trị mới cho đối tượng đó. Sau khi bạn thoát khỏi trình soạn thảo và cung cấp local pass phrase, phần mềm Tripwire sẽ cập nhật và lưu trữ các thay đổi của bạn.*

Cập nhật tập tin chính sách (policy file)

Đôi khi bạn muốn thay đổi các qui tắc trong tập tin chính sách của bạn để tương ứng với các vị trí tập tin hay các nguyên tắc chính sách mới. Một lệnh đặc biệt có sẵn để thực hiện công việc và cập nhật cơ sở dữ liệu không yêu cầu một sự tái tạo đầy đủ của tập tin chính sách. Việc này có thể tiết kiệm một lượng thời gian đáng kể và duy trì bảo mật bằng cách giữ tập tin chính sách được đồng bộ hoá với cơ sở dữ liệu nó dùng.

Cú pháp cho chế độ cập nhật chính sách (policy update mode) là:

```
[root@deep /]# tripwire { --update-policy /path/to/new/policy/file}
```

- Để cập nhật tập tin chính sách, dùng lệnh:

```
[root@deep /]# tripwire --update-policy /usr/TSS/policy/newtwpol.txt
```

Mặc định, Policy Update Mode chạy với tùy chọn "-secure-mode high". Bạn có thể bắt gặp các lỗi khi chạy với tùy chọn này nếu hệ thống tập tin đã bị thay đổi kể từ lần cập nhật cơ sở dữ liệu sau cùng, và nếu thay đổi gây ra một vi phạm trong chính sách mới. Sau khi chỉ rõ rằng tất cả các vi phạm được báo cáo trong chế độ bảo mật cao được cho phép, bạn có thể cập nhật tập tin chính sách trong chế độ bảo mật thấp để giải quyết tình huống này:

- Để cập nhật tập tin chính sách trong chế độ bảo mật thấp, dùng lệnh:

```
[root@deep /]# tripwire --update-policy --secure-mode low \
/usr/TSS/policy/newtwpol.txt
```

Các tập tin cài đặt

> /usr/TSS	> /usr/TSS/key/deep.openna.com-local.key
> /usr/bin	> /usr/man

> /usr/bin/siggen
> /usr/bin/twprint
> /usr/bin/twadmin
> /usr/bin/tripwire
> /usr/bin/twcfg.txt
> /usr/bin/tw.cfg
> /usr/TSS/policy
> /usr/TSS/policy/policyguide.txt
> /usr/TSS/policy/twpol.txt
> /usr/TSS/policy/tw.pol
> /usr/TSS/policy/twpol.txt.bak
> /usr/TSS/report
> /usr/TSS/db
> /usr/TSS/key
> /usr/TSS/key/site.key
> /usr/man/man4
> /usr/man/man4/twconfig.4
> /usr/man/man4/twpolicy.4
> /usr/man/man5
> /usr/man/man5/twfiles.5
> /usr/man/man8
> /usr/man/man8/siggen.8
> /usr/man/man8/tripwire.8
> /usr/man/man8/twadmin.8
> /usr/man/man8/twintro.8
> /usr/man/man8/twprint.8
> /usr/README
> /usr/Release_Notes
> /usr/License.txt

Linux Tripwire ASR 1.3.1

Giới thiệu tổng quan

Tripwire ASR 1.3.1 là phiên bản "Academic Source Release (ASR)" của phần mềm Tripwire. Theo chúng tôi, thì chúng tôi thích phiên bản 1.3.1 của phần mềm này hơn phiên bản 2.2.1 bởi vì nó có thể biên dịch và được cài đặt mà không cần bất kỳ các vấn đề tương thích nào trên tất cả các phiên bản của các hệ thống Linux.

Như được giải thích trong các mục đích của Tripwire ASR:

Với việc dẫn đến độ phức tạp được gia tăng và các cuộc tấn công rất tinh vi vào các hệ thống Unix, việc cần thiết có các công cụ để hỗ trợ trong việc phát hiện những người không có chủ quyền thay đổi hiệu chỉnh các tập tin trở nên cấp thiết. Tripwire là một công cụ hỗ trợ những người quản trị hệ thống và các người sử dụng trong việc theo dõi một tập hợp các tập tin đã được hoạch định cho bất kỳ các thay đổi nào. Được dùng với các tập tin hệ thống dựa trên một nguyên tắc cơ bản (nghĩa là thường ngày), Tripwire có thể thông báo cho những người quản trị hệ thống về các tập tin bị giả mạo hay hư hỏng, khi đó sẽ đo được thiệt hại biết trước có thể xảy ra.

Tripwire là một trình kiểm tra toàn vẹn tập tin và thư mục, một tiện ích so sánh một tập hợp các tập tin và các thư mục được hoạch định, với các thông tin được lưu trữ trong một cơ sở dữ liệu được sinh ra trước đó. Bất kỳ sự khác biệt nào xuất hiện thì cờ hiệu được bật hay được log lại, bao gồm cả việc thêm hay xóa các dữ liệu nhập vào. Khi nó chạy dựa vào các tập tin hệ thống trên một nguyên tắc cơ bản, thì bất kỳ thay đổi nào trong các tập tin hệ thống quan trọng sẽ được đánh dấu – và việc điều khiển hư hại cụ thể nào đó có thể được biết ngay lập tức. Với Tripwire, những người quản trị hệ thống có thể quyết định với một mức độ cao sự tin chắc rằng một tập hợp các tập tin định sẵn vẫn nguyên vẹn tức là không bị hiệu chỉnh nếu Tripwire báo cáo không có sự thay đổi nào.

Tóm tắt các chỉ thị cài đặt này

Các dòng lệnh là tương thích với Unix.

Đường dẫn nguồn là "/var/tmp" (các đường dẫn khác là cũng có thể).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ được thực hiện với người sử dụng có quyền cao nhất trong hệ thống (root).

Số phiên bản Tripwire là 1.3.1-1

Các gói cần thiết (Packages)

Tripwire Homepage Site: http://www.tripwiresecurity.com/

Bạn phải bảo đảm là tải về gói : Tripwire -1.3.1-1.tar.gz

Dạng lưu trữ Tarballs

Để kiểm soát việc cài đặt bạn nên tạo một danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt Logcheck, và sau đó so sánh chúng bởi lệnh

"**diff**" để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh "**find /* > Tripwire1**" trước và "**find /* > Tripwire2**' sau khi bạn cài đặt phần mềm, và chạy lệnh "**diff Tripwire1 Tripwire2 > Tripwire-Installed**" để lấy một danh sách các thay đổi nào đó.

Sự biên dịch Tripwire-1.3.1-1

Giải nén dạng lưu trữ tarball (tar.gz)

```
[root@deep /]# cp Tripwire-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf Tripwire-version.tar.gz
```

Biên dịch và tối ưu

Chuyển vào trong thư mục Tripwire mới và gõ vào như sau trên máy trạm terminal:

Soạn thảo tập tin **utils.c** (vi +462 src/utils.c) và thay đổi dòng:

```
else if (iscntrl(*pcin)) {
```

Chuyển thành:

```
else if (!(*pcin & 0x80) && iscntrl(*pcin)) {
```

Soạn thảo tập tin **congfig.parse.c** (vi +356 src/config.parse.c) và thay đổi dòng:

```
rewind(fpout);
```

Chuyển thành:

```
else {
rewind(fpin);
}
```

Soạn thảo tập tin **congfig.h** (vi +106 include/config.h) và thay đổi dòng:

```
#define CONFIG_PATH "/usr/local/bin/tw"
#define DATABASE_PATH "/var/tripwire"
```

Chuyển thành:

```
#define CONFIG_PATH "/etc"
#define DATABASE_PATH "/var/spool/tripwire"
```

Soạn thảo tập tin **congfig.h** (vi +165 include/config.h) và thay đổi dòng:

```
#define TEMPFILE_TEMPLATE "/tmp/twzXXXXXX"
```

Chuyển thành:

#define TEMPFILE_TEMPLATE "/var/tmp/.twzXXXXXX"

Soạn thảo tập tin **congfig.pre.y** (vi +66 src/config.pre.y) và thay đổi dòng:

```
#ifdef TW_LINUX
```

Trở thành:

```
#ifdef TW_LINUX_UNDEF
```

Soạn thảo tập tin **Makefile** (vi +13 Makefile) và thay đổi dòng:

```
DESTDIR = /usr/local/bin/tw
```

Trở thành:

```
DESTDIR = /usr/sbin

DATADIR = /var/tripwire
```

Trở thành:

```
DATADIR = /var/spool/tripwire
```

```
LEX = lex
```
Trở thành:
```
LEX = flex

CC=gcc
```
Trở thành:
```
CC=egcs

CFLAGS = -O
```
Chuyển thành:
```
CFLAGS = -O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomit-frame-pointer -fno-exceptions

[root@deep tw_ASR_1.3.1_src]# make
[root@deep tw_ASR_1.3.1_src]# make install

[root@deep tw_ASR_1.3.1_src]# chmod 700 /var/spool/tripwire/
[root@deep tw_ASR_1.3.1_src]# chmod 500 /usr/sbin/tripwire
[root@deep tw_ASR_1.3.1_src]# chmod 500 /usr/sbin/siggen
[root@deep tw_ASR_1.3.1_src]# rm -f /usr/sbin/tw.config
```

Các lệnh "**make**" và "**make install**" trên sẽ cấu hình phần mềm để bảo đảm rằng hệ thống của bạn có các thư viện và chức năng cần thiết để biên dịch thành công package, biên dịch tất cả các tập tin nguồn sang các tập tin dạng nhị phân có thể thi hành, và sau đó cài đặt các tập tin dạng nhị phân và bất kỳ các tập tin hỗ trợ vào những nơi thích hợp.

Lệnh "**chmod**" sẽ thay đổi quyền mặc định của thư mục "tripwire" là 700 (drwx------) chỉ có thể đọc, có thể ghi, và có thể thực thi bởi người sử dụng có quyền cao nhất "root". Nó sẽ tạo dạng nhị phân "/usr/sbin/tripwire" chỉ có thể đọc và có thể thi hành bởi "root" (-r-x------) và cuối cùng tạo chương trình "siggen" trong thư mục "/usr/sbin" có thể thi hành và có thể đọc bởi người sử dụng có quyền cao nhất "root".

Lệnh "**rm**" khi được dùng ở trên sẽ xóa tập tin "tw.config" trong thư mục "/usr/sbin". Chúng ta không cần đến tập tin này bởi vì chúng ta sẽ tạo một tập tin mới trong thư mục "/etc" sau đó.

Thu dọn sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf tw_ASR_version/ Tripwire-version.tar.gz
```

Lệnh "**rm**" như được dùng ở trên sẽ xóa tất cả các tập tin nguồn mà chúng ta đã dùng để biên dịch và cài đặt Tripwire. Nó cũng sẽ xóa dạng lưu trữ nén Tripwire từ thư mục "/var/tmp".

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy Tripwire cho Linux, tập tin sau là được yêu cầu và phải được tạo hay sao chép tới thư mục phù hợp trên máy server của bạn.

  Sao chép tập tin **tw.config** tới thư mục "/etc" .

  Sao chép tập tin **tripwire.verify** tới thư mục "/etc/cron.daily".

Bạn có thể lấy các tập tin cấu hình được liệt kê bên dưới trên dạng lưu trữ floppy.tgz được giải nén. Sao chép tập tin sau từ dạng lưu trữ floppy.tgz tới nơi phù hợp, hay sao chép và dán nó trực tiếp từ quyển sách này tới tập tin có liên quan.

Cấu hình cho tập tin "/etc/tw.config"

Tập tin "/etc/tw.config" là một tập tin cấu hình Tripwire nơi mà bạn quyết định đặt các tập tin hệ thống và các thư mục nào mà bạn muốn được theo dõi. Chú ý rằng kinh nghiệm và sự kiểm tra bao quát là cần thiết khi soạn thảo tập tin này trước khi bạn nhận tập tin báo cáo vừa ý cho công việc của bạn. Sau đây là một ví dụ, bạn có thể bắt đầu việc hiệu chỉnh theo ý muốn của riêng bạn.

Bước 1

Tạo tập tin tw.config (touch /etc/tw.config) và thêm vào tập tin này tất cả các tập tin và các thư mục mà bạn muốn theo dõi. Định dạng của tập tin cấu hình là được mô tả trong phần đầu (header) của nó và trong man page tw.config (5).

```
# Gerhard Mourani: gmourani@videotron.ca
# last updated: 1999/11/12

# First, root's "home"
/root                   R
!/root/.bash_history
/                       R

# OS itself
/boot/vmlinuz           R

# critical boot resources
/boot                   R

# Critical directories and files
/chroot                 R
/etc                    R
/etc/inetd.conf         R
/etc/nsswitch.conf      R
/etc/rc.d               R
/etc/mtab               L
/etc/motd               L
/etc/group              R
/etc/passwd             L

# other popular filesystems
/usr                    R
/usr/local              R
/dev                    L-am
```

```
/usr/etc                R

# truncate home
=/home                  R

# var tree
=/var/spool             L
/var/log                L
/var/lib                L
/var/spool/cron         L
!/var/lock

# unusual directories
=/proc                  E
=/tmp
=/mnt/cdrom
=/mnt/floppy
```

Bước 2

Bây giờ, để phục vụ cho các lý do bảo mật, thay đổi mode của tập tin này là 0600 với lệnh sau:

```
[root@deep /]# chmod 600 /etc/tw.config
```

Cấu hình của tập tin script "/etc/cron.daily/tripwire.verify"

Tập tin "tripwire.verify" là một script nhỏ được thi hành bởi chương trình cron của server của bạn hằng ngày để quét đĩa cứng của bạn cho các tập tin, thư mục có thể thay đổi và mail các kết quả tới cho các người quản trị hệ thống. Script này sẽ tự động thi hành thủ tục kiểm tra toàn vẹn cho bạn. Nếu bạn có ý định tự động công việc này, làm theo các bước đơn giản được trình bày bên dưới:

Bước 1

Tạo tập tin script tripwire.verify (touch /etc/cron.daily/tripwire.verify) và thêm vào đoạn script này:

```
#!/bin/sh
/usr/sbin/tripwire -loosedir -q | (cat <<EOF
This is an automated report of possible file integrity changes, generated by
the Tripwire integrity checker. To tell Tripwire that a taäp tin or entire
directory tree is valid, as root run:

/usr/sbin/tripwire -update [pathname|entry]
```

Nếu bạn muốn nhập vào một phiên kiểm tra và kiểm tra tính tương tác và chứng cứ như root chạy :

```
/usr/sbin/tripwire -interactive

Changed files/directories include:
EOF
cat
) | /bin/mail -s "File integrity report" root
```

Bước 2

Bây giờ, tạo script có thể thi hành và thay đổi mode của nó là 0700 với lệnh sau:

```
[root@deep /]# chmod 700 /etc/cron.daily/tripwire.verify
```

Sự an toàn của Tripwire

Vấn đề bảo mật

Để việc bảo mật được tốt hơn, điều cần lưu ý là tập tin cơ sở dữ liệu (tw.db_[hostname]) của Tripwire nên được chuyển tới một nơi nào đó (ví dụ như đĩa mềm) mà nó không thể được hiệu chỉnh. Điều này rất quan trọng bởi vì dữ liệu từ Tripwire chỉ đáng tin cậy như cơ sở dữ liệu của nó.

Bạn nên tạo một bản sao chép được in ra giấy các nội dung của cơ sở dữ liệu. Trong trường hợp bạn nghi ngờ tính toàn vẹn của cơ sở dữ liệu, bạn có thể so sánh thông tin của database dựa vào bản được in ra giấy này.

Các thông tin khác

Để biết thêm các chi tiết, có rất nhiều trang hướng dẫn (man pages) mà bạn có thể đọc:

```
$ man siggen (8)        - signature generation routine for Tripwire
$ man tripwire (8)      - a taäp tin integrity checker for UNIX systems
$ man tw.config (5)     - configuration taäp tin for Tripwire
```

Các lệnh

Các lệnh được liệt kê bên dưới là một số lệnh mà chúng ta thường dùng trong công việc, nhưng còn rất nhiều lệnh khác, xem các trang man cho biết thêm chi tiết.

Chạy Tripwire trong Interactive Checking Mode

Trong đặc trưng "Interactive Check Mode". Tripwire kiểm tra các tập tin hay các thư mục đã được thêm vào, xóa đi, hay được thay đổi từ cơ sở dữ liệu gốc và hỏi người sử dụng trong bất kỳ trường hợp nào mà cơ sở dữ liệu nhập vào nên được cập nhật. Mode này là cách tiện lợi nhất để lưu giữ cơ sở dữ liệu của bạn cập nhật, nhưng nó yêu cầu người sử dụng là "ở console". Nếu bạn có ý định dùng mode này, làm theo các bước đơn giản ở bên dưới.

Bước 1

Tripwire phải có một cơ sở dữ liệu để so sánh lại vì vậy chúng ta trước tiên phải tạo tập tin cơ sở dữ liệu thông tin. Hành động này sẽ tạo một tập tin được gọi là "tw.db_[hostname]" trong thư mục mà bạn chỉ định để lưu giữ các cơ sở dữ liệu cho bạn (trong đó [hostname] sẽ được thay thế với tên máy host của bạn).

- Để tạo tập tin cơ sở dữ liệu thông tin cho Tripwire, dùng lệnh:

```
[root@deep /]# cd /var/spool/tripwire/
```

```
[root@deep tripwire]# /usr/sbin/tripwire --initialize
```

Chúng ta di chuyển thư mục mà chúng ta chỉ định để lưu giữ các cơ sở dữ liệu, và sau đó chúng ta tạo tập tin cơ sở dữ liệu thông tin, tập tin này được sử dụng cho tất cả việc kiểm tra tính toàn vẹn theo sau.

Bước 2

Một khi tập tin cơ sở dữ liệu thông tin của Tripwire đã được tạo, chúng ta bây giờ có thể chạy Tripwire trong "Interactive Checking Mode". Mode này sẽ nhắc người sử dụng trong bất cứ trường hợp nào mỗi thay đổi trên hệ thống đều cần được cập nhật để phản ánh trạng thái hiện hành của tập tin.

- Để chạy trong Interactive Checking Mode, dùng lệnh:

```
[root@deep /]# cd /var/spool/tripwire/database/
[root@deep database]# cp tw.db_myserverhostname /var/spool/tripwire/
[root@deep database]# cd ..
[root@deep tripwire]# /usr/sbin/tripwire --interactive
Tripwire(tm) ASR (Academic Source Release) 1.3.1
File Integrity Assessment Software
(c) 1992, Purdue Research Foundation, (c) 1997, 1999 Tripwire
Security Systems, Inc. All Rights Reserved. Use Restricted to
Authorized Licensees.
### Phase 1:    Reading configuration file
### Phase 2:    Generating file list
### Phase 3:    Creating file information database
### Phase 4:    Searching for inconsistencies
###
###                Total files scanned: 15722
###                Files added: 34
###                Files deleted: 42
###                Files changed: 321
###
###                Total file violations: 397
###
added: -rwx------ root 22706 Dec 31 06:25:02 1999 /root/tmp/firewall
---> File: '/root/tmp/firewall'
---> Update entry? [YN(y)nh?]
```

Ghi chú: *Trong Interactive mode, Tripwire trước tiên báo cáo tất cả việc thêm, xoá, và thay đổi các tập tin sau đó cho phép người sử dụng để cập nhật việc nhập vào cơ sở dữ liệu.*

Chạy Tripwire trong Database Update Mode

Chạy Tripwire trong "Database Update Mode" xen lẫn với tập tin script "tripwire.verify" sau đó mail kết quả tới người quản trị hệ thống, việc này sẽ giảm thời gian quét hệ thống. Thay vì chạy Tripwire trong "Interactive Checking Mode" và mất nhiều thời gian chờ cho việc quét hệ thống tập tin kết thúc, tập tin script "tripwire.verify" sẽ quét hệ thống và báo cáo kết quả thông qua mail, sau đó bạn chạy Tripwire trong "Database Update Mode" và chỉ cập nhật các tập tin hay các thư mục riêng rẽ đã được thay đổi.

Ví dụ:

Nếu một tập tin riêng đã thay đổi, bạn có thể :

```
[root@deep /]# tripwire -update /etc/newly.installed.file
```

Hoặc nếu toàn bộ các tập tin hay thư mục đã thay đổi, bạn có thể chạy:

```
[root@deep /]# tripwire -update /usr/lib/Package_Dir
```

Trong trường hợp này, Tripwire tái tạo lại toàn bộ cơ sở dữ liệu cho mọi tập tin được chỉ định. Một bản copy của cơ sở dữ liệu cũ được tạo trong thư mục "./databases".

Một số cách có thể sử dụng phần mềm Tripwire

Tripwire có thể được dùng cho:

1. Kiểm tra toàn bộ hệ thống của bạn.
2. Lấy một danh sách của các tập tin mới được cài đặt hay bị xóa trên hệ thống của bạn.

Các tập tin cài đặt

> /etc/cron.daily/tripwire.verify
> /etc/tw.config
> /usr/man/man5/tw.config.5
> /usr/man/man8/siggen.8
> /usr/man/man8/tripwire.8
> /usr/sbin/tripwire
> /usr/sbin/siggen
> /var/spool/tripwire
> /var/spool/tripwire/tw.db_TEST

Một số site liên quan đến Tripwire

ViperDB
ViperDB Homepage: http://www.resentment.org/projects/viperdb/

FCHECK
FCHECK Homepage: http://sites.netscape.net/fcheck/fcheck.html

Sentinel
Sentinel Homepage: http://zurk.netpedia.net/zfile.html

Chương 13:

PHẦN MỀM BẢO MẬT
(Sự quản lý và sự giới hạn)

Trong chương này chúng ta sẽ đề cập đến các vấn đề sau:

- Linux GnuPG.

  Các lệnh.

- Thiết lập hạn ngạch (quota) trên hệ thống Linux của bạn.

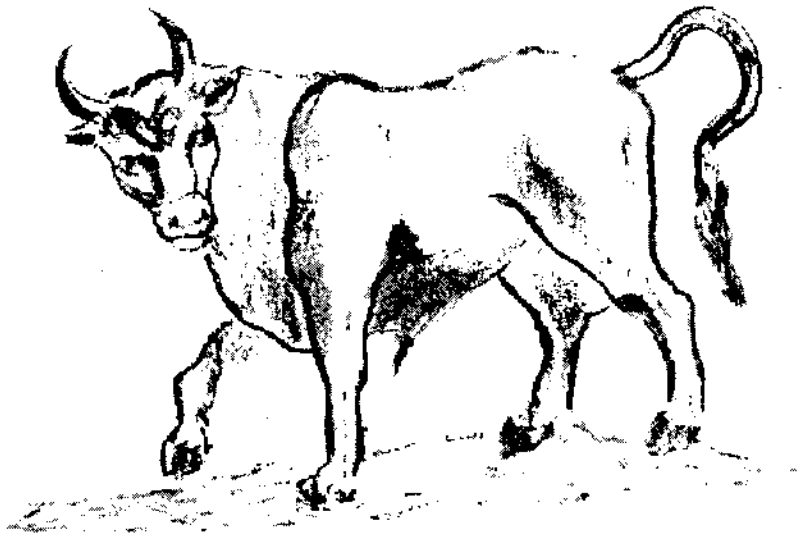
  Xây dựng một kernel hỗ trợ Quota.

  Thay đổi tập tin "/etc/fstab".

  Tạo các tập tin "quota.user" và "quota.group".

  Ấn định Quota cho các người dùng và các nhóm.

  Các lệnh.



Linux GnuPG

Giới thiệu tổng quan

Việc mã hóa dữ liệu là một đặc tính vô giá, nó cung cấp cho chúng ta một mức độ tin cậy cao trong bảo mật cho công việc của chúng ta. GnuPG là một công cụ làm nhiều chức năng không chỉ có việc mã hóa các thông điệp mail, nó còn có thể được dùng cho tất cả các loại mã hoá dữ liệu, và sự tận dụng của nó có thể được mở rộng với nhiều ứng dụng khác nữa tùy vào khả năng sáng tạo và ứng dụng của bạn. Gói dữ liệu (package) rpm GnuPG đã có sẵn và được cài đặt trên máy tính của bạn, nhưng phiên bản này chưa cập nhật và bạn nên cài đặt phiên bản mới nhất đã có để thích hợp với máy server và kiến trúc CPU của bạn.

Thông qua tập tin README của GnuPG package chúng ta sẽ thấy:
GnuPG là một công cụ của GNU dành cho việc bảo mật trong việc trao đổi thông tin và lưu trữ dữ liệu. Nó có thể dùng để mã hoá dữ liệu và tạo các chữ ký điện tử. Nó bao gồm tính năng quản lý một khóa cao cấp và tương thích với chuẩn internet OpenPGP đưa ra như đã được mô tả trong RFC2440.

Bởi vì GnuPG không sử dụng bất kỳ giải thuật mã hoá đã có bằng sáng chế nên nó không thể tương thích với các phiên bản PGP2. PGP 2.x chỉ dùng IDEA (có bằng sáng chế trên toàn cầu) và RSA (có bằng sáng chế trong nước Mỹ cho đến 20-09-2000).

Tóm tắt các chỉ thị cài đặt này

Các dòng lệnh thì tương thích với Unix.

Đường dẫn nguồn là "/var/tmp" (các đường dẫn khác là cũng có thể).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Phiên bản của GnuPG là 1.0.1

Các gói cần thiết (Packages)

GnuPG Homepage Site: http://www.gnupg.org/

Bạn phải bảo đảm là tải về gói : gnupg-1.0.1.tar.gz

Dạng lưu trữ Tarballs

Trước tiên tạo một danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt GnuPG, và sau đó so sánh chúng bằng lệnh **"diff"** để tìm ra các tập tin nào và được đặt ở đâu. Đơn giản chạy lệnh **"find /* > GnuPG1"** trước và **"find /* > GnuPG2"** sau khi bạn cài đặt phần mềm, và sử dụng lệnh **"diff GnuPG1 GnuPG2 > GnuPG-Installed"** để lấy một danh sách các thay đổi nào đó.

Quá trình biên dịch

Giải nén dạng lưu trữ tarball (tar.gz)

```
[root@deep /]# cp gnupg-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf gnupg-version.tar.gz
```

Biên dịch và tối ưu

Chuyển vào trong thư mục mới GnuPG và gõ vào như sau trên máy trạm của bạn:

```
CC="egcs" \
CFLAGS="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomit-frame-
pointer -fno-exceptions" \
./configure \
--prefix=/usr \
--enable-shared

[root@deep gnupg-1.0.1]# make
[root@deep gnupg-1.0.1]# make check
[root@deep gnupg-1.0.1]# make install
[root@deep gnupg-1.0.1]# strip /usr/bin/gpg
```

Lệnh trên "**make**" biên dịch tất cả các tập tin nguồn sang các tập tin dạng nhị phân có thể thi hành được, và "**make check**" sẽ chạy bất kỳ các trình thử nghiệm đi kèm với các package và cuối cùng lệnh "**make install**" cài đặt các tập tin dạng nhị phân và bất kỳ các tập tin hỗ trợ vào những nơi thích hợp. Lệnh "**strip**" sẽ giảm kích thước của dạng nhị phân "gpg" để thực hiện tốt hơn.

Dọn dẹp sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf gnupg-version/ gnupg-version.tar.gz
```

Lệnh "**rm**" được dùng ở trên sẽ xóa tất cả các tập tin nguồn mà chúng ta đã dùng để biên dịch và cài đặt GnuPG. Nó cũng sẽ xóa dạng lưu trữ nén GnuPG trong thư mục "/var/tmp".

Các lệnh

Các lệnh được liệt kê bên dưới là một số lệnh mà chúng ta thường dùng, nhưng có rất nhiều lệnh sẵn có. Kiểm tra man page để biết thêm thông tin chi tiết.

Tạo một khóa

Trước hết, chúng ta phải tạo một cặp khoá mới (bao gồm public và private) nếu đây là lần đầu sử dụng phần mềm của GnuPG thì có thể dùng các đặc tính mã hóa của nó.

Bước 1

- Để tạo một cặp khóa, dùng lệnh sau:

```
[root@deep /]# gpg --gen-key
gpg (GnuPG) 1.0.1; Copyright (C) 1999 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

gpg: /root/.gnupg: directory created
gpg: /root/.gnupg/options: new options file created
gpg: you have to start GnuPG again, so it can read the new options file
This asks some questions and then starts key generation.
```

Bước 2

- Để khởi động GnuPG lại với lệnh sau:

```
[root@deep /]# gpg --gen-key
gpg (GnuPG) 1.0.1; Copyright (C) 1999 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

gpg: /root/.gnupg/secring.gpg: keyring created
gpg: /root/.gnupg/pubring.gpg: keyring created
Please select what kind of key you want:
   (1) DSA and ElGamal (default)
   (2) DSA (sign only)
   (4) ElGamal (sign and encrypt)
Your selection? 1
DSA keypair will have 1024 bits.
About to generate a new ELG-E keypair.
minimum keysize is 768 bits
default keysize is 1024 bits
   highest suggested keysize is 2048 bits
What keysize do you want? (1024) 2048
Do you really need such a large keysize? y
Requested keysize is 2048 bits
Please specify how long the key should be valid.
0 = key does not expire
<n> = key expires in n days
<n>w = key expires in n weeks
<n>m = key expires in n months
<n>y = key expires in n years
Key is valid for? (0) 0
correct (y/n)? y

You need a User-ID to identify your key; the software constructs the user id
```

```
from Real Name, Comment and Email Address in this form:
"Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

        Real name: Gerhard Mourani
Email address: gmourani@videotron.ca
Comment: [Press Enter]
You selected this USER-ID:
"Gerhard Mourani <gmourani@videotron.ca>"

        Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o
You need a Passphrase to protect your secret key.

We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
+++++..+++++++++++++++..+++++.+++++++++++++++++++++++++
+++++++++++++++..++++++++++.+++++++++++++++++++++++++
+.+++++++++++++++...++++++++++++++++++++++++.++++
+
..+++++>+++++...+++++++++++++++>+++++.....>+++++.....<++++
+.....
.....+++++^^^^
public and secret key created and signed.
```

Một cặp khoá được tạo (bao gồm một khoá bí mật và một khoá công khai-public key) trong thư mục home "root" (~/root).

Nhập các khoá vào (Importing Keys)

Một khi cặp khóa riêng của chúng ta được tạo, chúng ta có thể bắt đầu đặt public key của chúng ta vào cơ sở dữ liệu chứa tất cả các khóa (public keyring database) chúng ta có từ một số third party đáng tin cậy theo thứ tự để có thể dùng cặp khoá của anh ta/cô ta cho việc mã hóa và chứng thực trong việc trao đổi thông tin.

- Để nhập vào (import) Public Keys tới keyring của chúng ta, dùng lệnh sau:

```
[root@deep /]# gpg --import <file>
```

Như ví dụ sau:

```
[root@deep /]# gpg --import Red Hat2.asc
gpg: key DB42A60E: public key imported
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: Total number processed: 1
gpg: imported: 1
```

Lệnh ở trên sẽ thêm tất cả các khoá mới tới cơ sở dữ liệu khoá (keyring database) của chúng ta và sẽ cập nhật tất cả các khoá đã có sẵn. Điều quan trọng đáng lưu ý là GnuPG không nhập vào các khoá mà nó chưa được tự đăng ký (self-signed). Trong ví dụ ở trên chúng ta nhập vào tập tin Public Key "Red Hat2.asc"

từ công ty Red Hat Linux, được tải về từ site Red Hat trên Internet, vào keyring của chúng ta.

Ký nhận khoá (Key signing)

Khi bạn nhập các khoá (import keys) vào trong cơ sở dữ liệu khoá công cộng (public keyring) và bảo đảm rằng đã tin tưởng người ký nhận (third party) trong public key do mọi người yêu cầu, Bạn có thể khởi tạo ký nhận(signing) các khoá của anh ta/cô ta. Việc ký nhận một khoá xác định rằng bạn biết được chủ sở hữu của các khóa.

- Để ký nhận một khoá cho công ty Red Hat mà chúng ta đã thêm vào cơ sở dữ liệu keyring ở trên, dùng lệnh sau:

  [root@deep /]# **gpg --sign-key <UID>**

  Như ví dụ sau:

  [root@deep /]# **gpg --sign-key Red Hat**

  pub 1024D/DB42A60E created: 1999-09-23 expires: never trust: -/q
  sub 2048g/961630A2 created: 1999-09-23 expires: never
  (1) Red Hat, Inc security@Red Hat.com

  pub 1024D/DB42A60E created: 1999-09-23 expires: never trust: -/q
  Fingerprint: CA20 8686 2BD6 9DFC 65F6 ECC4 2191 80CD DB42 A60E

  Red Hat, Inc <security@Red Hat.com>
  Are you really sure that you want to sign this key
  with your key: "Gerhard Mourani <gmourani@videotron.ca>"

  Really sign? **y**

  You need a passphrase to unlock the secret key for
  user: "Gerhard Mourani <gmourani@videotron.ca>"
  1024-bit DSA key, ID E92D6C97, created 1999-12-30

  Enter passphrase:

Ghi chú: *Bạn chỉ nên ký nhận một khóa để xác nhận khi mà bạn bảo đảm tuyệt đối rằng khoá là thật sự đáng tin ! Bạn nên đừng bao giờ ký nhận một khoá dựa trên bất kỳ sự mệnh danh nào.*

Mã hoá và giải mã

Sau khi cài đặt, nhập khoá vào, ký nhận khóa và cấu hình mọi thứ theo cách mà chúng ta muốn, chúng ta có thể khởi tạo việc mã khoá và giải mã của chúng ta.

- Mã hoá và ký nhận dữ liệu cho người dùng Red Hat mà chúng ta đã thêm vào cơ sở dữ liệu ở trên của chúng ta, dùng lệnh sau:

  [root@deep /]# **gpg -sear Red Hat <file>**

Như ví dụ sau:

```
[root@deep /]# gpg -sear Red Hat message-to-Red Hat.txt
You need a passphrase to unlock the secret key for
user: "Gerhard Mourani (Open Network Architecture)
<gmourani@videotron.ca>"
1024-bit DSA key, ID BBB4BA9B, created 1999-10-26
Enter passphrase:
```

Các đối số hợp lệ trên gồm: "s" dành cho việc ký (signing) (Để tránh rủi ro mà một người nào đó tự cho là bạn, nó rất hữu ích để ký mọi thứ mà bạn mã hóa), "e" dành cho việc mã hóa (encrypting), "a" để tạo vỏ bọc ASCII cho việc xuất ra ngoài (".asc" sẵn sàng cho việc gửi bằng mail), "r" để mã hóa chỉ danh tên người dùng và <file> là thông điệp bạn muốn mã hoá.

- Để giải mã dữ liệu, dùng lệnh sau:

```
[root@deep /]# gpg -d <file>
```

Lấy ví dụ:

```
[root@deep /]# gpg -d message-to-Gerhard.asc
You need a passphrase to unlock the secret key for
user: "Gerhard Mourani (Open Network Architecture)
<gmourani@videotron.ca>"
2048-bit ELG-E key, ID 71D4CC44, created 1999-10-26 (main key ID
BBB4BA9B)
Enter passphrase:
```

Trong đó "-d" là dùng cho việc giải mã (decrypting) và <file> là thông điệp mà bạn muốn giải mã. Một điều quan trọng là public key của người gửi thông điệp mà chúng ta muốn giải mã phải có trong cơ sở dữ liệu "public keyring" của chúng ta, nếu không chương trình sẽ không làm việc.

Xuất ra khoá public của bạn (Exporting your public key)

Bạn có thể mở rộng phạm vi của bạn bằng cách xuất ra và phân phối public key của bạn trên khắp thế giới. Điều này có thể được thực hiện bằng cách phổ biến nó lên trên trang chủ (home page) của bạn, thông qua một key server có sẵn trên internet, hoặc bất kỳ phương pháp nào khác. GnuPG có một số tùy chọn để giúp bạn phổ biến các public key của bạn.

- Để trích ra public key của bạn theo dạng mã ASCII, dùng lệnh sau:

```
[root@deep /]# gpg --export --armor > Public-key.asc
```

Trong đó "--export" là dùng cho việc trích ra Public-key của bạn từ tập tin được mã hóa, "--armor" là để tạo ra dạng ASCII mà bạn có thể mail, phổ biến hay đặt nó trên một trang web và ">Public-key.asc" là để đặt kết quả trong một tập tin mà bạn đã đặt tên Public-key.asc.

Kiểm tra chữ ký (Checking the signature)

Một khi bạn đã trích ra public key của bạn và xuất ra nó, mọi người biết hoặc lấy public key của bạn nên có thể kiểm tra xem dữ liệu có được mã hoá hay không và có thật được ký bởi chính bạn hay không.

- Kiểm tra chữ ký của dữ liệu được mã hoá, dùng lệnh sau:

```
[root@deep /]# gpg --verify <Data>
```

Tùy chọn "--verify" sẽ kiểm tra chữ ký trong đó <Data> là dữ liệu/ tập tin được mã hoá mà bạn muốn kiểm tra.

Một số cách có thể dùng phần mềm GnuPG

GnuPG có thể được dùng:

1. Gửi thông điệp mail được mã hóa.
2. Mã hóa các tập tin dự phòng trước khi truyền tải qua mạng.
3. Mã hóa các tập tin riêng rẽ có tính nhạy cảm (nghĩa là một tập tin mà xử lý tất cả các mật khẩu của bạn).

Các tập tin được cài đặt

```
> /usr/bin/gpg
> /usr/lib/gnupg
> /usr/lib/gnupg/rndunix
> /usr/lib/gnupg/rndegd
> /usr/lib/gnupg/tiger
> /usr/man/man1/gpg.1
> /usr/share/gnupg
> /usr/share/gnupg/options.skel
```

Thiết đặt hạn ngạch (Quota) trên hệ thống Linux

Giới thiệu tổng quan

Quota là một công cụ quản trị hệ thống dùng để theo dõi và giới hạn cách sử dụng đĩa của các người sử dụng và/hoặc 'các nhóm người sử dụng' trên tập tin hệ thống. Hai đặc tính của việc lưu trữ đĩa với "quota" đã có sẵn cho việc đặt các giới hạn: trước tiên là số inode (số tập tin) một hay một nhóm người sử dụng có thể chiếm hữu và thứ hai là số disk block (số lượng khoảng trống đĩa tính bằng kilobytes-kb) có thể được cấp phát cho một hay một nhóm các người sử dụng. Với quota, người quản trị hệ thống có thể ấn định vùng không gian đĩa được phép sử dụng đến từng người sử dụng (user) và những user này không được phép sử dụng các vùng khác trên hệ thống. Chương trình này được xử lý trên một người sử dụng, trên cơ sở hệ thống tập tin và phải đặt cho mỗi hệ thống tập tin một cách riêng biệt.

Xây dựng một kernel hỗ trợ Quota

Việc đầu tiên mà bạn cần phải làm là bảo đảm rằng hạt nhân (kernel) của bạn đã được xây dựng với sự hỗ trợ của Quota đã có tác dụng. Trong phiên bản kernel 2.2.14 bạn cần bảo đảm rằng bạn có câu trả lời Y với các câu hỏi sau đây:

```
Filesystems
Quota support (CONFIG_QUOTA) [N/y/?] Y
```

Ghi chú: *Nếu bạn đã theo dõi chương Linux Kernel trong quyển sách này và đã biên dịch lại kernel của bạn, thì tùy chọn "Quota support" được trình bày ở trên được thiết lập sẵn rồi.*

Thay đổi tập tin "/etc/fstab"

Tập tin "/etc/fstab" chứa đựng thông tin về các hệ thống tập tin khác nhau được cài đặt trên máy Linux server của bạn. Quota phải được cho phép trong tập tin fstab trước khi bạn có thể dùng nó. Bởi vì Quota phải được đặt cho mỗi hệ thống tập tin một cách riêng biệt, và bởi vì trong tập tin fstab, mỗi hệ thống tập tin là được mô tả trên một dòng riêng biệt, quota phải được đặt trên mỗi dòng riêng biệt trong tập tin fstab cho phần mà bạn cho phép hỗ trợ quota.

Với chương trình quota, phụ thuộc vào các mục đích của bạn, các nhu cầu, v.v..., bạn có thể cho phép chỉ quota đối với các người sử dụng, nhóm người sử dụng hay cả hai (bao gồm các người sử dụng và các nhóm người sử dụng). Như các ví dụ bên dưới, chúng ta sẽ dùng thư mục "/home" trên partition "/dev/sda6" và trình bày cho bạn ba khả năng có thể.

Khả năng 1:

- Để cho phép hỗ trợ quota cho người dùng (user quota) trên một hệ thống tập tin chỉ định, soạn thảo tập tin fstab (vi /etc/fstab) và thêm vào tùy chọn

"usrquota" tới trường (field) thứ tư sau từ "defaults" hay bất kỳ các tùy chọn nào khác mà bạn có thể đặt cho hệ thống tập tin chỉ định này.

Như ví dụ sau, thay đổi:

```
/dev/sda6 /home          ext2  defaults 1 2
/dev/sda6 /home          ext2  nosuid,nodev 1 2
```

Thành:

```
/dev/sda6 /home  ext2  defaults,usrquota 1 2
/dev/sda6 /home  ext2  nosuid,nodev,usrquota 1 2
```

Khả năng 2:

- Để cho phép hỗ trợ quota cho nhóm người dùng (group quota) trên một hệ thống tập tin, soạn thảo tập tin fstab (vi /etc/fstab) và thêm vào tùy chọn "grpquota" cho trường (field) thứ tư sau từ "defaults" hay bất kỳ các tùy chọn nào khác mà bạn có thể đặt cho hệ thống tập tin chỉ định này.

Như ví dụ sau, thay đổi:

```
/dev/sda6 /home          ext2  defaults 1 2
/dev/sda6 /home          ext2  nosuid,nodev 1 2
```

Thành:

```
/dev/sda6 /home  ext2  defaults,grpquota 1 2
/dev/sda6 /home  ext2  nosuid,nodev,grpquota 1 2
```

Khả năng 3:

- Để cho phép hỗ trợ quota cho cả hai bao gồm người sử dụng và nhóm người dùng trên một hệ thống tập tin chỉ định, soạn thảo tập tin fstab (vi /etc/fstab) và thêm vào tùy chọn "usrquota, grpquota" tới trường (field) thứ tư sau từ "defaults" hay bất kỳ các tùy chọn nào khác mà bạn có thể đặt cho hệ thống tập tin chỉ định này.

Như ví dụ sau, thay đổi:

```
/dev/sda6 /home          ext2  defaults 1 2
/dev/sda6 /home          ext2  nosuid,nodev 1 2
```

Thành:

```
/dev/sda6 /home  ext2  defaults, usrquota, grpquota 1 2
/dev/sda6 /home  ext2  nosuid,nodev, usrquota, grpquota 1 2
```

Tạo các tập tin "quota.user" và "quota.group"

Sau khi hiệu chỉnh tập tin "/etc/fstab", trong thứ tự cho các quota được thiết lập trên một hệ thống tập tin, thư mục gốc của hệ thống tập tin (nghĩa là thư mục /home trong ví dụ của chúng ta) phải chứa đựng một tập tin được sở hữu riêng bởi root, được gọi là "quota.user" nếu như bạn muốn dùng user quota, tương tự

"quota.group" nếu như bạn muốn dùng group quota, hoặc cả hai nếu như bạn muốn dùng user và group quota.

Bước 1

- Để tạo các tập tin "quota.user" và/hoặc "quota.group", như "root" vào thư mục gốc của partition mà bạn muốn cho phép thiết đặt quota (nghĩa là thư mục /home) và tạo "quota.user" và/hoặc "quota.group" bằng cách thực hiện:

```
[root@deep /]# touch /home/quota.user
[root@deep /]# touch /home/quota.group
[root@deep /]# chmod 600 /home/quota.user
[root@deep /]# chmod 600 /home/quota.group
```

Lệnh "touch" sẽ tạo các tập tin rỗng mới trong thư mục "home" được đặt tên là "quota.user" và "quota.group". Lệnh "chmod" sẽ đặt chế độ của các tập tin này là chỉ đọc-ghi (read-write) bởi super-user "root".

Ghi chú: *Cả hai tập tin quota thu được, "quota.user" và "quota.group", sẽ được sở hữu riêng bởi root, với quyền đọc-ghi cho root và không cho bất kỳ người dùng nào khác.*

Bước 2

Bây giờ chúng ta phải khởi tạo các tập tin "quota.user" và "quota.group" trong thư mục gốc của hệ thống tập tin để không nhận một thông điệp lỗi về quota trong suốt quá trình khởi động lại server của bạn.

- Để khởi tạo các tập tin "quota.user" và/hoặc "quota.group", dùng các lệnh sau:

```
[root@deep /]# edquota -u wahib
[root@deep /]# edquota -g wahib
```

Các bước trên chỉ cần thiết để khởi tạo các tập tin "quota.user" và/hoặc "quota.group"; lệnh edquota (-u) sẽ hiệu chỉnh quota cho người dùng "wahib" và (-g) sẽ hiệu chỉnh quota cho nhóm wahib. Chú ý rằng bạn phải hiệu chỉnh một UID/GID có sẵn trên hệ thống của bạn để khởi tạo các tập tin thành công.

Bước 3

Sau khi bạn đã kết thúc cài đặt các tùy chọn phù hợp cho chương trình quota trong tập tin "/etc/fstab", các tập tin "quota.user" và/hoặc "quota.group" đã được tạo và được khởi tạo chuẩn bị làm việc, bạn phải khởi động lại hệ thống để việc thay đổi mà bạn đã tạo trong tập tin "/etc/fstab" và/hoặc các tập tin "quota.user, quota.group" có hiệu lực.

- Để khởi động lại hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# reboot
```

Ấn định Quota cho các người dùng và các nhóm

Sau khi hệ thống của bạn đã được khởi động lại bạn có thể gán các quota cho các người và các nhóm người sử dụng trên hệ thống của bạn. Thao tác này được thực hiện với lệnh "edquota". edquota(8).

Chương trình edquota

Chương trình edquota là một chương trình soạn thảo quota mà nó tạo ra một tập tin tạm thời của các quota đĩa hiện hành được dùng bởi super-user "root" để tạo các quota cho các người sử dụng hay nhóm các người sử dụng trong hệ thống. Ví dụ bên dưới trình bày cho bạn biết làm thế nào để cài đặt các quota cho các người sử dụng hay nhóm các người sử dụng trong hệ thống.

Gán quota cho một người sử dụng riêng biệt

Lưu ý: *Lấy ví dụ rằng bạn có một người sử dụng với mã vào hệ thống (login id) là "wahib" trên hệ thống của bạn. Dòng lệnh dưới đây giúp cho việc thiết đặt quota cho user "wahid" trên mỗi partition có quota được phép:*

Bước 1:

- Để soạn thảo và hiệu chỉnh quota cho người sử dụng "wahib", dùng lệnh sau:

```
[root@deep /]# edquota -u wahib
```

Các quota dành cho người sử dụng wahib:

```
/dev/sda6: blocks in use: 6, limits (soft = 0, hard = 0)
        inodes in use: 5, limits (soft = 0, hard = 0)
```

Sau khi thi hành lệnh trên, bạn sẽ thấy các dòng sau liên quan tới người dùng "wahib" xuất hiện trên màn hình. Và "blocks in use:" hiển thị tổng số khối (block) (tính bằng kilobytes) mà người dùng đã chiếm trên một partition. Dòng "inodes in use" hiển thị tổng số các tập tin mà người sử dụng đang dùng trên một partition. Các thông số này ("blocks in use, và inodes in use") được điều khiển và được cài đặt một cách tự động bởi hệ thống và bạn không cần phải cài đặt hay sửa đổi chúng.

Bước 2:

- Để gán 5MB quota cho người sử dụng "wahib, thay đổi các thông số sau trong trình soạn thảo vi:

Các quota dành cho người sử dụng wahib:

```
/dev/sda6: blocks in use: 6, limits (soft = 0, hard = 0)
inodes in use: 5, limits (soft = 0, hard = 0)
```

Chuyển thành:

Các quota dành cho người sử dụng wahib:

```
/dev/sda6: blocks in use: 6, limits (soft = 5000, hard = 0)
```

inodes in use: 5, limits (soft = 0, hard = 0)

Với "soft limit" (soft=) chỉ rõ số lượng dùng đĩa tối đa một quota user là được cho phép có.

Với "hard limit" (hard=) chỉ rõ giới hạn tuyệt đối về việc dùng đĩa một quota user không thể vượt quá nó. Ghi chú rằng giá trị "hard limit" chỉ làm việc khi thông số "grace period" được đặt.

Thông số "grace period"

Thông số "grace period" cho phép bạn đặt một giới hạn thời gian trước khi giá trị giới hạn bị bắt buộc trên một hệ thống tập tin với quota được hỗ trợ (xem soft limit ở trên để biết thêm thông tin). Lấy ví dụ thông số này có thể được dùng để cảnh báo các người sử dụng của bạn về một chính sách mới mà sẽ đặt một quota 5MB không gian đĩa trong thư mục home của họ trong 7 ngày. Bạn có thể đặt phần mặc định 0 ngày cho tham số này tới bất kỳ thời gian nào mà bạn cảm thấy hợp lý.

Việc thay đổi các thông số cài đặt này yêu cầu hai bước sau (trong ví dụ của tôi, tôi lấy 7 ngày).

Bước 1

- Soạn thảo thông số grace period, bằng cách dùng lệnh sau:

```
[root@deep /]# edquota -t
Time units may be: days, hours, minutes, or seconds
Grace period before enforcing soft limits for users:
/dev/sda6: block grace period: 0 days, file grace period: 0 days
```

Bước 2

- Hiệu chỉnh grace period là 7 ngày. Thay đổi hay đặt các thông số sau trong trình soạn thảo vi:

```
Time units may be: days, hours, minutes, or seconds
Grace period before enforcing soft limits for users:
/dev/sda6: block grace period: 0 days, file grace period: 0 days
```

Trở thành:

```
Time units may be: days, hours, minutes, or seconds
Grace period before enforcing soft limits for users:
/dev/sda6: block grace period: 7 days, file grace period: 7 days
```

Ghi chú: *Lệnh "edquota -t" hiệu chỉnh thời gian soft giới hạn cho mỗi hệ thống tập tin với sự hỗ trợ quota là được phép.*

Gán quota cho một nhóm riêng biệt

Lưu ý: *Lấy ví dụ rằng bạn có một nhóm người sử dụng với mã vào hệ thống (login id) là "webusers" trên hệ thống của bạn. Lệnh sau dẫn dắt bạn*

vào trình soạn thảo vi để soạn thảo quota cho nhóm "webusers" trên mỗi partition mà các quota được phép:

- Soạn thảo và hiệu chỉnh quota cho nhóm "webusers", dùng lệnh sau:

```
[root@deep /]# edquota -g webusers
Quotas for group webusers:
/dev/sda6: blocks in use: 6, limits (soft = 0, hard = 0)
inodes in use: 6, limits (soft = 0, hard = 0)
```

Thủ tục giống như đối với gán quota cho một người sử dụng riêng biệt như được mô tả ở trên, bạn phải hiệu chỉnh thông số của "soft=" và lưu lại thay đổi của bạn.

Gán quota cho các nhóm người sử dụng với cùng giá trị

Chương trình edquota có một tùy chọn đặc biệt (-p), nó gán các quota cho các nhóm người sử dụng với cùng giá trị được gán cho một người sử dụng ban đầu. Giả sử rằng bạn muốn gán các người sử dụng bắt đầu ở UID 500 trên hệ thống giống giá trị như người dùng "wahib", trước tiên chúng tôi sẽ soạn thảo và cài đặt thông tin quota của wahib, sau đó thi hành:

- Để gán quota cho nhóm người sử dụng với cùng giá trị, dùng lệnh sau:

```
[root@deep /]# edquota -p wahib `awk -F: '$3 > 499 {print $1}' /etc/passwd`
```

Chương trình edquota sẽ nhân đôi quota mà chúng ta cài đặt cho người sử dụng "wahib" cho tất cả người sử dụng trong tập tin "/etc/password" mà bắt đầu sau UID 499.

Tham khảo tài liệu kỹ thuật chi tiết

Để biết thêm chi tiết, có rất nhiều man page mà bạn có thể đọc:

$ man edquota (8)	**- edit user quotas**
$ man quota (1)	**- display disk usage and limits**
$ man quotacheck (8)	**- scan a file system for disk usages**
$ man quotactl (2)	- manipulate disk quotas
$ man quotaon, quotaoff (8)	**- turn file system quotas on and off**
$ man repquota (8)	**- summarize quotas for a taăp tin system**
$ man rquota (3)	**- implement quotas on remote machines**

Các lệnh

Các lệnh được liệt kê bên trên là một số lệnh mà chúng ta thường dùng, nhưng còn nhiều lệnh hơn có sẵn. Kiểm tra man page để biết thêm thông tin chi tiết.

Quota

Quota hiển thị khoảng không gian đĩa của người sử dụng và các giới hạn trên một hệ thống tập tin.

- Để hiển thị việc dùng đĩa của người sử dụng và các giới hạn, dùng lệnh:

```
[root@deep /]# quota -u wahib
Disk quotas for user wahib (uid 501):
  Filesystem  blocks  quota  limit  grace  files  quota  limit  grace
  /dev/sda6   6001*   6000   0      none   7      0      0
```

- Để hiển thị group quota cho nhóm mà người sử dùng là thành viên, dùng lệnh:

```
[root@deep /]# quota -g wahib
Disk quotas for group wahib (gid 501):
  Filesystem  blocks  quota  limit  grace  files  quota  limit  grace
  /dev/sda6   5995*   5000   0      none   1      0      0
```

Ghi chú: *Nếu group quota không được cài đặt cho người dùng được chỉ định, thì bạn sẽ nhận được thông điệp sau:*

```
Disk quotas for group wahib (gid 501): none
```

Repquota

Thủ tục Repquota tóm tắt lại các thông tin quota của việc dùng đĩa và các quota cho các hệ thống tập tin được chỉ định. Nó cũng in ra cho mỗi người dùng số lượng tập tin hiện hành và số lượng không gian được dùng (tính bằng kilobytes).

- Đây là một ví dụ mẫu xuất ra các repquota cho bạn (các kết quả có thể khác nhau):

```
[root@deep /]# repquota -a
```

Block limits					File limits			
User	used	soft	hard	grace	used	soft	hard	grace
Roo --	21	0	0	4	0	0		
Named --	6	0	0	5	0	0		
Admin --	388657		0	0	21	21	0	0
Wahib --	6001	0	0	7	0	0		

Block limits				File limits				
User	used	soft	hard	grace	used	soft	hard	grace

root	--	21	0	0	4	0	0	
named	--	6	0	0	5	0	0	
admin	--	388657		0	0	2121	0	0
wahib	--	6001	6000	0	none	7	0	0

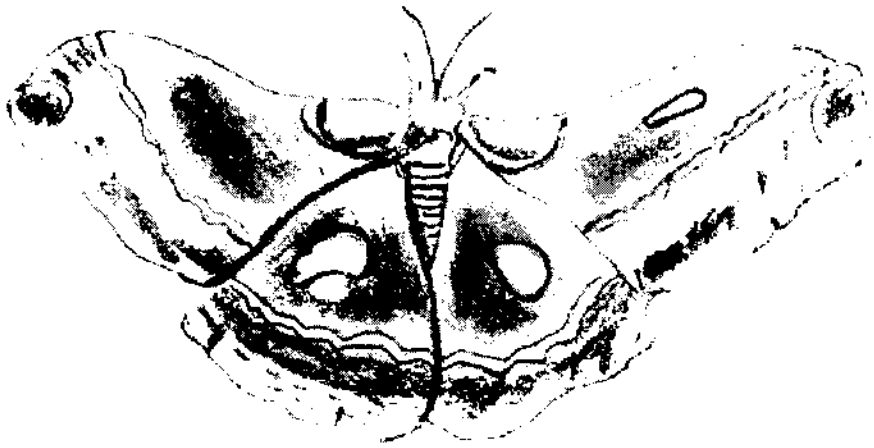
Chương 14:

PHẦN MỀM SERVER

(Dịch vụ mạng BIND/DNS)

Trong chương này chúng ta sẽ đề cập đến các phần sau:

- Server Linux DNS và BIND.
- Cấu hình.
- Caching-only name server.
- Primary master name server.
- Secondary slave name server.
- An toàn cho ISC BIND/DNS.
- Công cụ quản trị DNS.
- Công cụ sử dụng DNS.



Server Linux DNS và BIND

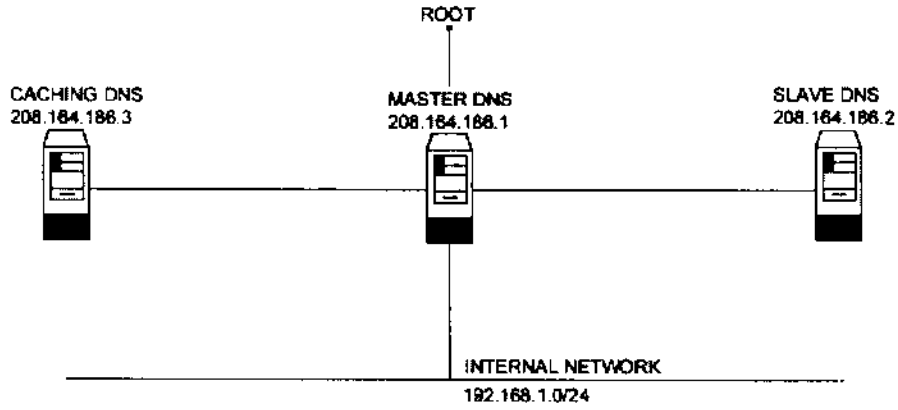
Tổng quan

Một khi chúng ta đã cài đặt tất cả phần mềm bảo mật cần thiết trên Linux server, đây là thời điểm để cải tiến và điều chỉnh phần mạng (network) của server của chúng ta. DNS là một trong những dịch vụ mạng quan trọng nhất cho sự trao đổi thông tin trên mạng IP, và vì lý do này, tất cả các máy Linux client sẽ được cài đặt những chức năng lưu giữ (caching) ở một mức độ tối thiểu nào đó. Việc cài đặt một caching server cho các máy client nội bộ sẽ làm giảm bớt tải trên các máy primary server. Một Caching chỉ rõ tên máy chủ sẽ tìm kiếm trả lời cho những tên truy vấn và ghi nhớ phần đáp án này để khi nào chúng ta cần, nó đáp ứng ngay không cần mất nhiều thời gian vô ích.

Vì những nguyên do bảo mật, điều rất quan trọng là DNS không tồn tại sẵn giữa các máy trên mạng và máy bên ngoài. Để tăng tính năng an toàn hơn, đơn giản dùng các địa chỉ IP kết nối với những máy bên ngoài từ bên trong mạng và ngược lại.

Trong cấu hình và cài đặt, chúng ta sẽ chạy chương trình BIND/DNS với user không phải root và trong một môi trường chrooted. Chúng tôi sẽ cung cấp cho bạn ba cấu hình khác nhau: một cái đơn giản chỉ lưu trữ tên máy (client), cái thứ hai là cho slave (secondary server) và cái thứ ba là cho master name server (primary server).

Cấu hình thứ nhất simple caching name server sẽ được dùng cho máy chủ của bạn mà không hoạt động như master hoặc slave name server, cấu hình của slave và master sẽ được dùng cho máy chủ của bạn mà hoạt động như master và slave name server. Thường thường, cấu hình sẽ bao gồm: một cái sẽ hoạt động như master, cái khác như slave và cái còn lại tương tự như simple caching client server.



Đây là hình ảnh minh họa cấu hình của DNS mà chúng tôi dùng trong quyển sách này. Chúng tôi cố gắng chỉ ra những cách cài đặt khác nhau (caching

only DNS, Master DNS, và Slave DNS) trên những máy chủ khác nhau. Một số có thể tồn tại, phụ thuộc vào nhu cầu cần thiết cũng như cấu trúc mạng của bạn.

Hướng dẫn cài đặt

Các lệnh này tương thích với hệ điều hành Unix.

Đường dẫn nguồn là "/var/tmp" (cũng có thể chọn đường dẫn khác)

Sự cài đặt này được kiểm tra trên Red Hat Linux 6.1 và 6.2

Tất cả các bước cài đặt sẽ sử dụng tài khoản super-user account "root".

Số phiên bản ISC BIND là 8.2.2- patchlevel5

Gói (Packages)

ISC BIND Homepage: http://www.isc.org/

ISC BIND FTP Site: 204.152.184.27

Bạn phải chắc chắn chép xuống : bind-contrib.tar.gz, bind-doc.tar.gz, bind-src.tar.gz

Tarballs

Trước tiên hãy tạo ra danh sách của các tập tin trên hệ thống trước và sau khi bạn cài đặt BIND, sau đó so sánh chúng bằng cách dùng lệnh "diff" để tìm ra tập tin nào được đặt ở đó. Chỉ đơn giản thực hiện lệnh "find /*> DNS1" trước và "find /*> DNS2" và sau đó thực hiện lệnh "diff DNS1 DNS2 > DNS-installed" để lấy được danh sách của các tập tin đã được thay đổi.

Biên dịch

Giải nén tarball (tar.gz)

```
[root@deep /]# mkdir /var/tmp/bind
[root@deep /]# cp bind-contrib.tar.gz /var/tmp/bind/
[root@deep /]# cp bind-doc.tar.gz /var/tmp/bind/
[root@deep /]# cp bind-src.tar.gz /var/tmp/bind/
```

Chúng ta tạo một thư mục có tên "bind" để xử lý các tập tin tar và sao chép chúng vào thư mục mới này.

Hãy di chuyển vào trong thư mục mới "bind" (cd /var/tmp/bind) và giải nén những tập tin tar:

```
[root@deep bind]# tar xzpf bind-contrib.tar.gz
[root@deep bind]# tar xzpf bind-doc.tar.gz
[root@deep bind]# tar xzpf bind-src.tar.gz
```

Cấu hình và tối ưu

Bước 1

ISC BIND không thể vận hành như super-user root; vì lý do này chúng ta phải tạo một tài khoản người dùng đặc biệt không có đặc quyền để chạy chương trình ISC BIND/DNS

```
[root@deep /]# useradd -c "DNS Server" -u 53 -s /bin/false -r –d \
/chroot/named named 2>/dev/null || :
```

Bước 2

Hiệu chỉnh tập tin Makefile.set (vi src/port/linux/Makefile.set) và thêm hoặc hiệu chỉnh để thành:

```
'CC=egcs -D_GNU_SOURCE'
'CDEBUG=-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro - fomit-frame-pointer -fno-exceptions -g'
'DESTBIN=/usr/bin'
'DESTSBIN=/usr/sbin'
'DESTEXEC=/usr/sbin'
'DESTMAN=/usr/man'
'DESTHELP=/usr/lib'
'DESTETC=/etc'
'DESTRUN=/var/run'
'DESTLIB=/usr/lib/bind/lib'
'DESTINC=/usr/lib/bind/include'
'LEX=flex -8 -l'
'YACC=yacc -d'
'SYSLIBS=-lfl'
'INSTALL=install'
'MANDIR=man'
'MANROFF=cat'
'CATEXT=$$N'
'PS=ps p'
'AR=ar crus'
'RANLIB=:'
```

Dòng đầu tiên trình bày tên trình biên dịch GCC (egcs), dòng thứ hai có cờ hiệu tối ưu. Dòng "DESTLIB=" chỉ định rõ đường dẫn của thư mục chứa trong thư viện (library) cho chương trình ISC BIND và dòng "DESTINC=" là nơi chúng ta đặt thư mục include của BIND.

Biên dịch và tối ưu

Gõ những lệnh sau trên terminal của bạn

```
[root@deep bind]# make -C src
```

```
[root@deep bind]# make clean all -C src SUBDIRS=../doc/man
[root@deep bind]# make install -C src
[root@deep bind]# make install -C src SUBDIRS=../doc/man
```

Lệnh "make" biên dịch tất cả tập tin nguồn thành những tập tin nhị phân thi hành được, sau đó "make install" cài đặt một số tập tin hỗ trợ vào những vị trí tương ứng.

```
[root@deep bind]# strip /usr/bin/addr
[root@deep bind]# strip /usr/bin/dig
[root@deep bind]# strip /usr/bin/dnsquery
[root@deep bind]# strip /usr/bin/host
[root@deep bind]# strip /usr/bin/nslookup
[root@deep bind]# strip /usr/bin/nsupdate
[root@deep bind]# strip /usr/bin/mkservdb
[root@deep bind]# strip /usr/sbin/named
[root@deep bind]# strip /usr/sbin/named-xfer
[root@deep bind]# strip /usr/sbin/ndc
[root@deep bind]# strip /usr/sbin/dnskeygen
[root@deep bind]# strip /usr/sbin/irpd
[root@deep bind]# mkdir /var/named
```

Lệnh "strip" loại bỏ tất cả các biểu tượng từ những tập tin object. Điều này có nghĩa là tất cả các tập tin nhị phân của chúng ta sẽ có kích cỡ nhỏ lại. Rồi sau đó, việc này sẽ cải tiến kết quả thực hiện vì chỉ một vài dòng sẽ được đọc bởi hệ thống khi nó được thực thi. Lệnh "mkdir" tạo một thư mục mới có tên "/var/named"

Dọn dẹp sau quá trình cài đặt

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf bind/
```

Phần này sẽ gỡ bỏ tất cả tập tin nguồn mà chúng ta đã dùng để biên dịch và cài đặt ISC BIND/DNS.

Cấu hình

Những tập tin cấu hình cho những dịch vụ khác nhau thì rất cụ thể, phụ thuộc vào nhu cầu cần thiết và cấu trúc mạng máy tính của bạn. Nhiều người có thể cài đặt DNS server tại nhà như là caching–only server, mặc dù nhiều công ty có thể cài đặt nó như là primary, secondary và caching DNS server.

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách

- Để chạy trên môi trường caching-only name server, các tập tin sau đòi hỏi phải được tạo hoặc sao chép đến các thư mục tương ứng trên máy chủ của bạn.

  Sao chép tập tin named.conf vào thư mục "/etc/".

Sao chép tập tin db.127.0.0 vào thư mục "/var/named/".

Sao chép tập tin db.cache vào thư mục "/var/named/".

Sao chép tập tin script named vào thư mục "/etc/rc.d/init.d/".

- Để chạy trên môi trường master name server, các tập tin sau đòi hỏi phải được tạo hoặc sao chép đến các thư mục tương ứng trên máy chủ của bạn.

  Sao chép tập tin named.conf vào thư mục "/etc/".

  Sao chép tập tin db.127.0.0 vào thư mục "/var/named/".

  Sao chép tập tin db.cache vào thư mục "/var/named/".

  Sao chép tập tin db.208.164.186 vào thư mục "/var/named/".

  Sao chép tập tin db.openna vào thư mục "/var/named/".

  Sao chép tập tin script named vào thư mục "/etc/rc.d/init.d/".

- Để chạy trên môi trường slave name server, các tập tin sau đòi hỏi phải được tạo hoặc sao chép đến các thư mục tương ứng trên máy chủ của bạn.

  Sao chép tập tin named.conf vào thư mục "/etc/".

  Sao chép tập tin db.127.0.0 vào thư mục "/var/named/".

  Sao chép tập tin db.cache vào thư mục "/var/named/".

  Sao chép tập tin script named vào thư mục "/etc/rc.d/init.d/".

Caching-Only Name Server

Các caching-only name server là những server (máy chủ) không có thẩm quyền cho bất kỳ domain (miền) nào ngoại trừ 0.0.127.in-addr.arpa (máy cục bộ). Một caching-only name server có thể tìm kiếm tên bên trong hoặc ngoài zone (vùng) của bạn, hay trong các primary và slave name server. Nhưng cái khác của caching-only name server là ban đầu nó sẽ tìm kiếm tên trong vùng của bạn và cuối cùng nó sẽ tìm trong primary hoặc slave names servers để tìm câu trả lời.

Những tập tin cần thiết để cài đặt simple caching name server là:

named.conf
db.127.0.0
db.cache
named script

Cấu hình của tập tin "/etc/named.conf" dành cho simple caching name server.

Dùng cấu hình này cho tất cả máy chủ trong mạng máy tính của bạn mà các server này không làm việc như một master hoặc slave name server. Hãy cài đặt một caching server đơn giản cho các máy client cục bộ sẽ làm giảm bớt tải

trên server chính (primary server) của mạng. Nhiều người sử dụng trên các kết nối quay số (dialup) có thể dùng cấu hình này để phục vụ cho mục đích như thế.

Tạo tập tin named.conf (touch /etc/named.conf) và thêm những dòng sau:

```
options {
directory "/var/named";
forwarders { 208.164.186.1; 208.164.186.2; };
forward only;
};
//
// a caching only nameserver config
zone "." in {
type hint;
file "db.cache";
};
zone "0.0.127.in-addr.arpa" in {
type master;
file "db.127.0.0";
};
```

Trong dòng "forwarders", 208.164.186.1 và 208.164.186.2 là những địa chỉ IP trên DNS server dành cho Primary (Master) và Secondary (Slave), chúng cũng có thể là địa chỉ IP trong DNS server của ISP hoặc DNS server khác.

Để nâng cao độ bảo mật BIND/DNS server. Nếu forwarder ngưng hoạt động hoặc không đáp ứng lại những gì ta yêu cầu, thì bạn có thể dừng máy chủ này và cho thực thi trên máy chủ off-site. Với tùy chọn "forward only" thiết lập trong tập tin "named.conf" máy chủ sẽ không tìm kiếm thông tin từ máy chủ khác khi forwarder không nhận được câu trả lời.

Cấu hình của tập tin "/var/named/db.127.0.0" cho một simple caching name server

Dùng cấu hình này cho tất cả máy chủ hiện hữu trên mạng máy tính của bạn mà không phải là một master hoặc slave name server. Tập tin "db.127.0.0" đảm nhận việc loopback trên mạng. Hãy tạo những tập tin sau trong thư mục "/var/named/".

Tạo tập tin db.127.0.0 (touch /var/named/db.127.0.0) và thêm vào những dòng sau:

```
$TTL 345600
@     IN    SOA    localhost. root.localhost. (
                        00          ; Serial
                        86400       ; Refresh
                        7200        ; Retry
                        2592000     ; Expire
                        345600 )    ; Minimum
      IN    NS     localhost.

1     IN    PTR    localhost.
```

Cấu hình của tập tin "/var/named/db.cache" dành cho một simple caching name server

Trước khi khởi động máy chủ DNS bạn phải sao chép tập tin "db.cache" vào trong thư mục "/var/named/". Tập tin "db.cache" chỉ ra những vùng "root" cho máy server của bạn biết.

Hãy dùng những lệnh sau trên máy Unix khác để tìm tập tin db.cache cho máy chủ DNS hoặc lấy tập tin này từ chương trình nguồn trên CD-ROM Red Hat Linux.

- Để tìm tập tin mới db.cache cho máy DNS của bạn, dùng lệnh sau

```
[root@deep]# dig @.aroot-servers.net . ns > db.cache
```

Đừng quên sao chép tập tin "db.cache" vào thư mục "/var/named/" trên máy của bạn tại nơi mà bạn đang cài đặt DNS server sau khi nhặt nó trên mạng Internet.

Chú thích *: Với những lí do về bảo mật, những địa chỉ bên trong mạng như 192.168.1/24 không được tính đến trong các tập tin cấu hình DNS, vì yếu tố quan trọng đó nên DNS không thể tồn tại sẵn giữa máy trong mạng và các máy bên ngoài.*

Primary master name server

Một primary master name server cho một zone (vùng) đọc dữ liệu cho zone dựa vào một tập tin trên chính máy chủ đó và có thẩm quyền trên zone đó.

Những tập tin cần thiết để cài đặt một primary master name server là:

```
named.conf
db.127.0.0
db.208.164.186
db.openna
db.cache
named script
```

Cấu hình của tập tin "/etc/named.conf" dành cho một master name server

Dùng cấu hình này cho máy chủ trên mạng, hoạt động như là một master name server. sau khi biên dịch DNS, bạn cần thiết lập một tên miền chính cho máy chủ này. Để ví dụ, chúng ta sử dụng tên miền là "openna.com" và giả sử địa chỉ IP là 208.164.186.0, để làm được việc này, thêm vào những dòng sau vào tập tin "/etc/named.conf".

Tạo tập tin named.conf (touch /etc/named.conf) và thêm:

```
options {
directory "/var/named";
fetch-glue no;
recursion no;
```

```
allow-query { 208.164.186/24; 127.0.0/8; };
allow-transfer { 208.164.186.2; };
transfer-format many-answers;
};
// These files are not specific to any zone
zone "." in {
type hint;
file "db.cache";
};
zone "0.0.127.in-addr.arpa" in {
type master;
file "db.127.0.0";
};
// These are our primary zone files
zone "openna.com" in {
type master;
file "db.openna ";
};
zone "186.164.208.in-addr.arpa" in {
type master;
file "db.208.164.186";
};
```

Tùy chọn "fetch-glue no" có thể được dùng kết hợp với tùy chọn "recursion no" để ngăn chặn việc gia tăng cache của máy chủ hoặc dẫn tới hư hỏng dữ liệu, hơn nữa, có thể làm mất tính năng tự phục hồi đặt máy chủ của bạn thành kiểu thụ động. Có nghĩa là, nó không bao giờ gởi những câu truy vấn đến máy chủ khác hoặc giải quyết câu truy vấn. Một non-recursive name server thì rất khó nhái theo. Bởi vì nó không gởi những câu truy vấn và do đó cũng không cất giữ dữ liệu nào.

Trong dòng "allow-query", 208.164.186/24 và 127.0.0/8 là những địa chỉ IP được cho phép hỏi máy chủ những vấn đề thông thường.

Trong dòng "allow-transfer", 208.164.186.2 là địa chỉ IP được cho phép nhận vùng chuyển giao. Bạn phải chắc chắn rằng chỉ có slave name servers thực sự mới có thể chuyển giao vùng từ name server của bạn. Như đã đề cập ở trên, tính năng này thường được dùng bởi spammers và spoofers IP.

Chú thích: *Tùy chọn "recursion no", "allow-query" và "allow-transfer" trong tập tin "named.conf" ở trên là những tính năng bảo mật.*

Cấu hình của tập tin "/var/named/db.127.0.0" dành cho một master và slave name server

Cả hai master và slave name server có thể dùng tập tin cấu hình này. Tập tin "db.127.0.0" đảm nhận việc loopback trên mạng. Hãy tạo những tập tin sau trong thư mục "/var/named/"

Tạo tập tin db.127.0.0 (touch /var/named/db.127.0.0) và thêm:

```
; Revision History: April 22, 1999 - admin@mail.openna.com
; Start of Authority (SOA) records.
$TTL 345600
@  IN  SOA  deep.openna.com. admin.mail.openna.com. (
                        00          ; Serial
                        86400       ; Refresh
                        7200        ; Retry
                        2592000     ; Expire
                        345600 )    ; Minimum

; Name Server (NS) records.
    NS  deep.openna.com.
    NS  mail.openna.com.

; only One PTR record.
1    PTR  localhost.
```

Cấu hình của tập tin "/var/named/db.208.164.186" dành cho một master name server

Dùng cấu hình này cho máy server trên mạng máy tính của bạn hoạt động như một master name server. Tập tin "db.208.164.186" ánh xạ tên máy thành địa chỉ. Hãy tạo các tập tin sau trong thư mục "/var/named/"

Tạo tập tin db.208.164.186 (touch /var/named/db.208.164.186) và thêm:

```
; Revision History: April 22, 1999 - admin@mail.openna.com
; Start of Authority (SOA) records.
$TTL 345600
@ IN SOA deep.openna.com. admin.mail.openna.com. (

00              ; Serial
86400           ; Refresh
7200            ; Retry
2592000         ; Expire
345600 )        ; Minimum

; Name Server (NS) records.
    NS deep.openna.com.
    NS mail.openna.com.

; Addresses Point to Canonical Names (PTR) for Reverse lookups
1 PTR deep.openna.com.
2 PTR mail.openna.com.
3 PTR www.openna.com.
```

Cấu hình của tập tin "/var/named/db.openna" dành cho một master name server

Dùng cấu hình này cho máy trên mạng máy tính của bạn hoạt động như một master name server. Tập tin "db.openna" ánh xạ tên máy thành địa chỉ. Hãy tạo các tập tin sau trong thư mục "/var/named/"

Tạo tập tin db.openna (touch /var/named/db.openna) và thêm:

```
; Revision History: April 22, 1999 - admin@mail.openna.com
; Start of Authority (SOA) records.
$TTL 345600
@  IN  SOA  deep.openna.com.  admin.mail.openna.com.  (
                        00          ; Serial
                        86400       ; Refresh
                        7200        ; Retry
                        2592000     ; Expire
                        345600 )    ; Minimum

; Name Server (NS) records.
     NS   deep.openna.com.
     NS   mail.openna.com.

; Mail Exchange (MX) records.
     MX   0  mail.openna.com.

; Address (A) records.
localhost               A       127.0.0.1
deep                    A       208.164.186.1
mail                    A       208.164.186.2
www                     A       208.164.186.3

; Aliases in Canonical Name (CNAME) records.
;www                    CNAME   deep.openna.com.
```

Cấu hình của tập tin "/var/named/db.cache" dành cho một master và slave name server

Trước khi khởi động máy chủ DNS, bạn phải sao chép tập tin "db.cache" vào trong thư mục "/var/named/". Tập tin "db.cache" chỉ ra những vùng "root" cho máy chủ của bạn biết.

Hãy dùng những lệnh sau trên máy Unix khác để tìm tập tin db.cache cho máy chủ DNS hoặc lấy tập tin này từ chương trình nguồn trên CD-ROM Red Hat Linux.

- Để tìm tập tin mới db.cache cho máy DNS của bạn, dùng lệnh sau

```
[root@deep /]# dig @.aroot-servers.net . ns > db.cache
```

Đừng quên sao chép tập tin "db.cache" vào thư mục "/var/named/" trên máy của bạn tại nơi mà bạn đang cài đặt DNS server sau khi truy cập nó từ trên mạng Internet.

Secondary slave name server

Mục đích của slave name server là chia sẻ tải thực hiện với master server, hoặc là giải quyết toàn bộ tải thực hiện khi master server bị ngừng hoạt động. Một slave name server nạp dữ liệu của nó ở trên mạng từ một server khác (thường thì master name server, nhưng nó cũng có thể nạp dữ liệu slave name server khác) tiến trình này gọi là một vùng chuyển giao.

Những tập tin cần thiết khi cài đặt một secondary slave name server là:

named.conf

db.127.0.0

db.cache

named script

Cấu hình của tập tin "/etc/named.conf" dành cho slave name server

Dùng cấu hình này cho server trên mạng máy tính của bạn hoạt động như một slave name server. Bạn phải sửa đổi tập tin "named.conf" trên máy slave name server. Thay đổi mọi sự kiện từ primary đến secondary ngoại trừ "0.0.127.in-addr.arpa" và thêm một dòng master server với địa chỉ IP của master server như trình bày dưới đây.

Tạo tập tin named.conf (touch /etc/named.conf) và thêm vào những dòng sau:

```
options {
        directory "/var/named";
        fetch-glue no;
        recursion no;
        allow-query { 208.164.186/24; 127.0.0/8; };
        allow-transfer { 208.164.186.1; };
        transfer-format many-answers;
};

// These files are not specific to any zone
zone "." in {
        type hint;
        file "db.cache";
};

zone "0.0.127.in-addr.arpa" in {
        type master;
        file "db.127.0.0";
};

// These are our slave zone files
zone "openna.com" in {
        type slave;
        file "db.openna";
        masters { 208.164.186.1; };
};

zone "186.164.208.in-addr.arpa" in {
        type slave;
        file "db.208.164.186";
        masters { 208.164.186.1; };

  };
```

Điều này nói lên rằng name server là một slave dành cho vùng "openna.com" và nên tìm phiên bản phù hợp với vùng này và lưu trữ trên máy "203.164.186.1".

Một slave name server không cần truy lục tất cả tập tin cơ sở dữ liệu của nó ở trên mạng bởi vì những tập tin "db.127.0.0" và "db.cache" đều giống nhau như trên một primary master, vì thế bạn có thể giữ lại một bản sao cục bộ của những tập tin này trên slave name server.

Sao chép tập tin "db.127.0.0" từ master name server vào slave name server.

Sao chép tập tin "db.cache" từ master name server vào slave name server.

Cấu hình của tập tin script "/etc/rc.d/init.d/named" dành cho các loại name server

Cấu hình tập tin script "/etc/rc.d/init.d/named" để khởi động và ngưng chương trình BIND/DNS daemon trên máy của bạn. Tập tin script cấu hình này có thể được dùng cho tất cả các loại name server (caching, master, hoặc slave).

Hãy tạo tập tin named (touch /etc/rc.d/init.d/named) và thêm vào :

```
    RETVAL=$?
    [ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/named
    echo
    ;;
status)
    /usr/sbin/ndc status
    exit $?
    ;;
restart)
    $0 stop
    $0 start
    ;;
reload)
    /usr/sbin/ndc reload
    exit $?
    ;;
probe)
    # named knows how to reload intelligently; we don't want linuxconf
    # to offer to restart every time
    /usr/sbin/ndc reload >/dev/null 2>&1 || echo start
    exit 0
    ;;

*)
    echo "Usage: named {start|stop|status|restart}"
    exit 1
esac

exit $RETVAL
```

```
#!/bin/sh
#
# named         This shell script takes care of starting and stopping
#               named (BIND DNS server).
#
# chkconfig: - 55 45
# description: named (BIND) is a Domain Name Server (DNS) \
# that is used to resolve host names to IP addresses.
# probe: true

# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0

[ -f /usr/sbin/named ] || exit 0

[ -f /etc/named.conf ] || exit 0

RETVAL=0

# See how we were called.
case "$1" in
  start)
        # Start daemons.
        echo -n "Starting named: "
        daemon named
        RETVAL=$?
        [ $RETVAL -eq 0 ] && touch /var/lock/subsys/named
        echo
        ;;
  stop)
        # Stop daemons.
        echo -n "Shutting down named: "
        killproc named
```

Bây giờ, tạo script có thể được thực thi này và thay đổi quyền mặc định

```
[root@deep]# chmod 700 /etc/rc.d/init.d/named
```

Hãy tạo biểu tượng liên kết rc.d cho BIND/DNS với lệnh :

```
[root@deep]# chkconfig --add named
```

Script BIND/DNS sẽ không tự thực thi daemon khi bạn khởi động lại máy chủ. Bạn có thể thay đổi việc thi hành mặc định của nó bằng lệnh sau:

```
[root@deep]# chkconfig --level 345 named on
```

Khởi động DNS server (một cách thủ công) bằng lệnh sau:

```
[root@deep]# /etc/rc.d/init.d/named start
Starting named:                                [ OK ]
```

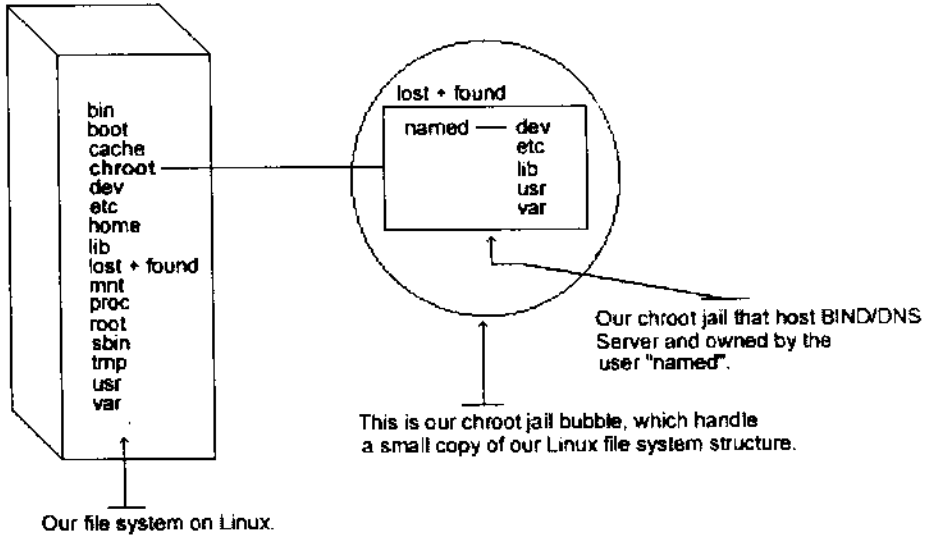
An toàn cho ISC BIND/DNS

Thực thi chương trình ISC BIND/DNS trong một chroot jail

Phần này tập trung để ngăn ngừa chương trình ISC BIND/DNS được dùng như một điểm tấn công hệ thống. Vì ISC BIND/DNS thực hiện với chức năng phức tạp và mối quan hệ khá lớn. Với phần mềm này thì khả năng về lỗi sẽ ảnh hưởng khá cao về mặt an toàn. Thật vậy, trong quá khứ, lợi dụng những lỗi này những kẻ tấn công từ xa đã (hoặc sẽ) chiếm giữ quyền truy cập root mà máy chủ đang vận hành chương trình ISC BIND/DNS.

Để giảm đến mức tối thiểu rủi ro này, ISC BIND/DNS có thể chạy chương trình như user non-root, vì nó sẽ giới hạn những phần nguy hiểm cùng những gì được thực hiện của một người dùng bình thường với một lệnh cục bộ. Tuy nhiên, điều này cũng chưa đủ cho những đòi hỏi bảo mật nhất của DNS servers, nên bước kế tiếp có thể thực hiện – đó là vận hành chương trình ISC BIND/DNS trong chroot jail.

Lợi ích chính của chroot jail là jail sẽ giới hạn phần tập tin hệ thống của chương trình DNS daemon có thể truy xuất trên thư mục root. Thêm vào đó, vì jail chỉ cần hỗ trợ cho DNS, nên những chương trình liên quan đến ISC BIND/DNS có thể cũng bị giới hạn tối đa. Quan trọng nhất, không cần những chương trình setuid-root, nó cũng có thể được dùng để dành phần truy xuất thư mục root và tách khỏi jail.



Ghi chú: *Chương trình nhị phân "named" phải được lưu trữ trong một thư mục liệt kê trong biến môi trường PATH của bạn. Phần còn lại của tài liệu này, tôi giả sử đường dẫn thư mục của chương trình gốc là "/usr/sbin/named".*

Sau đây là những bước cần thiết để chạy phần mềm ISC BIND/DNS trong chroot jail:

Bước 1

Chúng ta phải tìm thư viện dùng chung (có tên là DNS Daemon), và sau đó nó sẽ được sao chép vào chroot jail.

- Để tìm ra thư viện dùng chung, thực hiện theo bước sau :

```
[root@deep /]# ldd /usr/sbin/named
libc.so.6 => /lib/libc.so.6 (0x40017000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Hãy tạo một chú thích cho những tập tin trên; bạn sẽ cần chúng sau này trong những bước thực hiện kế tiếp.

Bước 2

Bây giờ hãy thiết lập môi trường chroot và tạo thư mục gốc cho jail. Nên chọn là "/chroot/named" bởi vì chúng tôi muốn đặt những tập tin về DNS trên những hệ thống tập tin riêng lẻ để tránh trường hợp hệ thống tập tin bị tấn công. Ngay từ đầu trong qui trình cài đặt hệ thống Linux, chúng tôi đã tạo một partition đặc biệt có tên là "/chroot" dành riêng cho mục đích này.

```
[root@deep /]# /etc/rc.d/init.d/named stop (require only if an existing named daemon is running)
Shutting down named:                [ OK ]
[root@deep /]# mkdir -p /chroot/named
```

Kế tiếp, hãy tạo phần còn lại của các thư mục như sau:

```
[root@deep /]# mkdir /chroot/named/dev
[root@deep /]# mkdir /chroot/named/lib
[root@deep /]# mkdir /chroot/named/etc
[root@deep /]# mkdir -p /chroot/named/usr/sbin
[root@deep /]# mkdir -p /chroot/named/var/run
[root@deep /]# mkdir /chroot/named/var/named
```

Bây giờ sao chép phần chính của tập tin cấu hình, những tập tin vùng, tên và chương trình named-xfer vào những vị trí tương ứng trong thư mục chroot jail.

```
[root@deep /]# cp /etc/named.conf /chroot/named/etc/
[root@deep /]# cd /var/named ; cp -a . /chroot/named/var/named/
[root@deep /]# mknod /chroot/named/dev/null c 1 3
[root@deep /]# chmod 666 /chroot/named/dev/null
```

```
[root@deep /]# cp /usr/sbin/named /chroot/named/usr/sbin/
[root@deep /]# cp /usr/sbin/named-xfer /chroot/named/usr/sbin/
```

Chú thích quan trọng: *Người sở hữu của thư mục "/chroot/named/var/named" và tất cả các tập tin trong thư mục này phải xử lý đặt tên "named" dưới slave server và chỉ có slave server mà thôi, bạn không thể nào làm một vùng chuyển giao được.*

- Để tạo thư mục "named" và tất cả tập tin của nó được xử lý bởi chính "named" đặt dưới slave server, dùng lệnh :

```
[root@deep /]# chown -R named.named /chroot/named/var/named/
```

Bước 3

Sao chép các thư mục thư viện dùng chung được định rõ ở bước 1 bên trên vào thư mục lib:

```
[root@deep /]# cp /lib/libc.so.6 /chroot/named/lib/
[root@deep /]# cp /lib/ld-linux.so.2 /chroot/named/lib/
```

Bước 4

Sao chép những tập tin "localtime" và "nsswitch.conf" vào thư mục etc của chrooted để cho những phần nhập vào được điều chỉnh đúng với múi giờ trên máy cục bộ của bạn:

```
[root@deep /]# cp /etc/localtime /chroot/named/etc/
[root@deep /]# cp /etc/nsswitch.conf /chroot/named/etc/
```

Bước 5

Chúng ta phải cài đặt một vài tập tin dưới thư mục "/chroot/named/etc" với bit không biến đổi để làm tăng tính bảo mật :

- Hãy đặt bit không biến đổi trên tập tin "nsswich.conf":

```
[root@deep /]# cd /chroot/named/etc/
[root@deep etc]# chattr +i nsswitch.conf
```

- Hãy đặt bit không biến đổi trên tập tin "named.conf":

```
[root@deep /]# cd /chroot/named/etc/
[root@deep etc]# chattr +i named.conf
```

Một tập tin với thuộc tính "+i " không thể nào được sửa đổi, xóa hoặc đổi tên; không có liên kết nào được thực hiện trên tập tin này cũng như không dữ liệu nào có thể được ghi lên đó. Chỉ có superuser có thể thiết lập hoặc xóa thuộc tính này.

Bước 6

Thêm một tài khoản UID và GID mới để thực thi chương trình daemon "named" nếu nó chưa được cài đặt sẵn. Đây là điều quan trọng vì sử dụng tài khoản root có thể làm giảm tính năng của jail, do vậy chỉ dùng user id khác đã

tồn tại sẵn trên hệ thống có thể cho phép các dịch vụ của bạn truy cập tài nguyên lẫn nhau.

Hãy kiểm tra các tập tin "/etc/passwd" và "/etc/group" cho một số tài khoản UID/GID bất kỳ có sẵn trong hệ thống. Trong ví dụ, chúng tôi sẽ cho số "53" và đặt dưới tên "named".

```
[root@deep /]# useradd -c "DNS Server" -u 53 -s /bin/false -r –d \
/chroot/named named 2>/dev/null || :
```

Bước 7

Chúng ta phải làm cho syslogd (chương trình syslog daemon) nhận biết về dịch vụ chrooted mới:

Thông thường, việc xử lý để làm cho syslogd nhận biết các dịch vụ là thông qua "/dev/log". Như trường hợp của chroot jail, điều này không thể được, vì syslogd cần phải tuân theo "/chroot/named/dev/log" thay vì phần mặc định "/dev/log". Để làm việc này, hiệu chỉnh phần khởi động tập tin syslogd chỉ rõ những nơi thêm vào.

Hiệu chỉnh tập tin syslog (vi +24 /etc/rc.d/init.d/syslog) và thay đổi dòng:

```
daemon syslogd -m 0
```

Để thành:

```
daemon syslogd -m 0 -a /chroot/named/dev/log
```

Bước 8

Mặc định tập tin script named của ISC BIND/DNS thực thi daemon "named" ở bên ngoài chroot jail. Chúng ta phải thay đổi sao cho nó thực thi trong chroot jail. Hiệu chỉnh tập tin script named (vi /etc/rc.d/init.d/named) và thay đổi những dòng :

```
[ -f /usr/sbin/named ] || exit 0
```

Để thành:

```
[ -f /chroot/named/usr/sbin/named ] || exit 0
```

```
[ -f /etc/named.conf ] || exit 0
```

Để thành:

```
[ -f /chroot/named/etc/named.conf ] || exit 0
```

```
daemon named
```

Để thành:

```
daemon /chroot/named/usr/sbin/named -t /chroot/named/ -unamed -gnamed
```

Tùy chọn "-t" option báo cho "named" khi thực thi sử dụng môi trường chroot mới

Tùy chọn "-u" chỉ định tài khoản user để thực thi

Tùy chọn "-g" chỉ định tài khoản group để thực thi

Bước 9

Trong phiên bản BIND 8.2, lệnh "ndc" của phần mềm ISC BIND/DNS đã trở thành một tập tin nhị phân (trước đây, nó là một tập tin script). Để sửa lỗi này, package ISC BIND/DNS phải được biên dịch lại từ tập tin nguồn.

Để làm việc này ở phần đầu của thư mục nguồn ISC BIND/DNS

Dành cho phần tiện ích ndc:

```
[root@deep /]# cp bind-src.tar.gz /vat/tmp
[root@deep /]# cd /var/tmp/
[root@deep tmp]# tar xzpf bind-src.tar.gz
[root@deep tmp]# cd src
[root@deep src]# cp port/linux/Makefile.set port/linux/Makefile.set-orig
```

Hiệu chỉnh tập tin Makefile.set (vi port/linux/Makefile.set) và tạo sự thay đổi được liệt kê dưới đây:

```
'CC=egcs -D_GNU_SOURCE'
'CDEBUG=-O9 -funroll-loops -ffast-math -malign-double
mcpu=pentiumpro -march=pentiumpro - fomit-frame-pointer -fno-
exceptions -g'
'DESTBIN=/usr/bin'
'DESTSBIN=/chroot/named/usr/sbin'
'DESTEXEC=/chroot/named/usr/sbin'
'DESTMAN=/usr/man'
'DESTHELP=/usr/lib'
'DESTETC=/etc'
'DESTRUN=/chroot/named/var/run'
'DESTLIB=/usr/lib/bind/lib'
'DESTINC=/usr/lib/bind/include'
'LEX=flex -8 -l'
'YACC=yacc -d'
'SYSLIBS=-lfl'
'INSTALL=install'
'MANDIR=man'
'MANROFF=cat'
'CATEXT=$$N'
'PS=ps p'
'AR=ar crus'
'RANLIB=:'
```

Điểm khác nhau giữa Makefile mà chúng ta đã dùng trước đây và bây giờ là ta sửa đổi các dòng "DESTSBIN=", "DESTEXEC=", và "DESTRUN=" để chỉ ra thư

mục chrooted của phần mềm BIND/DNS. Với sự sửa đổi này, chương trình "ndc" biết vị trí để tìm "named".

```
[root@deep src]# make clean
[root@deep src]# make
[root@deep src]# cp bin/ndc/ndc /usr/sbin/
[root@deep src]# cp: overwrite `/usr/sbin/ndc'? y
[root@deep src]# strip /usr/sbin/ndc
```

Chúng ta tạo tập tin nhị phân, sau đó sao chép kết quả chương trình "ndc" vào thư mục "/usr/sbin " và ghi chồng lên tập tin cũ. Nhớ đừng quên chạy lệnh strip với tập tin nhị phân mới "ndc" để cải tiến phần thực thi được tốt hơn.

Bước 10

Đó cũng là một ý tưởng hay để xây dựng những chương trình nhị phân mới "named" và "named-xfer" nhưng phải chắc rằng phiên bản đó phải giống như phiên bản được dùng cho hai tập tin "named" và "ndc".

Dành cho chương trình "named"

```
[root@deep /]# cd /var/tmp/src
[root@deep src]# cp port/linux/Makefile.set-orig port/linux/Makefile.set
[root@deep src]# cp: overwrite `port/linux/Makefile.set'? y
```

Hiệu chỉnh tập tin Makefile.set (vi port/linux/Makefile.set) và tạo sự thay đổi được liệt kê dưới đây:

```
CC=egcs -D_GNU_SOURCE'
'CDEBUG=-O9 -funroll-loops -ffast-math -malign-double -
mcpu=pentiumpro -march=pentiumpro -fomit-frame-pointer -fno-
exceptions -g'
'DESTBIN=/usr/bin'
'DESTSBIN=/usr/sbin'
'DESTEXEC=/usr/sbin'
'DESTMAN=/usr/man'
'DESTHELP=/usr/lib'
'DESTETC=/etc'
'DESTRUN=/var/run'
'DESTLIB=/usr/lib/bind/lib'
'DESTINC=/usr/lib/bind/include'
'LEX=flex -8 -l'
'YACC=yacc -d'
'SYSLIBS=-lfl'
'INSTALL=install'
'MANDIR=man'
'MANROFF=cat'
'CATEXT=$$N'
'PS=ps p'
'AR=ar crus'
'RANLIB=:'

[root@deep src]# rm -f .settings
```

```
[root@deep src]# make clean
[root@deep src]# make
[root@deep src]# cp bin/named/named /chroot/named/usr/sbin
[root@deep src]# cp: overwrite `/chroot/named/usr/sbin/named'? y
[root@deep src]# cp bin/named-xfer/named-xfer /chroot/named/usr/sbin
[root@deep src]# cp: overwrite `/chroot/named/usr/sbin/named-xfer'? y
[root@deep src]# strip /chroot/named/usr/sbin/named
[root@deep src]# strip /chroot/named/usr/sbin/named-xfer
```

Chúng ta phải xoá tập tin ".setting" vì quá trình cài đặt hệ thống đã cất giữ những biến này, và chạy lệnh "make clean" để chắc chắn rằng không để lại phần rác rưởi nào còn tồn tại. Sau đó, chúng ta tạo tập tin nhị phân "named" và sao chép nó cùng với tập tin "named-xfer" vào thư mục của chrooted jail. Tương tự như những lần trước, chúng ta sẽ dùng lệnh "strip" để cải tiến phần thực hiện của những tập tin nhị phân mới.

Bước 11

Gỡ bỏ những tập tin và thư mục không cần thiết.

```
[root@deep /]# rm -f /usr/sbin/named
[root@deep /]# rm -f /usr/sbin/named-xfer
[root@deep /]# rm -f /etc/named.conf
[root@deep /]# rm -rf /var/named/
```

Hãy gỡ bỏ những tập tin nhị phân "named" và "named-xfer" trong thư mục "/usr/sbin" vì những phần đó chúng ta sẽ làm việc hàng ngày và được định vị dưới thư mục chroot. Tương tự áp dụng cho tập tin "named.conf" và thư mục "/var/named".

Bước 12

Chúng ta phải kiểm tra cấu hình chrooted jail mới của phần mềm ISC BIND/DNS.

- Việc đầu tiên là khởi động lại syslogd daemon của chúng ta với lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/syslog restart
Shutting down kernel logger:                [ OK ]
Shutting down system logger:                [ OK ]
Starting system logger:                     [ OK ]
Starting kernel logger:                     [ OK ]
```

- Bây giờ, khởi động chrooted jail mới của phần mềm ISC BIND/DNS với lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/named start
Starting named:                             [ OK ]
```

Phải chắc rằng chương trình đang chạy với tài khoản user "named" và với những đối số mới:

- Để xác minh rằng ISC BIND/DNS đang chạy như tài khoản user "named" với những đối số mới, dùng lệnh sau

```
[root@deep /]# ps auxw | grep named
named 11446 0.0 1.2 2444 1580 ? S 23:09 0:00
```

```
/chroot/named/usr/sbin/named -t /chroot/named/ -unamed -gnamed
```

Cột đầu tiên phải là "named" mà UID named daemon đang vận hành bên dưới. Kết thúc của dòng sẽ là "named -t/chroot/named/ -unamed-gnamed", mà đó là những đối số mới.

Dọn dẹp sau quá trình cài đặt

```
[root@deep /]# rm -rf /var/tmp/src bind-src.tar.gz
```

Phần này sẽ gỡ bỏ tập tin nguồn và tập tin tar mà chúng ta đã dùng để biên dịch và cài đặt phần mềm ISC BIND/DNS.

Tài liệu bổ sung

Để cho chi tiết hơn, có một vài trang man mà bạn cần đọc:

```
$ man dnsdomainname (1)          - show the system's DNS domain name
$ man dnskeygen (1)              - generate public, private, and shared secret keys for
                                   DNS Security
$ man dnsquery (1)               - query domain name servers using resolver
$ man named (8)                  - Internet domain name server (DNS)
$ man hesiod_to_bind [hesiod] (3)  Hesiod name server interface library
$ man ldconfig (8)               - determine run-time link bindings
$ man lesskey (1)                - specify key bindings for less
$ man raw (8)                    - bind a Linux raw character device
$ man mkfifo (1)                 - make FIFOs (named pipes)
$ man named-bootconf (8)         - convert name server configuration files
$ man named-xfer (8)             - ancillary agent for inbound zone transfers
$ man named.conf [named] (5)     - configuration taäp tin for
$ man Opcode (3)                 - Disable named opcodes when compiling perl code
$ man dig (1)                    - send domain name query packets to name servers
$ man nslookup (8)               - query Internet name servers interactively
$ man ndc (8)                    - name daemon control program
```

Các công cụ quản trị DNS

Tập lệnh liệt kê ở dưới có một số chúng ta thường dùng, nhưng một số thì không hiện hữu. Hãy kiểm tra trang man và tài liệu để nắm thông tin chi tiết hơn về phần này.

dig

Lệnh tiện ích "dig" (domain information groper) có thể được dùng để cập nhật tập tin "db.cache" bằng cách chỉ ra những vùng "root" cho máy chủ của bạn biết. Khi máy chủ này biết được vị trí của những vùng đó, nó sẽ truy vấn tập tin "db.cache" từ đó. Tên Root của những máy chủ thường thì không thay đổi, nhưng đôi khi thì thay đổi. Một thói quen tốt là cập nhật tập tin "db.cache" sau mỗi một hoặc hai tháng.

- Dùng lệnh sau để truy vấn tập tin mới db.cache cho máy server DNS của bạn:

```
[root@deep /]# dig @.aroot-servers.net . ns > db.cache
```

Sao chép tập tin db.cache đến thư mục "/var/named/"sau khi truy lục nó.

```
[root@deep /]# cp db.cache /var/named/
```

Ở đây @aroot-servers.net là địa chỉ của máy chủ root dành để truy vấn tập tin db.cache và db.cache là tên của tập tin mới db.cache của bạn.

ndc

Lệnh tiện ích "ndc" của phần mềm ISC BIND/DNS cho phép nhà quản trị hệ thống điều khiển việc tương tác thông qua một terminal quá trình hoạt động của một name server.

- Để nhận được những thông tin trợ giúp cho lệnh ndc, đầu tiên gõ ndc trên terminal và sau đó gõ tiếp help.

```
[root@deep /]# ndc
Type help -or- /h if you need help.
ndc> help
getpid
status
stop
exec
reload [zone] ...
reconfig (just sees new/gone zones)
dumpdb
stats
trace [level]
notrace
querylog
qrylog
help
quit
ndc> /e
```

Các công cụ sử dụng DNS

Những lệnh được liệt kê ở dưới thì có một số mà chúng ta thường dùng, hãy kiểm tra man pages và tài liệu để nắm được chi tiết và nhiều thông tin hơn.

nslookup

Chương trình nslookup cho phép người dùng chất vấn những tên miền trên Internet theo loại tương tác hoặc không tương tác. Trong kiểu tương tác, người dùng có thể chất vấn tên những servers để lấy thông tin về những hosts và vùng khác nhau sau đó hiển thị danh sách của những hosts đó trong miền đó. Trong kiểu không tương tác người dùng có thể chỉ hiển thị tên và thông tin yêu cầu cho một máy hoặc một miền.

Kiểu tương tác có rất nhiều tùy chọn và lệnh, bạn nên xem qua trang man hoặc phần trợ giúp dành cho chương trình nslookup trước khi sử dụng.

- Nhập nslookup trong kiểu tương tác, dùng lệnh:

```
[root@deep /]# nslookup
Default Server: deep.openna.com
Address: 208.164.186.1
> help
$Id: nslookup.help,v 8.4 1996/10/25 18:09:41 vixie Exp $
```

```
Commands:       (identifiers are shown in uppercase, [] means optional)
NAME            - print info about the host/domain NAME using default server
NAME1 NAME2     - as above, but use NAME2 as server
help or ?       - print info on common commands; see nslookup(1) for details
set OPTION      - set an option
all             - print options, current server and host
[no]debug       - print debugging information
[no]d2          - print exhaustive debugging information
```

- Để thực thi trên kiểu không tương tác, dùng lệnh:

```
[root@deep /]# nslookup www.Red Hat.com
Server: deep.openna.com
Address: 208.164.186.1

Non-authoritative answer:
Name: www.portal.Red Hat.com
Addresses: 206.132.41.202, 206.132.41.203
Aliases: www.Red Hat.com
```

Ở đây www.Red Hat.com là tên máy hoặc địa chỉ Internet của những name server đã được tìm kiếm.

dnsquery

Chương trình dnsquery truy vấn tên miền ở các máy qua phần gọi thư viện thực thi "/etc/resolv.conf"

- Để truy vấn tên miền của các máy chủ sử dụng resolver, dùng lệnh:

```
[root@deep /]# dnsquery <host>
```

Ví dụ:

```
[root@deep /]# dnsquery www.Red Hat.com
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 40803
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 4
;; www.Red Hat.com, type = ANY, class = IN
www.Red Hat.com.      2h19m46s IN CNAME www.portal.Red Hat.com.
Red Hat.com.          2h18m13s IN NS ns.Red Hat.com.
Red Hat.com.          2h18m13s IN NS ns2.Red Hat.com.
Red Hat.com.          2h18m13s IN NS ns3.Red Hat.com.
Red Hat.com.          2h18m13s IN NS speedy.Red Hat.com.
ns.Red Hat.com.       1d2h18m8s IN A 207.175.42.153
ns2.Red Hat.com.      1d2h18m8s IN A 208.178.165.229
ns3.Red Hat.com.      1d2h18m8s IN A 206.132.41.213
speedy.Red Hat.com.   2h18m13s IN A 199.183.24.251
```

Ở đây <host> là tên của máy mà bạn muốn truy vấn

host

Chương trình host tìm kiếm những tên máy bằng cách dùng DNS

- Để tìm kiếm những tên máy sử dụng máy chủ miền, dùng lệnh:

```
[root@deep /]# host <FQDN, domain names, host names, or host numbers>
```

Ví dụ:

```
[root@deep /]# host Red Hat.com
Red Hat.com has address 207.175.42.154
```

Ở đây <FQDN, những tên miền, các tên máy, hoặc những địa chỉ IP của máy> thì hoặc là FQDN (www.Red Hat.com), những tên miền (Red Hat.com), các tên máy (www) hoặc địa chỉ IP của máy (207.175.42.154)

- Để tìm tất cả thông tin về một máy được duy trì bởi DNS, dùng lệnh:

```
[root@deep /]# host <-a domain names >
```

Ví dụ

```
[root@deep /]# host -a Red Hat.com
Trying null domain
rcode = 0 (Success), ancount=6
The following answer is not authoritative:
The following answer is not verified as authentic by the server:
Red Hat.com     8112 IN NS      ns.Red Hat.com
Red Hat.com     8112 IN NS      ns2.Red Hat.com
Red Hat.com     8112 IN NS      ns3.Red Hat.com
Red Hat.com     8112 IN NS      speedy.Red Hat.com
Red Hat.com     8112 IN A       207.175.42.154
Red Hat.com     11891 IN        SOA     ns.Red Hat.com noc.Red Hat.com(
                2000021402 ;serial (version)
                3600 ;refresh period
                1800 ;retry refresh this often
                604800 ;expiration period
                86400 ;minimum TTL
                )
For authoritative answers, see:
Red Hat.com     8112 IN NS      ns.Red Hat.com
Red Hat.com     8112 IN NS      ns2.Red Hat.com
Red Hat.com     8112 IN NS      ns3.Red Hat.com
Red Hat.com     8112 IN NS      speedy.Red Hat.com
Additional information:
ns.Red Hat.com  94507   IN      A       207.175.42.153
ns2.Red Hat.com         94507   IN      A       208.178.165.229
ns3.Red Hat.com         94507   IN      A       206.132.41.213
speedy.Red Hat.com  8112        IN      A       199.183.24.251
```

Tùy chọn này cũng có thể dùng để tìm tất cả thông tin được duy trì bởi máy chủ tên miền (domain server) về máy này, trong ví dụ của chúng ta "Red Hat.com"

- Để liệt kê toàn bộ thông tin về một miền, dùng lệnh:

```
[root@deep /]# host <-l domain names >
```

Ví dụ

```
[root@deep /]# host -l openna.com
openna.com name server deep.openna.com
openna.com name server mail.openna.com
localhost.openna.com has address 127.0.0.1
deep.openna.com has address 208.164.186.1
mail.openna.com has address 208.164.186.2
```

www.openna.com has address 208.164.186.3

Tùy chọn này, sẽ tải xuống một vùng cơ sở dữ liệu đầy đủ của tên miền cho domain "openna.com". Lệnh này sẽ được dùng chỉ khi thật sự cần thiết vì có thể bạn sẽ nhận một kết quả rất lớn.

Các tập tin đã cài đặt:

> /etc/rc.d/init.d/named
> /etc/rc.d/rc0.d/K45named
> /etc/rc.d/rc1.d/K45named
> /etc/rc.d/rc2.d/K45named
> /etc/rc.d/rc3.d/K45named
> /etc/rc.d/rc4.d/K45named
> /etc/rc.d/rc5.d/K45named
> /etc/rc.d/rc6.d/K45named
> /etc/named.conf
> /usr/bin/addr
> /usr/bin/nslookup
> /usr/bin/dig
> /usr/bin/dnsquery
> /usr/bin/host
> /usr/bin/nsupdate
> /usr/bin/mkservdb
> /usr/lib/bind
> /usr/lib/bind/include
> /usr/lib/bind/include/arpa
> /usr/lib/bind/include/arpa/inet.h
> /usr/lib/bind/include/arpa/nameser.h
> /usr/lib/bind/include/arpa/nameser_compat.h
> /usr/lib/bind/include/isc
> /usr/lib/bind/include/isc/eventlib.h
> /usr/lib/bind/include/isc/misc.h
> /usr/lib/bind/include/isc/tree.h
> /usr/lib/bind/include/isc/logging.h
> /usr/lib/bind/include/isc/heap.h
> /usr/lib/bind/include/isc/memcluster.h
> /usr/lib/bind/include/isc/assertions.h
> /usr/lib/bind/include/isc/list.h
> /usr/lib/bind/include/isc/dst.h
> /usr/lib/bind/include/isc/irpmarshall.h
> /usr/lib/bind/include/netdb.h
> /usr/lib/bind/include/resolv.h
> /usr/lib/bind/include/res_update.h
> /usr/lib/bind/include/irs.h
> /usr/lib/bind/include/irp.h
> /usr/lib/bind/include/hesiod.h
> /usr/lib/bind/include/sys
> /usr/lib/bind/include/net
> /usr/lib/bind/lib
> /usr/lib/bind/lib/libbind.a
> /usr/lib/bind/lib/libbind_r.a
> /usr/lib/nslookup.help
> /usr/man/man1/dig.1
> /usr/man/man1/host.1
> /usr/man/man1/dnsquery.1
> /usr/man/man1/dnskeygen.1
> /usr/man/man3/hesiod.3
> /usr/man/man3/gethostbyname.3
> /usr/man/man3/inet_cidr.3
> /usr/man/man3/resolver.3
> /usr/man/man3/getnetent.3
> /usr/man/man3/tsig.3
> /usr/man/man3/getaddrinfo.3
> /usr/man/man3/getipnodebyname.3
> /usr/man/man5/resolver.5
> /usr/man/man5/irs.conf.5
> /usr/man/man5/named.conf.5
> /usr/man/man7/hostname.7
> /usr/man/man7/mailaddr.7
> /usr/man/man8/named.8
> /usr/man/man8/ndc.8
> /usr/man/man8/named-xfer.8
> /usr/man/man8/named-bootconf.8
> /usr/man/man8/nslookup.8
> /usr/man/man8/nsupdate.8
> /usr/sbin/ndc
> /usr/sbin/named
> /usr/sbin/named-xfer
> /usr/sbin/irpd
> /usr/sbin/dnskeygen
> /usr/sbin/named-bootconf
> /var/named

Chương 15:

SERVER SOFTWARE

(Dịch vụ Mail Network)

Trong chương này chúng ta sẽ đề cập đến các phần sau:

- Linux Sendmail Server.

  Cấu hình.

  Bảo mật Sendmail.

  Các công cụ quản trị Sendmail.

  Các công cụ sử dụng Sendmail.

- Linux Imap & Pop Server.

  Cấu hình.

  Cho phép dịch vụ IMAP or POP thông qua tcp-wrappers inetd super server.

  Bảo mật IMAP/POP.



Linux sendmail server

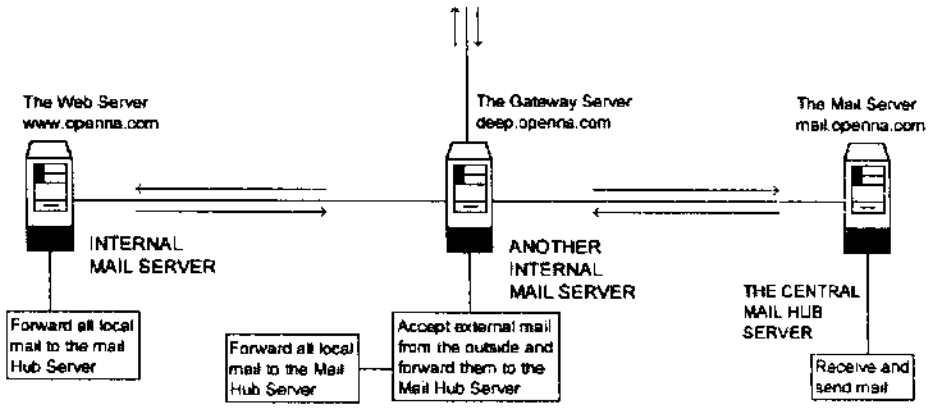
Tổng quan

Sendmail là một trong những chương trình truyền tải thư tín điện tử (Internet Mail Transport Agents (MTAs)) được sử dụng rộng rãi nhất trên thế giới. Mục đích của MTA chỉ là gởi thư tín từ máy này đến máy khác. Chương trình Sendmail không phải là một chương trình gởi nhận thư tín dành cho máy khách, mà bạn dùng để gởi nhận thư. Thay vào đó, thực sự chương trình này truyền những thư tín của bạn qua mạng vi tính hoặc mạng Internet đến những nơi theo ý muốn của bạn. Trong quá khứ, Sendmail đã từng là hệ thống dễ dàng cho bọn tội phạm khai thác, nhưng hiện nay với sự ra đời của phiên bản 8, thì việc khai thác của bọn tội phạm trở nên khó khăn hơn.

Chúng tôi sẽ cung cấp cho bạn hai cách cài đặt và cấu hình khác nhau mà bạn có thể thiết lập cho Sendmail; một là cho kiểu Central Mail Hub Relay, và hai là cho kiểu nội bộ (local) hoặc quan hệ theo kiểu client và server (Neighbor Client & Servers).

Cấu hình Central Mail Hub Relay Server sẽ được dùng cho máy chủ có nhiệm vụ gởi, nhận và chuyển thư tín cho các máy mail server và client nội bộ (local) và láng giềng (neighbor) bạn có thể có trong mạng. Một local hoặc neighbor client đề cập tới tất cả các máy client hoặc server trong mạng mà sendmail đang chạy và gởi tất cả mail ra ngoài đến máy chủ Central Mail Hub Relay để phân phối. Loại máy client cục bộ thì không bao giờ nhận thư trực tiếp từ mạng Internet, thay vào đó, tất cả thư của các máy tính này từ mạng Internet được giữ lại trên máy chủ Mail Hub. Đó là một ý tưởng hay để chạy một máy chủ Mail Hub cho tất cả các máy tính trong mạng để gởi và nhận thư. Cấu trúc này sẽ giới hạn bớt công việc quản lý trên server cũng như client và giúp tăng cường tính bảo mật cho mạng máy tính của bạn.

Bạn có thể cấu hình neighbor Sendmail để nó chỉ chấp nhận thư, mà những thư này được phát sinh cục bộ, theo cách này cô lập những máy neighbor cho việc bảo mật sẽ dễ dàng hơn. Gateway server (bên ngoài bức tường lửa hoặc một phần của bức tường lửa) hoạt động như một máy uỷ quyền (proxy) có nhiệm vụ nhận thư từ bên ngoài (thông qua các quy luật của bức tường lửa) chuyển vào phân phối bên trong mạng, và tiếp theo là chuyển đến máy chủ Central Mail Hub. Cũng cần lưu ý rằng Gateway server được định cấu hình giống như máy chủ neighbor Sendmail, để nó không bao giờ chấp nhận thư gởi đến từ bên ngoài (mạng Internet)



Phía trên là hình ảnh minh họa cho cấu hình Sendmail mà chúng ta dùng trong sách này, chúng tôi sẽ cố gắng đưa ra cho bạn những cách cài đặt khác nhau (Central Mail Hub Relay, local or neighbor client & servers) trên những máy chủ khác nhau. Có thể bớt hay thêm các chương trình, tất cả phụ thuộc vào nhu cầu và cấu trúc mạng của bạn.

Hướng dẫn cài đặt

Những lệnh này thì tương thích với hệ điều hành Unix.

Đường dẫn nguồn sẽ là "/var/tmp" (cũng có thể chọn những đường dẫn khác).

Sự cài đặt này đã được kiểm tra trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước quá trình cài đặt sẽ dùng tài khoản của super-user root .

Phiên bản Sendmail là 8.10.1.

Gói (packages)

Trang chủ của Sendmail : **http://www.sendmail.org/**

FTP Site của Sendmail: 204.152.184.34

Bạn phải chắc chắn đã chép xuống :sendmail.8.10.1.tar.gz

Tarballs

Đây là một ý tưởng hay để tạo một danh sách của các tập tin trên hệ thống trước khi bạn cài đặt Sendmail và một sau khi cài đặt, và kế đó so sánh chúng bằng cách dùng lệnh "diff" để tìm ra tập tin nào được đặt trong hệ thống. Chỉ đơn giản thực hiện lệnh 'find /* > sendmail1' trước và 'find /* > sendmail2' sau khi bạn cài đặt phần mềm và dùng lệnh 'diff sendmail1 sendmail2 > Sendmail-installed' để lấy được danh sách của các tập tin đã được thay đổi.

Biên dịch

Giải nén tập tin tarball (tar.gz)

```
[root@deep /]#cp sendmail.version.tar.gz /var/tmp
[root@deep/]#cd /var/tmp
[root@deep tmp]#tar xzpf sendmail.version.tar.gz
```

Di chuyển vào thư mục mới của Sendmail vừa mới tạo trong /var/tmp và thực hiện theo những bước sau:

Bước 1

Hiệu chỉnh tập tin smrsh.c (vi +77 smrsh/smrsh.c) và thay đổi dòng:

```
#define CMDDIR    "/usr/adm/sm.bin"
```

Để thành

```
#define CMDDIR    "/etc/smrsh"
```

Với sự thay đổi này, định rõ đường dẫn tìm kiếm mặc định cho những lệnh thực thi bởi chương trình "smrsh". Nó cho phép chúng ta giới hạn vị trí lưu giữ của chương trình.

Biên dịch và tối ưu

Bước 1:

Tập tin script Build của Sendmail sử dụng mặc định một tập tin cấu hình tương ứng với loại hệ điều hành của bạn, để thu nhận những thông tin về các định nghĩa phục vụ cho việc cài đặt hệ thống và những giá trị biên dịch khác. Tập tin này được lưu dưới thư mục con có tên là "devtools/OS" và nếu bạn muốn thực thi trên hệ điều hành Linux, nó sẽ được đặt tên là "Linux". Chúng ta sẽ thiết lập lại vị trí của tập tin cấu hình này sao cho nó phù hợp với tiến trình cài đặt và mặc định đặt nó trong thư mục con "devtools/OS" của thư mục nguồn Sendmail, từ đó tập chỉ thị Build của Sendmail mặc định sẽ tìm kiếm vị trí của tập tin cấu hình trong thư mục này trong suốt quá trình biên dịch của Sendmail.

Hãy di chuyển vào thư mục mới của Sendmail, hiệu chỉnh tập tin Linux (vi devtools/OS/Linux), và xoá tất cả các dòng ban đầu sau đó thêm những dòng dưới đây vào trong tập tin :

```
define(`confENVDEF', `-DPICKY_QF_NAME_CHECK -DXDEBUG=0')
define(`confCC', `egcs')
define(`confOPTIMIZE', `-O9 -funroll-loops -mcpu=pentiumpro –
march=pentiumpro -fomit-frame-pointer -fno-exceptions')
define(`confLIBS', `-lnsl')
define(`confLDOPTS', `-s')
define(`confMANROOT', `/usr/man/man')
define(`confMANOWN', `root')
define(`confMANGRP', `root')
define(`confMANMODE', `644')
define(`confMAN1SRC', `1')
define(`confMAN5SRC', `5')
```

```
define(`confMAN8SRC', `8')
define(`confDEPEND_TYPE', `CC-M')
define(`confNO_HELPFILE_INSTALL')
define(`confSBINGRP', `root')
define(`confSBINMODE', `6755')
define(`confUBINOWN', `root')
define(`confUBINGRP', `root')
define(`confEBINDIR', `/usr/sbin')
```

Điều này nói lên rằng tập tin Linux tự thiết lập những cấu hình đặc trưng:

```
define(`confENVDEF', `-DPICKY_QF_NAME_CHECK -DXDEBUG=0')
```

Tùy chọn macro này được dùng chính để chỉ ra phần mã có nhiệm vụ đặc biệt hoặc là bao gồm hết thảy hoặc là loại ra tất cả. Với định nghĩa "-DPICKY_QF_NAME_CHECK" Sendmail sẽ ghi một lỗi nếu tên của tập tin "qf" không đúng định dạng và sẽ đổi tên tập tin "qf" thành tập tin "QF". Đối số "-DXDEBUG=0" sẽ vô hiệu hóa những bước bổ sung vào bên trong được kiểm tra trong suốt quá trình biên dịch.

define(`confCC', `egcs')

Tùy chọn macro này định nghĩa trình biên dịch C được dùng để biên dịch Sendmail. Trong trường hợp này chúng ta dùng trình biên dịch C "egcs" cho tối ưu hơn.

```
define(`confOPTIMIZE', `-O9 -funroll-loops -mcpu=pentiumpro -
march=pentiumpro -fomit-framepointer -fno-exceptions')
```

Tùy chọn macro này định nghĩa cờ hiệu chuyển đến CC cho sự tối ưu liên quan đến cấu trúc đặc trưng của CPU.

define(`confLIBS', `-lnsl')

Tùy chọn macro này định nghĩa những cờ hiệu -l hợp qui cách ld.

define(`confLDOPTS', `-s')

Tùy chọn macro này định nghĩa những tùy chọn thiết bị kết nối chuyển đến ld.

define(`confMANROOT', `/usr/man/man')

Tùy chọn macro này định nghĩa vị trí cài đặt những trang man của Sendmail.

define(`confMANOWN', `root')

Tùy chọn macro này định nghĩa người sở hữu cho tất cả những trang man đã cài đặt của Sendmail.

define(`confMANGRP', `root')

Tùy chọn macro này định nghĩa nhóm cho tất cả những trang man đã cài đặt của Sendmail.

define(`confMANMODE', `644')

Tùy chọn macro này định nghĩa kiểu cho tất cả những trang man đã cài đặt của Sendmail.

define(`confMAN1SRC', `1')

Tùy chọn macro này định nghĩa chương trình nguồn (source) cho tất cả những trang man đã cài đặt trong confMAN1.

define(`confMAN5SRC', `5')

Tùy chọn macro này định nghĩa chương trình nguồn (source) cho tất cả những trang man đã cài đặt trong confMAN5.

define(`confMAN8SRC', `8')

Tùy chọn macro này định nghĩa chương trình nguồn (source) cho tất cả những trang man đã cài đặt trong confMAN8.

define(`confDEPEND_TYPE', `CC-M')

Tùy chọn macro này chỉ rõ làm thế nào xây dựng các phần phụ thuộc với Sendmail.

define(`confNO_HELPFILE_INSTALL')

Tùy chọn macro này mặc định chỉ rõ không cài đặt các tập tin trợ giúp cho Sendmail. Để bảo mật tốt hơn, một vài quản trị viên kinh nghiệm đã khuyến cáo như vậy.

define(`confSBINGRP', `root')

Tùy chọn macro này định nghĩa nhóm cho tất cả Sendmail setuid binaries.

define(`confSBINMODE', `6755')

Tùy chọn macro này định nghĩa kiểu cho tất cả Sendmail setuid binaries.

define(`confUBINOWN', `root')

Tùy chọn macro này định nghĩa chủ quyền cho Sendmail binaries.

define(`confUBINGRP', `root')

Tùy chọn macro này định nghĩa nhóm cho Sendmail binaries.

define(`confEBINDIR', `/usr/sbin')

Tùy chọn macro này định nghĩa nơi để cài đặt các chương trình nhị phân (binaries) được thực thi từ các chương trình nhị phân khác. Trên Red Hat Linux, đường dẫn phải được đặt cho thư mục "/usr/sbin"

Bước 2

Bây giờ chúng ta biên dịch và cài đặt Sendmail cho server:

```
[root@deep sendmail-8.10.1]# cd sendmail
[root@deep sendmail]# sh Build
[root@deep sendmail]# sh Build install
[root@deep sendmail]# cd ..
[root@deep sendmail-8.10.1]# cd mailstats
[root@deep mailstats]# sh Build install
[root@deep mailstats]# cd ..
[root@deep sendmail-8.10.1]# cd smrsh
[root@deep smrsh]# sh Build install
[root@deep smrsh]# cd ..
[root@deep sendmail-8.10.1]# cd makemap
[root@deep makemap]# sh Build install
[root@deep makemap]# cd ..
[root@deep sendmail-8.10.1]# cd praliases
[root@deep praliases]# sh Build install
[root@deep praliases]# cd ..
[root@deep sendmail-8.10.1]# ln -fs /usr/sbin/sendmail /usr/lib/sendmail
[root@deep sendmail-8.10.1]# chmod 511 /usr/sbin/smrsh
[root@deep sendmail-8.10.1]# install -d -m 755 /var/spool/mqueue
[root@deep sendmail-8.10.1]# chown root.mail /var/spool/mqueue
[root@deep sendmail-8.10.1]# mkdir /etc/smrsh
```

Trước khi cài đặt cho hệ thống của bạn, Sendmail đòi hỏi thực hiện lệnh "**sh Build**". Lệnh này có thể xây dựng và tạo những phần phụ thuộc cần thiết cho các tập tin nhị phân khác nhau được yêu cầu bởi sendmail trước khi cài đặt trên hệ thống.

Lệnh "sh Build install" sẽ cài đặt sendmail, mailstats, makemap, praliases, smrsh cũng tương ứng với những trang man trên hệ thống của bạn nếu biên dịch với lệnh này.

Lệnh "ln -fs" sẽ làm một nối kết tượng trưng của Sendmail nhị phân đến thư mục "/usr/lib". Điều này được đòi hỏi, vì lẽ rằng một vài chương trình hy vọng tìm ra được nhị phân Sendmail trong thư mục đó (/usr/lib).

Lệnh "install" sẽ tạo thư mục con "mqueue" có quyền 755 trong thư mục "/var/spool". Với nhiều lý do khác nhau một thông điệp thư tín tạm thời không được phân phối đi. Để chắc rằng những thư tín nhấn này được phân phối sau cùng,

Sendmail tạm cất giữ chúng trong hàng đợi cho tới khi nào chúng được phân phối đi một cách thành công

Lệnh "chown" sẽ thiết lập quyền UID "root" và quyền GID "mail" cho thư mục "mqueue".

Lệnh "mkdir" sẽ tạo thư mục "/etc/smrsh" trên hệ thống của bạn. Thư mục này là nơi mà chúng ta sẽ đặt tất cả chương trình mailer để cho phép Sendmail có thể thực thi được.

Chú thích: *Chương trình "makemap", và "praliases" chỉ được cài đặt trên máy chủ Central Mail Hub. "makemap" cho phép bạn tạo nên một lược đồ cơ sở dữ liệu giống với tập tin "/etc/mail/aliases.db" hoặc là "/etc/mail/access.db" cho sendmail. "praliases" trình bày hệ thống thư bí danh (mail aliases) (có nội dung trong tập tin /etc/mail/aliases). Vì rằng sẽ tốt hơn nếu chỉ sử dụng 1 vị trí (như Central Mail Hub của chúng ta) để điều khiển và quản lý tất cả các tập tin db trong mạng máy tính của chúng ta, nên không cần sử dụng hai chương trình "makemap" và "praliases" và xây dựng các tập tin db trên những máy khác trong mạng máy tính.*

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy một Central Mail Hub Server, các tập tin sau đòi hỏi phải được tạo hoặc sao chép đến các thư mục thích hợp trên máy chủ của bạn.

  Sao chép tập tin Sendmail trong thư mục "/etc/sysconfig"

  Sao chép tập tin Sendmail script trong thư mục "/etc/rc.d/init.d/"

  Sao chép tập tin local-host-names trong thư mục "/etc/mail"

  Sao chép tập tin access trong thư mục "/etc/mail"

  Sao chép tập tin aliases trong thư mục "/etc/mail".

  Tạo các tập tin virtusertable, domaintable, mailertable và .db trong thư mục "/etc/mail".

- Để chạy một Server, Local hoặc Neighbor Client, các tập tin sau đòi hỏi phải được tạo hoặc sao chép vào thư mục thích hợp trên máy chủ của bạn.

  Sao chép tập tin sendmail trong thư mục "/etc/sysconfig"

  Sao chép tập tin sendmail script trong thư mục "/etc/rc.d/init.d/"

  Sao chép tập tin tên máy cục bộ trong thư mục "/etc/mail"

Bạn có thể nhận được các tập tin cấu hình liệt kê dưới đây dựa trên tập tin floppy.tgz, sao chép các tập tin sau từ việc giải nén tập tin floppy.tgz đến những

nơi thích hợp hoặc sao chép và cắt dán chúng trực tiếp từ quyển sách này đến tập tin có liên quan.

Tập tin "/etc/mail/sendmail.mc" cho Central Mail Hub

Thay vì mỗi máy chủ riêng lẻ hoặc máy trạm trong mạng điều khiển chính những thư tín đó, máy chủ trung tâm điều khiển tất cả thư tín sẽ thuận lợi hơn. Một máy chủ như thế được gọi là Mail Hub. Sự tiện lợi của một Central Mail Hub là:

- Tất cả các thư đều được gởi đến Hub, và không một thư nào được gởi trực tiếp đến máy khách.
- Tất cả các thư gởi đi từ máy khách thì đều được gởi đến Hub và sau đó Hub sẽ gởi những thư này đến địa chỉ cần gởi.
- Tất cả các thư gởi đi bắt nguồn từ một máy chủ đơn và không cần biết đến tên client thì được hiểu như là gởi ra ngoài
- Máy khách thì không cần chạy một chương trình Sendmail daemon để nhận mail.

"sendmail.cf" là tập tin đầu tiên được đọc bởi sendmail khi nó chạy và là một trong những phần quan trọng nhất của sendmail. Nhiều mục được lưu trữ trong tập tin này là vị trí của các tập tin khác, các quyền mặc định cho những tập tin này và các thư mục mà sendrmail cần. Chương trình macro m4 của Linux được dùng bởi sendmail V8 để tạo ra một tập tin cấu hình sendmail. Chương trình macro này sẽ tạo ra tập tin cấu hình "/etc/mail/sendmail.cf" bằng cách xử lý tập tin có phần mở rộng là ".mc". Với lí do này, chúng ta sẽ tạo tập tin (sendmail.mc) và đặt những giá trị macro cần thiết vào để cho phép chương trình m4 xử lý (đọc) dữ liệu nhập vào và các định nghĩa của macro, và sau đó thay thế những macro này với những giá trị của nó và xuất kết quả này ra ngoài để tạo tập tin "sendmail.cf". Để có nhiều thông tin, bạn nên tham khảo tài liệu sendmail và tập tin README dưới thư mục con "cf" của V8 sendmail.

Bước 1

Hãy tạo tập tin sendmail.mc (touch /var/tmp/sendmail-version/cf/cf/sendmail.mc) và thêm vào những dòng lệnh sau

```
define(`confDEF_USER_ID',``8:12")dnl
OSTYPE(`linux')dnl
DOMAIN(`generic')dnl
define(`confTRY_NULL_MX_LIST',true)dnl
define(`confDONT_PROBE_INTERFACES',true)dnl
define(`PROCMAIL_MAILER_PATH',`/usr/bin/procmail')dnl
define(`LOCAL_MAILER_FLAGS', `ShPfn')dnl
define(`LOCAL_MAILER_ARGS', `procmail -a $h -d $u')dnl
FEATURE(`smrsh',`/usr/sbin/smrsh')dnl
FEATURE(`mailertable')dnl
```

```
FEATURE(`virtusertable',`hash -o /etc/mail/virtusertable')dnl
FEATURE(`redirect')dnl
FEATURE(`always_add_domain')dnl
FEATURE(`use_cw_file')dnl
FEATURE(`local_procmail')dnl
FEATURE(`access_db')dnl
FEATURE(`blacklist_recipients')dnl
FEATURE(`dnsbl')dnl
MAILER(`local')dnl
MAILER(`smtp')dnl
MAILER(`procmail')dnl
```

Điều này nói lên rằng tập tin sendmail.mc tự thiết lập những cấu hình đặc trưng với:

define(`confDEF_USER_ID',``8:12'')dnl

Tùy chọn của cấu hình này chỉ rõ user id mặc định. Trong trường hợp chúng ta dùng user là "mail" và nhóm là "mail", thì tương ứng với số ID là "8:12" (xem tập tin /etc/passwd và /etc/group) .

OSTYPE(`linux')dnl

Tùy chọn cấu hình này chỉ rõ hệ điều hành mặc định của Sendmail sẽ được chạy trên đó; trong trường hợp chúng ta dùng hệ thống "linux". Đây là phần mẩu tin nhỏ được yêu cầu bởi tập tin "mc".

DOMAIN(`generic')dnl

Tùy chọn cấu hình này sẽ chỉ định và mô tả một cách tỉ mỉ miền (domain) tương ứng với môi trường của bạn.

define(`confTRY_NULL_MX_LIST',true)dnl

Tùy chọn cấu hình này chỉ định liệu máy chủ nhận thư là MX tốt nhất cho một máy hay không và nếu được như vậy thì cố gắng kết nối trực tiếp với máy chủ đó.

define(`confDONT_PROBE_INTERFACES',true)dnl

Tùy chọn cấu hình này, nếu thiết đặt đúng (true), có nghĩa là Sendmail sẽ không được chèn những tên và địa chỉ của bất kỳ những giao tiếp cục bộ vào trong $=w class (danh sách các địa chỉ "tương đương").

define(`PROCMAIL_MAILER_PATH',`/usr/bin/procmail')dnl

Tùy chọn cấu hình này thiết lập đường dẫn để cài đặt chương trình Procmail trong máy chủ của bạn. Vì đường dẫn trong Red Hat Linux thì khác nhau giữa các phiên bản, nên chúng ta phải chỉ ra rõ đường dẫn mới cho macro này. Chú thích này quan trọng là vì macro này cũng được dùng bởi FEATURE(`local_procmail') như được định nghĩa sau trong tập tin này.

define(`LOCAL_MAILER_FLAGS', `ShPfn')dnl

Định nghĩa tùy chọn của cấu hình này xác định các cờ hiệu được dùng bởi mailer cục bộ (procmail). Để biết thêm chi tiết xin xem qua tài liệu của Sendmai.

define(`LOCAL_MAILER_ARGS', `procmail -a $h -d $u')dnl

Định nghĩa tùy chọn của cấu hình này xác định những đối số phải được truyền đến mailer cục bộ (procmail). Để biết thêm chi tiết xin xem qua tài liệu của Sendmai.

FEATURE(`smrsh',`/usr/sbin/smrsh')dnl

Macro m4 này cho phép sử dụng "smrsh" (shell hạn chế trong Sendmail) thay vì sử dụng mặc định là "/bin/sh" cho việc gởi nhận thư. Với tính năng này bạn có thể kiểm soát được tất cả những chương trình chạy qua thư điện tử, thông qua những tập tin "/etc/mail/aliases" và "~/.forward". Vị trí mặc định của chương trình "smrsh" là "/usr/libexec/smrsh". Khi chúng ta cài đặt "smrsh" ở vị trí khác, chúng ta cần phải thêm một đối số vào chương trình smrsh để ra vị trí mới là "/usr/sbin/smrsh". Cách dùng "smrsh" được giới thiệu bởi CERT, vì thế bạn có thể mạnh dạn sử dụng tính năng này càng nhiều lần càng tốt.

FEATURE(`mailertable')dnl

Macro m4 này cho phép sử dụng "mailertable" (cơ sở dữ liệu được chọn dành cho tác nhân phân phối mới). Một mailertable là một cơ sở dữ liệu ánh xạ những tên "host.domain" thành tác nhân phân phối và cặp tên miền mới. Với tính năng này, thư tín có thể được phân phát đến những vùng chỉ định rõ hoặc tác nhân phân phối riêng lẻ để đến một miền mới. Thường thì, tính năng này chỉ có trên máy chủ Central Mail Hub.

FEATURE(`virtusertable',`hash -o /etc/mail/virtusertable')dnl

Macro m4 này cho phép sử dụng "virtusertable" (hỗ trợ cho những miền ảo), cho phép tạo nhiều miền ảo trên một máy. Một virtusertable là một cơ sở dữ liệu ánh xạ những miền ảo thành những địa chỉ mới. Với tính năng này, thư tín cho những vùng ảo có thể được phân bố đến những máy cục bộ, từ xa, hoặc địa chỉ của người dùng. Thông thường tính năng này chỉ ở trên máy chủ Central Mail Hub.

FEATURE(`redirect')dnl

Macro m4 này cho phép sử dụng tập tin của "redirect" (hỗ trợ cho địa chỉ .REDIRECT). với tính năng này, thư tín để địa chỉ đến một tài khoản người dùng ít sử dụng là "wahib". Những tài khoản ít sử dụng này phải được thiết lập trong tập tin aliases ở trên mail server. Thông thường tính năng này chỉ có trên máy chủ Central Mail Hub.

FEATURE(`always_add_domain')dnl

Macro m4 này cho phép sử dụng "always_add_domain" (thêm vào miền cục bộ ngang với thư cục bộ). Với tính năng này, tất cả những địa chỉ mail được phần

phối nội bộ sẽ có tên domain đầy đủ. Đây là một trong những tính năng an toàn và được đề nghị thiết lập đặt trưng này cho các nguyên nhân bảo mật.

FEATURE(`use_cw_file')dnl

Macro m4 này cho phép sử dụng "use_cw_file" (sử dụng tập tin /ect/mail/local-host-names cho những tên máy cục bộ). Với tính năng này bạn có thể khai báo một danh sách của các máy chủ trong tập tin "/etc/mail/local-host-names" mà máy nội bộ nào hành động như một người nhận MX. Nói cách khác đặc tính này khiến cho tập tin "/ect/mail/local-host-names" được đọc luân phiên các tên trên máy chủ cục bộ.

FEATURE(`local_procmail')dnl

Macro m4 này cho phép sử dụng "local_procmail" (dùng procmail như là tác nhân phân phát cục bộ). Với tính năng này bạn có thể dùng procmail như là tác nhân phân phát Sendmail.

FEATURE(`access_db')dnl

Macro m4 này dùng để truy cập tính năng của cơ sở dữ liệu. Với đặc tính này bạn có khả năng truy cập db để cho phép hoặc từ chối chấp nhận những thư xuất phát từ những miền được chỉ định. Thông thường tính năng này chỉ có sẵn trên máy chủ Central Mail Hub.

FEATURE(`blacklist_recipients')dnl

Macro m4 này dùng để ngăn chặn thư tín đến cho tên của người dùng, máy chủ, hoặc địa chỉ mail nào đó. Với tính năng này bạn có thể ngăn chặn không ai được gởi thư đến user nobody, máy chủ foo.mydomain.com, hoặc guest@bar.mydomain.com.

FEATURE(`dnsbl')dnl

Marco m4 cho phép Sendmail từ chối mail từ bất kỳ site nào trong danh sách Realtime blackhole database (cơ sở dữ liệu bị đánh dấu đen) "rbl.maps.vix.com". DNS căn cứ vào sự từ chối là một cơ sở dữ liệu được duy trì trong DNS của spammers. Để chi tiết hơn, xin xem qua http://maps.vix.com/rbl/

MAILER(`local'), MAILER(`smtp'), and MAILER(`procmail')dnl

Marco m4 này cho phép sử dụng "local", "smtp",và "procmail" như tác nhân phân phối (mặc định trong Sendmail, tác nhân phân phối không được khai báo một cách tự động). Với tính năng này bạn có thể chỉ rõ mục nào bạn muốn hỗ trợ và mục nào thì không. Các tùy chọn MAILER(`local'), MAILER(`smtp'), and MAILER(`procmail') sẽ hỗ trợ cho local, smtp, esmtp, smtp8 và bao gồm cả procmail trả lời cho tác nhân phân phối. Đây là lưu ý quan trọng rằng MAILER(`smtp') luôn luôn đứng trước MAILER(`procmail').

Chú thích: *Đôi khi, có một miền nào đó mà bạn muốn tiếp tục truyền tải có thể kết thúc trong danh sách RBL. Trong trường hợp này, Sendmail cho*

phép ghi đè lên những miền đó để thư điện tử có thể nhận được. Để làm được việc đó, đơn giản là chỉ hiệu chỉnh lại tập tin "/etc/mail/access"và thêm vào những thông tin của miền tương ứng.

Ví dụ:

Blacklisted.domain OK

Bước 2

Bây giờ tập tin cấu hình "sendmail.mc" được tạo, chúng ta có thể tạo tập tin cấu hình "sendmail.cf" từ những câu lệnh sau:

```
[root@deep /]# cd /var/tmp/sendmail-version/cf/cf/
[root@deep cf]# m4 ../m4/cf.m4 sendmail.mc > /etc/mail/sendmail.cf
```

Chú thích: *Ở dây "../m4/cf.m4" bảo chương trình m4 biết nơi tìm kiếm thông tin của tập tin cấu hình mặc định.*

Tập tin "null.mc" chỉ dành cho những máy local hay neighbor client và server

Bởi vì các máy client cục bộ không bao giờ nhận thư tín trực tiếp từ bên ngoài, và trả lời (gởi đi) tất cả thư tín thông qua Mail Hub server. Chúng ta sẽ tạo một tập tin đặc biệt gọi là "null.mc", sau này khi được xử lý sẽ tạo ra một tập tin cấu hình "sendmail.cf" trả lời cho những thiết lập máy neighbor hoặc các máy local server client của chúng ta. Tập tin macro m4 này thì được tạo và cấu hình đơn giản bởi vì nó không cần nhiều đặc tính như tập tin cấu hình (sendmail.mc) mà Central Mail Hub server đã làm.

Bước 1:

Tạo tập tin null.mc (touch /var/tmp/sendmail-version/cf/cf/null.mc) và thêm vào những dòng sau:

```
OSTYPE(`linux')dnl
DOMAIN(`generic')dnl
FEATURE(`nullclient',`mail.your-domain.com')dnl
undefine(`ALIAS_FILE')dnl
```

Điều này nói lên rằng tập tin null.mc tự thiết lập cấu hình đặc trưng :

OSTYPE(`linux')

Tùy chọn cấu hình này chỉ rõ hệ điều hành mặc định của Sendmail sẽ thực thi, trong trường hợp này là hệ thống "linux". Mục này là một trong những mẩu tin nhỏ được yêu cầu bởi tập tin "mc".

DOMAIN(`generic')

Tùy chọn cấu hình sẽ chỉ rõ và mô tả một miền đặc biệt tương ứng với môi trường của bạn.

FEATURE(`nullclient', `mail.your-domain.com')

Macro m4 này thiết lập máy khách không nhận thư một cách trực tiếp, mà gởi thư đến một Central Mail Hub và tất cả thư sẽ tập trung ở máy chủ này sau đó sẽ gởi đi một cách trực tiếp. Đặc trưng này tạo một tập tin cấu hình không chứa gì, nhưng hỗ trợ để chuyển tất cả thư đến máy chủ Mail Hub thông qua mạng SMTP-based cục bộ. Tham số 'mail.your-domain.com' là tên của một Mail Hub. Dĩ nhiên bạn nên thay đổi tên này để phù hợp với máy chủ Mail Hub, ví dụ: FEATURE(`nullclient', `my.mailhub.com').

undefine(`ALIAS_FILE')

Tùy chọn cấu hình này ngăn cản phiên bản nullclient của Sendmail cố truy xuất đến những tập tin "/etc/mail/aliases" và "/etc/mail/aliases.db". Với việc thêm dòng này trong tập tin ".mc", bạn không cần có một tập tin "bí danh" trên tất cả máy khách của Sendmail. Tập tin bí danh chỉ được yêu cầu trên máy chủ Mail Hub cho tất cả máy chủ và máy khách bí danh trên mạng.

Chú thích: *Chúng tôi lưu ý rằng với cách cấu hình này thì không mailer nào được định nghĩa và không aliasing hoặc forwarding nào được thực hiện.*

Bước 2

Hiện tại tập tin cấu hình "null.mc" của chúng ta đã được tạo, chúng ta có thể tạo tập tin cấu hình "sendmail.cf" từ những câu lệnh trong tất cả các máy chủ lân cận hoặc máy client của chúng ta với các câu lệnh sau:

```
[root@deep /]# cd /var/tmp/sendmail-version/cf/cf/
[root@deep cf]# m4 ../m4/cf.m4 null.mc > /etc/mail/sendmail.cf
```

Bước 3

Đôi khi sẽ không có thư nào được phân phối đến máy cục bộ của bạn, vì không có sự kết nối để lấy thư gởi về, cho nên lúc này bạn sẽ không cần chạy một chương trình Sendmail daemon trên máy lân cận hoặc máy khách nữa.

Để ngưng lại chương trình Sendmail daemon đang vận hành trên máy lân cận hoặc máy khách của bạn cần phải hiệu chỉnh hoặc tạo tập tin "/etc/sysconfig/sendmail" thay đổi và thêm vào các dòng sau:

```
DAEMON=yes
```

Để thành:

```
DAEMON=no
```

Và:

```
QUEUE=1h
```

Chú thích: *Dòng "QUEUE=1h" trong tập tin "/etc/sysconfig/sendmail" khiến chương trình Sendmail xử lý hàng đợi mỗi giờ một lần, chúng ta bỏ dòng này bởi vì Sendmail vẫn xử lý hàng đợi này theo đúng định kỳ ngay cả trong trường hợp mà Mail Hub bị ngừng hoạt động.*

Bước 4

Những máy cục bộ thì không bao giờ dùng tập tin aliases, access, hoặc lược đồ cơ sở dữ liệu. Trừ khi tất cả các cơ sở dữ liệu này đã lưu trữ và được sử dụng trên máy chủ Central Mail Hub để hỗ trợ cho tất cả các máy cục bộ, chúng ta có thể xoá một cách an toàn các dòng lệnh và các trang man từ các máy cục bộ của chúng ta.

```
/usr/bin/newaliases
/usr/man/man1/newaliases.1
/usr/man/man5/aliases.5
```

- Để xoá các tập tin dưới đây từ hệ thống, bạn sử dụng các lệnh sau:

```
[root@client /]# rm -f /usr/bin/newaliases
[root@client /]# rm -f /usr/man/man1/newaliases.1
[root@client /]# rm -f /usr/man/man5/aliases.5
```

Bước 5

Xóa bỏ chương trình không cần thiết Procmail ra khỏi toàn bộ máy chủ cục bộ Sendmail và máy khách. Vì những máy cục bộ gởi tất các thư cục bộ cũng như các thư ra ngoài đến mail Hub Server để phân phối đi, nên không cần dùng một tác nhân phân phối thư phức tạp như chương trình Procmail để làm việc này. Thay vào đó chúng ta chỉ cần dùng chương trình mặc định "/bin/mail".

- Để xoá chương trình Procmail ra khỏi hệ thống, sử dụng những lệnh sau:

```
[root@client]# rpm -e procmail
```

Những tập tin "/etc/mail/access và access.db" dành cho Central Mail Hub

Tập tin cơ sở dữ liệu "access" được tạo ra để có thể chấp nhận hoặc từ chối mail từ những domain được chọn. Ví dụ như, bạn có thể chọn để từ chối những thư tín bắt nguồn từ vùng spammers mà mình đã biết, hoặc có thể chấp nhận để gởi tất cả các thư tín từ mạng cục bộ, từ bây giờ trở đi, về mặc định việc gởi bị từ chối bởi Sendmail (đây là một đặc tính của AntiSpam). Trong tập tin "access" và"access.db" ở ví dụ dưới đây, chúng ta sẽ cho phép gởi nhận mail từ tất cả các máy cục bộ có địa chỉ IP là 192.168.1. Tập tin "access" và "access.db" không đòi hỏi phải cài đặt trong cho cục bộ hoặc máy khách lân cận. Nó chỉ đòi hỏi được cài đặt trong Central Mail Hub để điều khiển tất cả các thư tín của bạn. Cũng cần lưu ý rằng sử dụng một Central Mail Hub sẽ nâng cao được tính bảo mật cũng như là quản lý các máy chủ khác và các máy khách trong mạng có chương trình sendmail đang chạy.

Bước 1

Tạo tập tin access (touch /etc/mail/access) và thêm vào những dòng sau:

```
# Description showing bellow for the format of this taap tin comes from
# the Sendmail source distribution under "cf/README" file.
#
```

```
# The table itself uses e-mail addresses, domain names, and network
# numbers as keys. For example,
#
# spammer@aol.com REJECT
# cyberspammer.com REJECT
# 192.168.212 REJECT
#
# would refuse mail from spammer@aol.com, any user from cyberspammer.com
# (or any host within the cyberspammer.com domain), and any host on the
# 192.168.212.* network.
#
# The value part of the map can contain:
#
# OK Accept mail even if other rules in the
# running ruleset would reject it, for example,
# if the domain name is unresolvable.
# RELAY Accept mail addressed to the indicated domain or
# received from the indicated domain for relaying
# through your SMTP server. RELAY also serves as
# an implicit OK for the other checks.
# REJECT Reject the sender or recipient with a general
# purpose message.
# DISCARD Discard the message completely using the
# $#discard mailer. This only works for sender
# addresses (i.e., it indicates that you should
# discard anything received from the indicated
# domain).
# ### any text where ### is an RFC 821 compliant error code
# and "any text" is a message to return for
# the command.
#
# For example:
#
# cyberspammer.com 550 We don't accept mail from spammers
# okay.cyberspammer.com OK
# sendmail.org OK
# 128.32 RELAY
#
# would accept mail from okay.cyberspammer.com, but would reject mail
# from all other hosts at cyberspammer.com with the indicated message.
# It would allow accept mail from any hosts in the sendmail.org domain,
```

```
# and allow relaying for the 128.32.*.* network.
#
# You can also use the access database to block sender addresses based on
# the username portion of the address. For example:
#
# FREE.STEALTH.MAILER@ 550 Spam not accepted
#
# Note that you must include the @ after the username to signify that
# this database entry is for checking only the username portion of the
# sender address.
#
# If you use like we do in our "sendmail.mc macro configuration:
#
# FEATURE(`blacklist_recipients')
#
# then you can add entries to the map for local users, hosts in your
# domains, or addresses in your domain which should not receive mail:
#
# badlocaluser 550 Mailbox disabled for this username
# host.mydomain.com 550 That host does not accept mail
# user@otherhost.mydomain.com 550 Mailbox disabled for this recipient
#
# This would prevent a recipient of badlocaluser@mydomain.com, any
# user at host.mydomain.com, and the single address
# user@otherhost.mydomain.com from receiving mail. Enabling this
# feature will keep you from sending mails to all addresses that
# have an error message or REJECT as value part in the access map.
# Taking the example from above:
#
# spammer@aol.com REJECT
# cyberspammer.com REJECT
#
# Mail can't be sent to spammer@aol.com or anyone at cyberspammer.com.
#
# Now our configuration of access file,
# by default we allow relaying from localhost...
localhost.localdomain RELAY
localhost RELAY
127.0.0.1 RELAY
192.168.1 RELAY
```

Ghi nhớ: *Đừng quên chỉ ra những dãy địa chỉ IP riêng trong tập tin "access" này cho phép gởi hoặc không cho phép gởi từ bên trong mạng của bạn.*

Bước 2

Tạo tập tin access.db

Nên nhớ rằng, vì "/etc/mail/access " là một cơ sở dữ liệu, sau khi tạo tập tin văn bản được mô tả như trên, bạn phải dùng chương trình tiện ích "makemap" để tạo một bảng cơ sở dữ liệu.

- Để tạo "access database map", sử dụng lệnh sau

```
[root@deep /]# makemap hash /etc/mail/access.db < /etc/mail/access
```

Những tập tin "/etc/mail/aliases and aliases.db" dành cho Central Mail Hub

Aliasing là tiến trình chuyển đổi tên của một người nhận thư cục bộ thành tên khác (aliasing chỉ xuất hiện trong những tên nội bộ). Ví dụ như chuyển đổi một tên chung (như là root) thành một username thật trên hệ thống hoặc chuyển đổi một tên vào trong một danh sách nhiều tên (như mailling list). Sendmail sẽ tìm kiếm tên của những người nhận thư trong tập tin "aliases" bởi vì sendmail có thể tìm kiếm hàng ngàn tên trong tập tin "aliases", một bản sao chép của tập tin này được lưu trữ trong những vùng riêng biệt của cơ sở dữ liệu 'db" để cải thiện tốc độ tìm kiếm. Nếu cấu hình Sendmail của bạn dùng Central Server(Mail Hub) để điều khiển tất cả mail, thì bạn không cần cài đặt tập tin "aliases" và "aliases.db" trên máy neighbor or client.

Bước 1

Tạo tập tin aliases (touch /etc/mail/aliases) và thêm vào những dòng sau bằng giá trị mặc định.

```
#
# @(#)aliases 8.2 (Berkeley) 3/5/94
#
# Aliases in this file will NOT be expanded in the header from
# Mail, but WILL be visible over networks or from /bin/mail.
#
# >>>>>>>>>> The program "newaliases" must be run after
# >> NOTE >> this file is updated for any changes to
# >>>>>>>>>> show through to sendmail.
#
# Basic system aliases -- these MUST be present.
MAILER-DAEMON: postmaster
postmaster: root
# General redirections for pseudo accounts.
bin: root
```

```
daemon: root
nobody: root
# Person who should get root's mail
#root: admin
```

Chú thích: *Tập tin aliases của bạn có khả năng sẽ phức tạp hơn , nhưng dù có đúng như thế thì, trong ví dụ phải làm thế nào để chỉ ra những mẫu nhỏ nhất của aliases.*

Bước 2

Tạo tập tin aliases.db.

Vì "/etc/mail/aliases" là một cơ sở dữ liệu, nên sau khi tạo tập tin văn bản như trên, chúng ta phải dùng chương trình "makemap" để tạo bảng cơ sở dữ liệu.

- Để tạo "aliases database map", dùng lệnh sau:

```
[root@deep /]# makemap hash /etc/mail/aliases.db < /etc/mail/aliases
```

Những tập tin "/etc/mail/virtusertable,domaintable, mailertable, và virtusertable.db, domaintable.db, mailertable.db" dành cho Central Mail Hub

Tất cả những tập tin trên đều liên quan đến tính năng đặc biệt của Sendmail được điều chỉnh bởi người quản trị hệ thống. Một lần nữa nên nhớ rằng, những tính năng này thường chỉ đòi hỏi chạy trên Central Mail Hub server và sau đây là giải thích của mỗi tập tin.

Những tập tin virtusertable và virtusertable.db

Virtusestable là một cơ sở dữ liệu ánh xạ vào trong những miền ảo thành những địa chỉ mới, với tính năng này, những thư tín cho vùng ảo trên hệ thống của bạn có thể được phân phối đến địa chỉ cục bộ, địa chỉ từ xa, hoặc địa chỉ của người dùng riêng lẻ.

Những tập tin domaintable và domaintable.db

Domaintable là một cơ sở dữ liệu ánh xạ từ một miền cũ thành miền mới. Với tính năng này, nhiều tên miền trên hệ thống máy tính của bạn có thể được ghi lại từ những miền cũ đến những miền mới.

Những tập tin mailertable và mailertable.db

Mailertable là một cơ sở dữ liệu ánh xạ tên "host.domain" đến tác nhân phân bố đặc biệt và những cặp tên miền mới. Với tính năng này, thư tín có thể được phân phối thông qua việc sử dụng tác nhân phân phối cụ thể để đến tên miền mới cục bộ hoặc từ xa.

- Để tạo virtusertable, domaintable, mailertable, và những tập tin tương ứng ".db" vào trong thư mục "/etc/mail", sử dụng các lệnh sau :

```
[root@deep /]# for map in virtusertable domaintable mailertable
> do
> touch /etc/mail/${map}
```

```
> chmod 0644 /etc/mail/${map}
> makemap hash /etc/mail/${map}.db < /etc/mail/${map}
> chmod 0644 /etc/mail/${map}.db
> done
```

Tập tin "/etc/mail/local-host-names" dành cho tất cả các kiểu cấu hình:

Đọc luân phiên các tên trong tập tin "/etc/mail/local-host-names" cho máy chủ cục bộ. Và dùng tập tin này khai báo danh sách các host trên mạng máy tính của bạn để cho máy cục bộ này hoạt động giống như người nhận thư MX. Đơn giản chúng ta chỉ cần thêm vào tên của các máy cho máy chủ (nghĩa là mail.openna.com) để nó điều khiển các thư tín chuyển đến "/etc/mail/local-host-names". Đây là một ví dụ:

Tạo tập tin local-host-names (touch /etc/mail/local-host-names) và thêm vào những dòng sau:

```
# local-host-names - include all aliases for your machine here.
openna.com
deep.openna.com
www.openna.com
win.openna.com
mail.openna.com
```

Với kiểu cấu hình này, tất cả các thư tín gởi đi sẽ xuất hiện như thể là nó đã được gởi từ "openna.com" và bất kỳ thư tín gởi đến **www.openna.com** hoặc những máy chủ khác sẽ được phân phối đến mail Hub "mail.openna.com" của chúng ta.

Nên biết rằng, nếu bạn cấu hình hệ thống của bạn giả dạng (masquerade) tới hệ thống khác, thì bất e-mail gởi đi từ hệ thống của bạn sẽ gởi đến máy tính mà bạn đang masquerade. Trong ví dụ minh họa trên, những tập tin log theo định kỳ nên gởi đến **root@www.openna.com** bởi cron daemon của Linux nhưng đã được gởi đến **root@mail.openna.com** trong Mail Hub của chúng ta.

Cấu hình của tập tin "/etc/sysconfig/sendmail" dành cho tất cả các kiểu cấu hình:

Tập tin /etc/sysconfig/sendmail" được dùng để chỉ rõ thông tin cấu hình SENDMAIL, chẳng hạn như nếu Sendmail này đã vận hành giống như một daemon để chờ nhận thư và nó sẽ đợi trong bao lâu trước khi gởi một cảnh báo nếu các thư trong hàng đợi đã không được phân phối.

Hãy tạo tập tin sendmail (touch /etc/sysconfig/sendmail) và thêm những dòng vào tập tin này:

```
DAEMON=yes
QUEUE=1h
```

Tùy chọn "DAEMON=yes" chỉ dẫn Sendmail chạy như một daemon. Dòng này rất hữu ích khi các máy Sendmail client được định cấu hình để không chấp nhận thư tín trực tiếp từ bên ngoài và nhờ chuyển các thư cục bộ này đến Central Hub, không chạy một daemon cũng cải tiến vấn đề bảo mật. Nếu máy chủ hoặc máy khách của bạn đã định cấu hình như cách trên thì tất cả những gì bạn làm là thay đổi từ "DAEMON=yes" đến "DAEMON=no".

Thông thường thư tín được đặt vào trong hàng đợi vì nó không thể được truyền tải ngay tức khắc được. "QUEUE=1h" thiết lập một khoảng thời gian trước khi gởi một cảnh báo đến người gởi thư nếu những thư không được phân phối đi.

Cấu hình của tập tin script "/etc/rc.d/init.d/sendmail" cho tất cả các kiểu cấu hình

Định cấu hình tập tin script "/etc/rc.d/init.d/sendmail" để khởi động và ngưng lại chương trình Sendmail daemon.

Tạo tập tin sendmail script (touch /etc/rc.d/init.d/sendmail) và thêm:

```
#!/bin/sh
#
# sendmail This shell script takes care of starting and stopping
# sendmail.
#
# chkconfig: 2345 80 30
# description: Sendmail is a Mail Transport Agent, which is the program \
# that moves mail from one machine to another.
# processname: sendmail
# config: /etc/sendmail.cf
# pidfile: /var/run/sendmail.pid
# Source function library.
. /etc/rc.d/init.d/functions
# Source networking configuration.
. /etc/sysconfig/network
# Source sendmail configuration.
if [ -f /etc/sysconfig/sendmail ] ; then
    . /etc/sysconfig/sendmail
else
    DAEMON=yes
    QUEUE=1h
fi
# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0
[ -f /usr/sbin/sendmail ] || exit 0
RETVAL=0
```

```
# See how we were called.
case "$1" in
   start)
         # Start daemons.
         echo -n "Starting sendmail: "
         /usr/bin/newaliases > /dev/null 2>&1
         for i in virtusertable access domaintable mailertable ; do
            if [ -f /etc/mail/$i ] ; then
            makemap hash /etc/mail/$i < /etc/mail/$i
         fi
         done
         daemon /usr/sbin/sendmail $([ "$DAEMON" = yes ] && echo -bd) \
                        $([ -n "$QUEUE" ] && echo -q$QUEUE)
         RETVAL=$?
         echo
         [ $RETVAL -eq 0 ] && touch /var/lock/subsys/sendmail
         ;;
    stop)
         # Stop daemons.
         echo -n "Shutting down sendmail: "
         killproc sendmail
         RETVAL=$?
         echo
         [ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/sendmail
         ;;
   restart|reload)
         $0 stop
         $0 start
         RETVAL=$?
         ;;
   status)
         status sendmail
         RETVAL=$?
         ;;
    *)
         echo "Usage: sendmail {start|stop|restart|status}"
         exit 1
esac
exit $RETVAL
```

Bây giờ, hãy tạo script có thể thực thi được và thay đổi quyền mặc định của nó :

```
[root@deep /]# chmod 700 /etc/rc.d/init.d/sendmail
```

Hãy tạo biểu tượng kết nối rc.d cho Sendmail với lệnh:

```
[root@deep /]# chkconfig --add sendmail
```

Khởi động chương trình sendmail một cách thủ công bằng các lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/sendmail start
Starting sendmail: [ OK ]
```

Dọn dẹp sau quá trình cài đặt

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf sendmail-version/ sendmail.version.tar.gz
```

Lệnh "rm" được dùng ở trên sẽ gỡ bỏ hết những tập tin nguồn mà chúng ta đã dùng để biên dịch và cài đặt Sendmail. Nó cũng dùng để xoá những tập tin nén trong thư mục "/var/tmp".

Bảo mật Sendmail

Hạn chế của lệnh "smrsh" trong Sendmail

Chương trình smrsh được dùng như để thay thế cho "/bin/sh" trong chương trình mailer của sendmail. Nó là một tiện ích shell bị hạn chế cung cấp danh sách cụ thể của những chương trình thực thi có sẵn của Sendmail thông qua thư mục "/etc/smrsh". Chính xác hơn, dù là một tên vô danh tiểu tốt nào có thể dùng Sendmail để thực thi chương trình mà không cần thông qua tập tin aliases hoặc forward, smrsh giới hạn tập các chương trình mà nó có thể thực thi được. Khi liên kết với Sendmail, smrsh sẽ giới hạn một cách có hiệu quả phạm vi hoạt động của chương trình Sendmail và chỉ thực thi những chương trình đặt trong thư mục smrshl. Nếu bạn làm theo tất cả những gì mà chúng tôi đã thực hiện ở trên thì chương trình smrsh đã được biên dịch và cài đặt sẵn trên máy tính của bạn dưới thư mục "/usr/sbin/smrsh".

Bước 1

Việc đầu tiên mà chúng ta cần làm là xác định rõ danh sách của các lệnh "smrsh" sẽ cho phép chương trình sendmail hoạt động.

Về mặc định, chúng ta không giới hạn

"/bin/mail" (Nếu bạn có cài đặt nó trong hệ thống của bạn)

"/usr/bin/procmail" (Nếu bạn có cài đặt nó trong hệ thống của bạn)

Chú thích: *Bạn không nên bao gồm những chương trình như : sh(1), csh(1), perl(1), uudecode(1), hoặc sed (1) trong danh sách các lệnh mà bạn có thể chấp nhận.*

Bước 2

Bước kế tiếp mà bạn cần phải làm là lưu những chương trình cho phép sendmail thực thi vào trong thư mục "/etc/smrsh". Để ngăn chặn những chương

trình trùng lặp, và cho công việc được suôn sẻ, cách tốt nhất là bạn thiết lập sự liên kết với chương trình đã được phép trong thư mục "/etc/smrsh" hơn là sao chép những chương trình đến thư mục này.

Để cho phép chạy chương trình mail "/bin/mail", sử dụng những lệnh sau:

```
[root@deep /]# cd /etc/smrsh
[root@deep smrsh]# ln -s /bin/mail mail
```

Để cho phép chạy chương trình procmail "/usr/bin/procmail", sử dụng những lệnh sau:

```
[root@deep /]# cd /etc/smrsh
[root@deep smrsh]# ln -s /usr/bin/procmail procmail
```

Điều này cho phép chương trình mail và procmail được thực thi từ một tập tin ".forward" hoặc ".aliases" của người dùng, sử dụng cú pháp "program"

Chú thích: *Procmail chỉ đòi hỏi ở trên môi trường Mail Hub Server và không ở trong Local Client Mail Server. Nếu bạn đã định cấu hình hệ thống giống như Mail Hub Server thì sau đó phải làm sự kết nối với procmail như đã giải thích ở trên, nếu bạn đã định cấu hình hệ thống như một Local Client Server thì sau đó bỏ qua bước procmail ở trên.*

Bước 3

Bây giờ chúng ta có thể dùng cấu hình Sendmail để hạn chế shell. Chương trình mailer được xác định bởi một dòng đơn trong tập tin cấu hình Sendmail "/etc/mail/sendmail.cf". Bạn phải hiệu chỉnh dòng đơn "Mprog" này trong tập tin "sendmail.cf", bằng việc thay thế "/bin/sh" với "/usr/sbin/smrsh".

Hiệu chỉnh tập tin sendmail.cf (vi /etc/mail/sendmail.cf) và thay đổi dòng :

Ví dụ:

```
Mprog, P=/bin/sh, F=lsDFMoqeu9, S=10/30, R=20/40, D=$z:/, T=X-Unix, A=sh -c $u
```

Sẽ được đổi thành:

```
Mprog, P=/usr/sbin/smrsh, F=lsDFMoqeu9, S=10/30, R=20/40, D=$z:/, T=X-Unix, A=sh -
c $u
```

- Khởi động lại chương trình sendmail một cách thủ công với lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/sendmail restart
```

Chú thích: *Trong tập tin cấu hình "sendmail.mc" dành cho Mail Hub Server ở trên, chúng ta đã định cấu hình sẵn dòng "Mprog" để dùng hạn chế lệnh "/usr/sbin/smrsh" với macro m4 "FEATURE('smrsh','/usr/sbin/smrsh'), vì thế đừng có ngạc nhiên nếu "/usr/sbin/smrsh" đã được cài đặt trong tập tin của bạn "/etc/mail/sendmail.cf" dành cho Mail Hub relay. Thay vì, dùng phương pháp trên để chỉ ra các tập tin "/etc/mail/sendmail.cf" khác trên mạng máy tính của bạn giống như máy nullclient ("local or neighbor client and server") đã dùng tập tin cấu hình macro "null.mc" để tạo ra tập tin "/etc/mail/sendmail.cf".*

Tập tin "/etc/mail/aliases"

Một cách quản trị dở hoặc thiếu cẩn thận với tập tin "aliases" có thể dễ dàng bị can thiệp vào trình trạng không cần thiết phải tiết lộ. Ví dụ, nhiều hệ thống bán hàng tự động với một alias "decode" trong tập tin "/etc/mail/aliases". Mục đích là cung cấp phương cách tốt cho người dùng để chuyển tải những tập tin nhị phân phục vụ cho việc gởi, nhận mail. Ở phía gởi mail, chương trình sẽ chuyển đổi tập tin dạng nhị phân sang dạng ASCII với "uuencode", sau đó các mail ấy sẽ được "decode" ở phía nhận mail thông qua chương trình "/usr/bin/uuencode". Kết quả là các tập tin ASCII sẽ được chuyển đổi lại thành các tập tin nhị phân gốc.

Hãy gỡ bỏ dòng alias "decode" ra khỏi tập tin "/etc/mail/aliases" trên máy của bạn. Tương tự, không được cài những chương trình alias mà thực thi một chương trình trên máy cho tới khi việc kiểm tra được hoàn thành và phần gỡ bỏ đã làm xong.

Hiệu chỉnh tập tin aliases (vi /etc/mail/aliases) và gỡ bỏ những dòng sau :

```
# Basic system aliases -- these MUST be present.
MAILER-DAEMON: postmaster
postmaster:     root
# General redirections for pseudo accounts.
bin:            root
daemon:         root
games:          root          ← Loại bỏ dòng này.
ingres:         root          ← Loại bỏ dòng này.
nobody:         root
system:         root          ← Loại bỏ dòng này.
toor:           root          ← Loại bỏ dòng này.
uucp:           root          ← Loại bỏ dòng này.
# Well-known aliases.
manager:        root          ← Loại bỏ dòng này.
dumper:         root          ← Loại bỏ dòng này.
operator:       root          ← Loại bỏ dòng này.
# trap decode to catch security attacks
decode:         root          ← Loại bỏ dòng này.
# Person who should get root's mail
#root:          marc
```

- Để phần thay đổi có hiệu quả bạn cần thực thi lệnh sau:

```
[root@deep /]# /usr/bin/newaliases
```

Ngăn chặn việc lạm dụng chương trình sendmail của những người không được phép dùng

Chương trình Sendmail hiện tại bao gồm cả những tính năng Anti-Spam mạnh, nó sẽ giúp ngăn chặn những dịch vụ mail đang bị lạm dụng bởi những người dùng không được phép. Để làm được việc này, hãy hiệu chỉnh tập tin "/etc/mail/sendmail.cf" và làm một sự thay đổi về tập tin cấu hình để ngăn chặn hẳn spammers.

Hiệu chỉnh tập tin sendmail.cf (vi /etc/mail/sendmail.cf) và thay đổi dòng:

O PrivacyOptions=authwarnings

Để thành:

O PrivacyOptions=authwarnings,goaway

Việc cài đặt tùy chọn "goaway" là nguyên nhân mà Sendmail không chấp nhận tất cả lệnh SMTP "EXPN" nó cũng là lý do để chính nó không nhận những lệnh SMTP "VERB" và SMTP "VRFY". Những việc thay đổi này sẽ giúp ngăn chặn spammers dùng những tập lệnh "EXPN" và "VRFY" trong Sendmail.

Giới hạn người kiểm tra nội dung của hàng đợi

Thông thường, bất cứ người nào cũng có thể kiểm tra nội dung hàng đợi bằng cách dùng lệnh "mailq". Để hạn chế việc này, bạn phải chỉ định tùy chọn "restrictmailq" trong tập tin "/etc/mail/sendmail.cf". Với tùy chọn này, Sendmail chỉ cho phép nhóm người sử dụng có chức năng giống như quyền sở hữu của nhóm thư mục (root) kiểm tra những phần nội dung đó. Điều này cho phép hàng đợi được bảo vệ một cách đầy đủ với mode 0700. trong khi những người được chọn vẫn thấy được nội dung.

Hiệu chỉnh tập tin sendmail.cf (vi/etc/sendmail.cf) và thay đổi dòng:

O PrivacyOptions=authwarnings,goaway

Để thành:

O PrivacyOptions=authwarnings,goaway,restrictmailq

- Bây giờ hãy thay đổi mode của thư mục hàng đợi, để được bảo vệ một cách đầy đủ:

  [root@deep /]# chmod 0700 /var/spool/mqueue

- Bây giờ khởi động lại sendmail (một cách thủ công) để sự thay đổi này có hiệu lực:

  [root@deep /]# /etc/rc.d/init.d/sendmail restart
  Shutting down sendmail: [OK]
  Starting sendmail: [OK]

Chú thích: *Chúng ta đã thêm tùy chọn "goaway" vào dòng "PrivacyOption=" trong tập tin sendmail.cf và bây giờ tiếp tục thêm tùy chọn "restrictmailq" vào dòng này.*

Bất kỳ user nào không có quyền kiểm tra nội dung hàng đợi mà cố gắng chạy lệnh nầy sẽ nhận thông báo:

[user@deep /]$ /usr/bin/mailq

You are not permitted to see the queue

Giới hạn việc xử lý hàng đợi cho "root"

Thông thường , bất cứ người nào cũng có thể xử lý hàng đợi này với tùy chọn "-q". Để giới hạn việc xử lý hàng đợi của "root" và quyền sở hữu của thư mục hàng đợi bạn phải chỉ rõ tùy chọn "restrictqrun" trong tập tin "/etc/mail/sendmail.cf"

Hiệu chỉnh tập tin sendmail.cf (vi/etc/mail/sendmail.cf) và thay đổi dòng :

```
O PrivacyOptions=authwarnings,goaway,restrictmailq
```

Để thành:

```
O PrivacyOptions=authwarnings,goaway,restrictmailq,restrictqrun
```

- Bây giờ khởi động lại sendmail (một cách thủ công) để sự thay đổi này có hiệu lực:

```
[root@deep /]# /etc/rc.d/init.d/sendmail restart
Shutting down sendmail:          [ OK ]
Starting sendmail:               [ OK ]
```

Bất kỳ user nào không có quyền xử lý nội dung hàng đợi mà cố gắng chạy lệnh này sẽ nhận thông báo:

```
[user@deep /]$ /usr/sbin/sendmail -q
You do not have permission to process the queue
```

Những thông điệp chúc mừng của SMTP

Khi Sendmail chấp nhận một sự kết nối với SMTP thì nó sẽ gởi một thông điệp chúc mừng này đến máy chủ khác. Thông tin này xác định rõ máy cục bộ và là nơi đầu tiên mà nó sẵn sàng gởi tới.

Hiệu chỉnh tập tin sendmail.cf (vi /etc/mail/sendmail.cf) và thay đổi dòng :

```
O SmtpGreetingMessage=$j Sendmail $v/$Z; $b
```

Để thành:

```
O SmtpGreetingMessage=$j
```

- Bây giờ khởi động lại sendmail (một cách thủ công) để sự thay đổi này có hiệu lực:

```
[root@deep /]# /etc/rc.d/init.d/sendmail restart
Shutting down sendmail:          [ OK ]
Starting sendmail:               [ OK ]
```

Việc thay đổi này sẽ không ảnh hưởng đến bất cứ thứ gì, nhưng nó đã được giới thiệu bởi những người trong newgroup (nhóm thảo luận) news.admin.net-abuse.email như là một sự phòng bị hợp pháp. Nó hiệu chỉnh những biểu ngữ mà sendmail đã hiển thị trong quá trình kết nối.

Đặt bit không thay đổi trên các tập tin sendmail quan trọng

Những tập tin Sendmail quan trọng có thể được thiết lập là không thay đổi để cho việc bảo mật tốt hơn với lệnh "chattr" của Linux. Tập tin với thuộc tính "+i" không thể nào được hiệu chỉnh, xóa hoặc đổi tên; không có một sự kết nối nào được thiết lập với tập tin này và không một cơ sở dữ liệu nào được ghi trên đó. Chỉ có super-user có thể cài đặt hoặc xóa thuộc tính này.

- Hãy cài đặt bit không thay đổi trên tập tin "sendmail.cf".

```
[root@deep /]# chattr +i /etc/mail/sendmail.cf
```

- Hãy cài đặt bit không thay đổi trên tập tin "local-host-names".
  [root@deep /]# chattr +i /etc/mail/local-host-names
- Hãy cài đặt bit không thay đổi trên tập tin "aliases".
  [root@deep /]# chattr +i /etc/mail/aliases
- Hãy cài đặt bit không thay đổi trên tập tin "access".
  [root@deep /]# chattr +i /etc/mail/access

Tài liệu bổ sung

Để cho chi tiết hơn, có một vài trang man mà bạn cần đọc:

$ man aliases (5)	- Tập tin bí danh (alias) cho Sendmail.
$ man makemap (8)	- Tạo cơ sở dữ liệu ánh xạ cho sendmail.
$ man sendmail (8)	- Một tác nhân truyền tải thư điện tử.
$ man mailq (1)	- Cho biết các mail trong hàng đợi (mail queue).
$ man newaliases (1)	- Xây dựng lại cơ sở dữ liệu cho tập tin mail aliases.
$ man mailstats (8)	- Trình bày thống kê mail.
$ man praliases (8)	- Trình bày hệ thống mail aliases.

Các công cụ quản trị Sendmail

Tập lệnh liệt kê ở dưới có một số lệnh chúng ta thường dùng, nhưng một số thì ít khi hoặc chưa khi nào dùng đến. Hãy kiểm tra trang man và tài liệu để nắm thông tin chi tiết hơn về phần này

newaliases

Mục đích chương trình tiện ích "newaliases" của sendmail là tạo và cập nhật lại việc truy cập cơ sở dữ liệu cho tập tin mail aliases "/etc/mail/aliases" nó phải được thực thi sau mỗi lần bạn thay đổi nội dung của tập tin này để việc thay đổi đó mang lại hiệu quả nhiều hơn.

- Cập nhật tập tin aliases với tiện ích "newaliases", sử dụng dòng lệnh sau:
  [root@deep /]# /usr/bin/newaliases

makemap

Mục đích của chương trình tiện ích "makemap" là tạo bảng cơ sở dữ liệu trong Sendmail. Lệnh "makemap" chỉ được dùng khi nào mà bạn cần tạo một cơ sở dữ liệu mới cho tập tin giống như aliases, access, domaintable, mailertable và virtusertable.

- Thực thi makemap để tạo một cơ sở dữ liệu mới cho access, sử dụng lệnh sau:
  [root@deep /]# makemap hash /etc/mail/access.db < /etc/mail/access

Ở đây <hash> là định dạng cơ sở dữ liệu, makemap có thể nắm giữ ba định dạng cơ sở dữ liệu khác nhau; đó là "hash", "btree", hoặc "dbm". </etc/mail/access.db> là vị trí và tên của cơ sở dữ liệu mới sẽ được tạo. </etc/mail/access> là vị trí của tập tin mà ở đó makemap sẽ đọc tập tin chuẩn. Trong ví dụ, chúng ta đã tạo được một tập tin mới "access.db" với lệnh makemamp

ở trên. Để tạo cơ sở dữ liệu cho những tập tin khác như aliases, domaintable, mailertable và virtusertable, bạn phải chỉ rõ vị trí và tên tập tin tương ứng trong lệnh "makemap"

mailq

Mục đích tiện ích của chương trình "mailq" là liệt kê danh sách thư trong hàng đợi sẽ phân phối đến địa chỉ cần đến.

- Để liệt kê danh sách thư trong hàng đợi, sử dụng lệnh sau:

```
[root@deep /]# mailq
Mail queue is empty
```

Các công cụ sử dụng Sendmail

Tập lệnh liệt kê ở dưới có một số chúng ta thường dùng, nhưng một số lại không hiện hữu. Hãy kiểm tra trang man và tài liệu để nắm thông tin chi tiết hơn về phần này.

Mailstats

Mục đích của chương trình tiện ích "mailstats" là trình bày nội dung số liệu thống kê thư hiện tại.

- Để trình bày số liệu thống kê mail hiện tại, sử dụng lệnh sau

```
[root@deep /]# mailstats
Statistics from Tue Dec 14 20:31:48 1999
 M  msgsfr  bytes_from  msgsto  bytes_to  msgsrej  msgsdis  Mailer
 5     0        0K        1        3K        0        0     esmtp
 8  1259     19618K     1259     19278K      0        0     local
 ==========================================================
 T  1259     19618K     1260     19281K      0        0
```

praliases

Mục đích của chương trình tiện ích "praliases" là trình bày trên một dòng, không cần thứ tự nội dung hệ thống mail aliases hiện tại.

- Để trình bày hệ thống aliases hiện tại, sử dụng lệnh as

```
[root@deep /]# praliases
postmaster:root
daemon:root
root:admin
@:@
mailer-daemon:postmaster
bin:root
nobody:root
webadmin:admin
www:root
```

Những tập tin cài đặt cho Sendmail Central Mail Hub

> /etc/rc.d/init.d/sendmail
> /etc/rc.d/rc0.d/K30sendmail
> /etc/rc.d/rc1.d/K30sendmail
> /etc/rc.d/rc2.d/S80sendmail
> /etc/rc.d/rc3.d/S80sendmail
> /etc/rc.d/rc4.d/S80sendmail
> /etc/rc.d/rc5.d/S80sendmail
> /etc/rc.d/rc6.d/K30sendmail
> /etc/sysconfig/sendmail
> /etc/mail
> /etc/mail/statistics
> /etc/mail/sendmail.cf
> /etc/mail/access
> /etc/mail/access.db
> /etc/mail/aliases
> /etc/mail/aliases.db
> /etc/mail/virtusertable
> /etc/mail/virtusertable.db
> /etc/mail/domaintable
> /etc/mail/domaintable.db
> /etc/mail/mailertable
> /etc/mail/mailertable.db
> /etc/mail/local-host-names
> /etc/smrsh
> /usr/bin/newaliases
> /usr/bin/mailq
> /usr/bin/hoststat
> /usr/bin/purgestat
> /usr/lib/sendmail
> /usr/man/man1/mailq.1
> /usr/man/man1/newaliases.1
> /usr/man/man5/aliases.5
> /usr/man/man8/sendmail.8
> /usr/man/man8/mailstats.8
> /usr/man/man8/makemap.8
> /usr/man/man8/praliases.8
> /usr/man/man8/smrsh.8
> /usr/sbin/sendmail
> /usr/sbin/mailstats
> /usr/sbin/makemap
> /usr/sbin/praliases
> /usr/sbin/smrsh
> /var/spool/mqueue

Những tập tin cài đặt cho Sendmail server nội bộ hoặc client

> /etc/rc.d/init.d/sendmail
> /etc/rc.d/rc0.d/K30sendmail
> /etc/rc.d/rc1.d/K30sendmail
> /etc/rc.d/rc2.d/S80sendmail
> /etc/rc.d/rc3.d/S80sendmail
> /etc/rc.d/rc4.d/S80sendmail
> /etc/rc.d/rc5.d/S80sendmail
> /etc/rc.d/rc6.d/K30sendmail
> /etc/sysconfig/sendmail
> /etc/mail
> /etc/mail/statistics
> /etc/mail/sendmail.cf
> /etc/mail/local-host-names
> /etc/smrsh
> /usr/bin/mailq
> /usr/bin/hoststat
> /usr/bin/purgestat
> /usr/lib/sendmail
> /usr/man/man1/mailq.1
> /usr/man/man8/sendmail.8
> /usr/man/man8/mailstats.8
> /usr/man/man8/smrsh.8
> /usr/sbin/sendmail
> /usr/sbin/mailstats
> /usr/sbin/smrsh
> /var/spool/mqueue

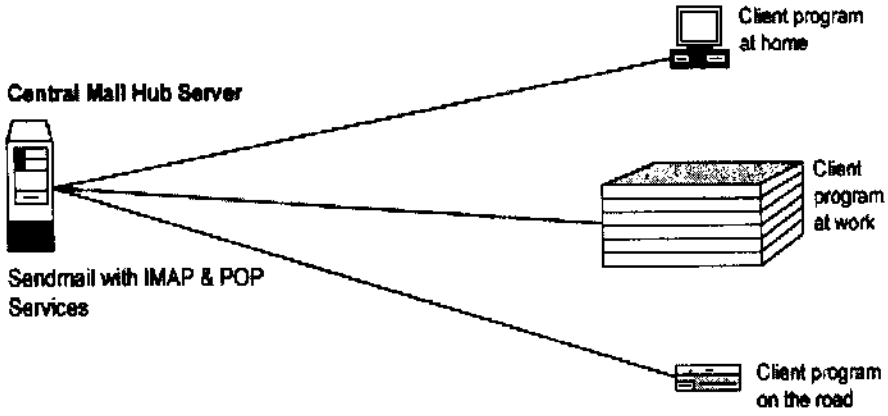
Linux IMAP & POP Server

Tổng quan

Nếu máy của bạn đã có cấu hình Sendmail giống như server Central Mail Hub, thì bạn bắt buộc phải cài thêm phần mềm IMAP/POP, nếu không bạn sẽ không nắm bắt được tiện ích của Linux Mail Server vì Sendmai là phần mềm có nhiệm vụ gởi thư từ máy này đến máy khác mà thôi. Một Mail Server là một server mà cùng một lúc có thể chạy được một hoặc nhiều chương trình sau : Một server cho dịch vụ IMAP (IMAP server), một server cho dịch vụ POP 3 (POP 3 server), một server cho dịch vụ POP 2 (POP 2 server), cùng một server cho dịch vụ SMTP (SMTP server). Ví dụ SMTP Server là Sendmail và phải được cài đặt sẵn trên Linux server của bạn như một Central Mail Hub trước khi tiếp tục cài đặt phần tiếp theo trong cuốn sách này. Bây giờ chúng ta sẽ tập trung vào việc cài đặt IMAP4, POP3, và POP2, mỗi dịch vụ này tương ứng với một gói dữ liệu riêng lẻ.

Với phần mềm IMAP &POP, một chương trình gởi thư điện tử dành cho "client" từ xa có thể truy cập những thư tín được lưu trữ trên Linux mail server giống như đã ở trên máy local. Ví dụ, thư điện tử đã nhận và lưu trữ trên một IMAP server thì người dùng có thể tải chúng về để phục vụ cho máy tính ở nhà hoặc ở cơ quan, ..., không cần thiết phải chuyển những thư tín hoặc những tập tin đi đi lại lại giữa các máy tính.

POP viết tắt từ cụm từ "Post Office Protocol" và đơn giản là nó cho phép bạn lấy những thư tín, truy cập và thậm chí là có thể xóa chúng đi. IMAP là POP trên steroids, nó cho phép bạn duy trì nhiều tài khoản, nhiều người truy cập một account một cách dễ dàng, giữ thư tín trên máy chủ, chỉ chép xuống phần đầu, hoặc phần nội dung của thư và không được đính kèm theo tập tin .... IMAP là phần mềm lý tưởng để dùng cho những ai làm việc không cố định một nơi. Về mặc định những server POP và IMAP là những nguồn phân bố tốt nhất khi ta cần.



Hướng dẫn cài đặt

Những lệnh dưới đây tương thích với hệ điều hành Unix.

Đường dẫn nguồn: "/var/tmp"(có thể chọn đường dẫn khác).

Sự cài đặt này đã được kiểm tra trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước cài đặt được thực hiện với quyền super-user "root".

Phiên bản IMAP là 4.7c.

Gói (Packages):

IMAP/POP Homepage: http://www.washington.edu/imap/
IMAP/POP FTP Site: 140.142.3.227 or 140.142.4.227

Bạn phải chắc chắn đã tải xuống : imap.tar.Z.

Những điều kiện cần có trước tiên:

- Sendmail server phải được cài đặt sẵn trên hệ thống máy tính của bạn để có thể sử dụng phần mềm IMAP và POP.

Chú thích: *Để nắm bắt được nhiều thông tin về những phần mềm được yêu cầu, hãy xem sự liên hệ giữa các chương trong quyển sách này.*

TARBALLS

Cách tốt nhất là bạn làm một danh sách của các tập tin trên hệ thống trước khi bạn cài đặt Imap, và công việc tiếp theo đó là so sánh chúng bằng cách dùng lệnh "diff" để tìm ra những tập tin nào được đặt ở đâú. Chỉ đơn giản cho chạy lệnh 'find/*> imap1' trước và 'find/*> imap2' sau khi bạn thiết lập phần mềm và dùng lệnh 'diff imap1 imap2 > imap-installed' để lấy được danh sách của các tập tin đã được thay đổi.

Biên dịch

Giải nén tarball(tar.Z)

```
[root@deep /]# cp imap.tar.Z /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf imap.tar.Z
```

Biên dịch và tối ưu

Di chuyển vào thư mục mới của IMAP/POP và làm theo những lệnh sau:

Bước 1

Hiệu chỉnh tập tin makefile(vi +719 src/osdep/unix/Makefile) và thay đổi dòng:

```
sh -c '(test -f /usr/include/sys/statvfs.h -a $(OS) != sc5 -a $(OS) != sco) &&
$(LN) flocksun.c flockbsd.c || $(LN) flocksv4.c flockbsd.c'
```

Để thành:

```
sh -c '(test -f /usr/include/sys/statvfs.h -a $(OS) != sc5 -a $(OS) != sco -a
$(OS) != lnx) && $(LN) flocksun.c flockbsd.c || $(LN) flocksv4.c flockbsd.c'
```

Sự sửa đổi này sẽ làm thay đổi tập tin "sys/stavfs". tập tin này cùng với thư viện mới glibc2.1 sẽ khác với những tập tin có sẵn ở trên hệ điều hành Sun.

Hiệu chỉnh tập tin Makefile (vi +354 src/osdep/unix/Makefile) và thay đổi dòng:

```
BASECFLAGS="-g -fno-omit-frame-pointer -O6 -DNFSKLUDGE" \
```

Để thành:

```
BASECFLAGS="-g -fno-omit-frame-pointer -O9 -funroll-loops -ffast-math -malign-double - mcpu=pentiumpro -march=pentiumpro -fomit-frame-pointer -fno-exceptions -DNFSKLUDGE" \
```

Đó là những cờ tối ưu cho quá trình biên dịch phần mềm IMAP/POP trên máy chủ.

Hiệu chỉnh tập tin Makefile (vi +61 src/osdep/unix/Makefile) và thay đổi những dòng:

```
ACTIVEFILE=/usr/lib/news/active
```

Để thành:

```
ACTIVEFILE=/var/lib/news/active
SPOOLDIR=/usr/spool
```

Để thành:

```
SPOOLDIR=/var/spool

RSHPATH=/usr/ucb/rsh
```

Để thành:

```
RSHPATH=/usr/bin/rsh

LOCKPGM=/etc/mlock
```

Để thành:

```
#LOCKPGM=/etc/mlock
```

Dòng "ACTIVEFILE=" chỉ rõ đường dẫn của thư mục "đang hoạt động" dành cho IMAP/POP, và "SPOOLDIR=" là nơi để chúng ta đặt thư mục "spool" của Linux IMAP/POP, cũng như "RSHPATH=" sẽ chỉ ra đường dẫn thư mục "rsh" trên hệ thống của chúng. Đây là lưu ý rất quan trọng, cho dù chúng ta không dùng những dịch vụ của rsh trên máy chủ, nhưng chúng ta phải chỉ rõ đúng thư mục của "rsh"

Hiệu chỉnh tập tin Makefile (vi +89 src/osdep/unix/Makefile) và thay đổi những dòng:

```
CC=cc
```

Để thành

```
CC=egcs
```

Dòng này cho biết tên của trình biên dịch GCC của chúng ta và chúng ta sẽ dùng nó để biên dịch phần mềm IMAP/POP.

Bước 2

Bây giờ, chúng ta phải biên dịch và cài đặt IMAP & POP trên Mail Server:

```
[root@deep imap-4.7c]# make lnp
[root@deep imap-4.7c]# install -m 644 ./src/ipopd/ipopd.8c /usr/man/man8/ipopd.8c
[root@deep imap-4.7c]# install -m 644 ./src/imapd/imapd.8c /usr/man/man8/imapd.8c
[root@deep imap-4.7c]# install -s -m 755 ./ipopd/ipop2d /usr/sbin
[root@deep imap-4.7c]# install -s -m 755 ./ipopd/ipop3d /usr/sbin
[root@deep imap-4.7c]# install -s -m 755 ./imapd/imapd /usr/sbin
```

```
[root@deep imap-4.7c]# install -m 644 ./c-client/c-client.a /usr/lib
[root@deep imap-4.7c]# ln -fs /usr/lib/c-client.a /usr/lib/libimap.a
[root@deep imap-4.7c]# mkdir -p /usr/include/imap
[root@deep imap-4.7c]# install -m 644 ./c-client/*.h /usr/include/imap
[root@deep imap-4.7c]# install -m 644 ./src/osdep/tops-20/shortsym.h /usr/include/imap
[root@deep imap-4.7c]# chown root.mail /usr/sbin/ipop2d
[root@deep imap-4.7c]# chown root.mail /usr/sbin/ipop3d
[root@deep imap-4.7c]# chown root.mail /usr/sbin/imapd
```

Những lệnh trên sẽ cấu hình phần mềm đảm bảo hệ thống máy tính của bạn có những chức năng và thư viện cần thiết để biên dịch thành công các package, biên dịch tất cả các tập tin nguồn thành kiểu nhị phân có thể thực thi được, sau đó cài đặt những tập tin nhị phân này và những tập tin phụ của chúng vào những vị trí thích hợp.

Cần lưu ý rằng lệnh "make lnp" ở trên sẽ cấu hình hệ thống Linux của bạn với khả năng Pluggable Authentication Modules (PAM) để cho khả năng bảo mật tốt hơn.

Lệnh "mkdir" sẽ tạo một thư mục mới với tên "imap" dưới thư mục "/usr/include", thư mục mới "imap" sẽ giữ lại tất cả tập tin header liên quan đến chương trình imapd "c-client", và những tập tin "shortsym.h".

Lệnh "chown" sẽ thay đổi quyền sở hữu của các chương trình nhị phân "ipop2d", "ipop3d", và "imapd" được cấp quyền bởi super-user "root", được cấp quyền nhóm bởi "mail".

Lệnh "ln -fs" sẽ tạo một biểu tượng kết nối từ tập tin "c-client.a" đến tập tin "libimap.a" có thể yêu cầu bởi một vài chương trình thứ ba, để bạn có thể cài đặt trong tương lai.

***Chú thích**: Vì nhiều nguyên do bảo mật, nếu bạn chỉ dùng những dịch vụ imapd, gỡ bỏ hai tập tin nhị phân ipop2 và ipop3 khỏi mail server của bạn. Tương tự cho dịch vụ ipopd; nếu bạn chỉ muốn dùng dịch vụ ipopd, thì gỡ bỏ dịch vụ imapd khỏi server Mail và nếu bạn có ý định muốn dùng imapd và popd thì bạn phải giữ lại hai dịch vụ ấy.*

Thu dọn lại sau quá trình cài đặt

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf imap-version/ imap.tar.Z
```

Lệnh "rm" được dùng như ở trên sẽ gỡ bỏ hết các tập tin nguồn mà chúng ta đã dùng để biên dịch và cài đặt IMAP/POP. Nó cũng sẽ gỡ bỏ hết các tập tin nén của IMAP/POP trong thư mục "var/tmp"

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy IMAP/POP server, các tập tin sau được yêu cầu và phải được tạo hoặc sao chép đến các thư mục tương ứng trên server của bạn.

Copy tập tin imap đến thư mục "/etc/pam.d/" nếu bạn có ý định sử dụng dịch vụ imapd này.

Copy tập tin pop đến thư mục "/etc/pam.d/" nếu bạn có ý định sử dụng dịch vụ popd này.

Cấu hình tập tin "/etc/pam.d/imap"

Cấu hình tập tin "/etc/pam.d/imap" dùng sự chứng thực pam.

Tạo tập tin imap (touch /etc/pam.d/imap) và thêm:

```
#%PAM-1.0
auth        required /lib/security/pam_pwdb.so   shadow   nullok
account     required /lib/security/pam_pwdb.so
```

Ghi chú: *Tập tin này chỉ cần thiết trong trường hợp nếu bạn có ý định muốn sử dụng dịch vụ IMAP.*

Cấu hình tập tin "/etc/pam.d/pop"

Cấu hình tập tin "/etc/pam.d/pop" dùng sự chứng thực pam

Tạo tập tin pop (touch /etc/pam.d/pop) và thêm:

```
#%PAM-1.0
auth        required  /lib/security/pam_pwdb.so   shadow   nullok
account     required  /lib/security/pam_pwdb.so
```

Ghi chú: *Tập tin này chỉ cần thiết trong trường hợp nếu bạn có ý định muốn sử dụng dịch vụ POP.*

Cho phép dịch vụ IMAP or POP thông qua tcp-wrappers inetd super server

Tcp-wrappers thận trọng trong việc start và stop dịch vụ IMAP hoặc POP. Khi thực thi những dịch vụ ở trên inetd sẽ đọc những thông tin cấu hình của chính dịch vụ đó từ một tập tin cấu hình mặc định là "/etc/inetd.conf" và các trường trong tập tin cấu hình này sẽ được ngăn cách bởi một tab hoặc một khoảng trắng.

Bước 1

Hiệu chỉnh tập tin inetd.conf (vi /etc/inetd.conf) để thêm vào hoặc bỏ dấu "#" trên dòng liên quan đến nó. Nếu bạn muốn sử dụng IMAP thì bỏ dấu "#" trên dòng imap. Tương tự cho các dịch vụ khác như POP. Trong ví dụ dưới , chúng ta sẽ sử dụng dịch vụ IMAP.

```
#pop-2 stream tcp nowait root /usr/sbin/tcpd ipop2d

#pop-3 stream tcp nowait root /usr/sbin/tcpd ipop3d

imap stream tcp nowait root /usr/sbin/tcpd imapd
```

Đừng quên cập nhật tập tin "inetd.conf" bằng cách gởi tín hiệu SIGHUP (kill all-HUP inetd) sau khi thêm vào những dòng tương ứng.

- Cập nhật tập tin inetd.conf, sử dụng lệnh sau:

```
[root@deep /root]# killall -HUP inetd
```

Bước 2

Nếu bạn muốn cài đặt dịch vụ IMAP/POP theo một cách riêng nào đó và giới hạn máy client tới những địa chỉ IP này, bạn có thể dùng tính năng bảo mật của tcp-wrappers để điều khiển máy nào có thể kết nối tới server từ nơi nào đó. Nếu bạn có dự định cung cấp dịch vụ dial-up cho các client hoặc dịch vụ webmail, bạn không thể sử dụng đặc trưng này.

Hiệu chỉnh tập tin **hosts.deny** (vi /etc/hosts.deny) và thêm vào dòng:

ALL: ALL@ALL, PARANOID

Điều này có nghĩa là, tất cả dịch vụ, tất cả vị trí sẽ bị ngăn chặn, ngoại trừ chúng được gán quyền truy xuất trong tập tin "hosts.allow"

Hiệu chỉnh tập tin **hosts.allow** (vi /etc/hosts.allow) và thêm vào các dòng:

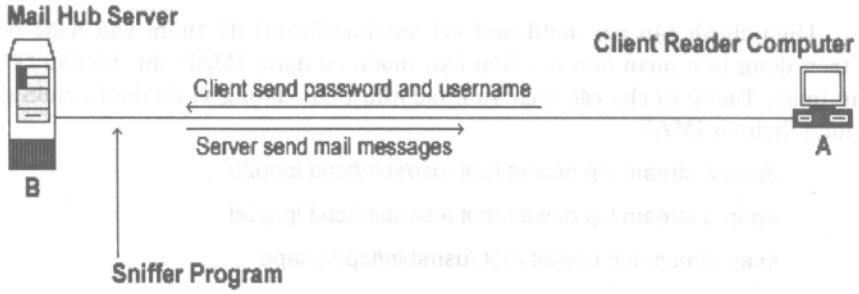
imapd: 216.209.228.34 my.domain.com

Điều này có nghĩa là chỉ có máy khách có IP là "216.209.228.34" với tên server "my.domain.com" thì được phép kết nối và sử dụng dịch vụ IMAP trên máy chủ.

Bảo mật cho IMAP/POP

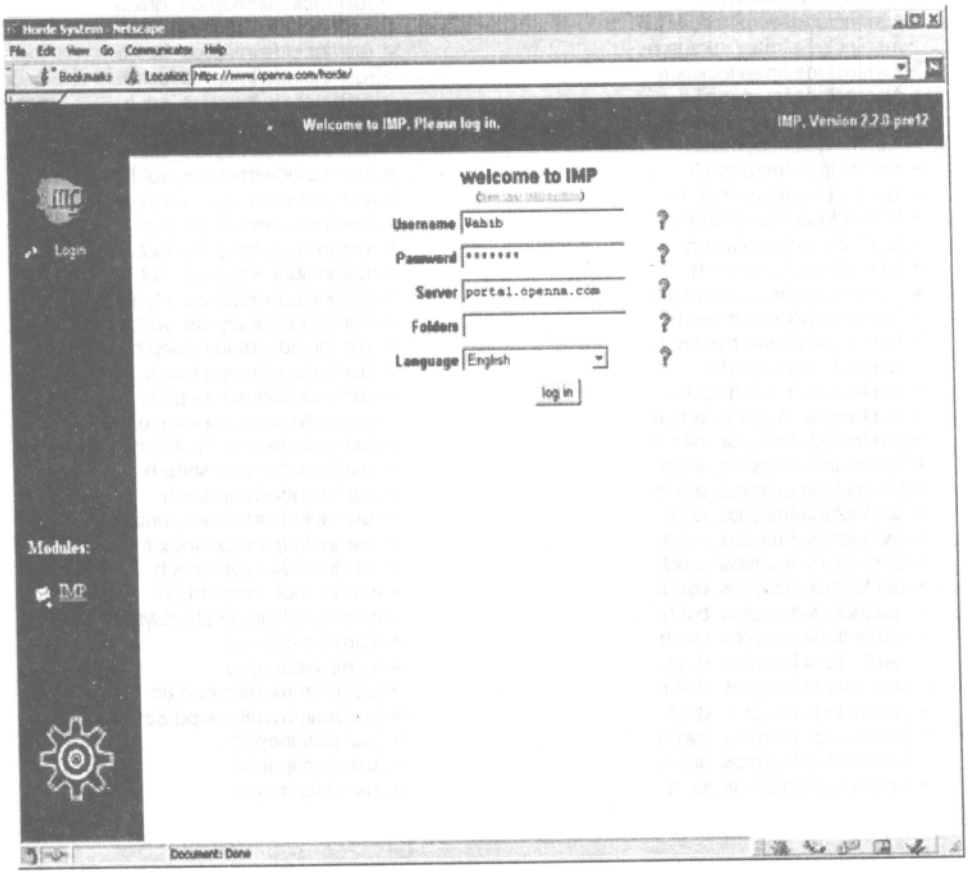
Bạn có thật sự cần dùng dịch vụ IMAP/POP ?

Mặc định chương trình IMAP/POP sử dụng mật khẩu thuần tuý là văn bản. Cho nên bất cứ ai chạy một chương trình sniffer kết nối với mạng của bạn có thể chộp được username/password của bạn và sử dụng chúng để truy xuất vào mail server xem như là bạn đang sử dụng vậy. Điều này không được vì bạn dùng dịch vụ IMAP/POP trên hệ thống LINUX cái mà bạn cần chạy IMAP/POP server một cách cục bộ. Hãy kiểm tra cấu hình máy chủ, nếu bạn sử dụng một server IMAP/POP từ xa thì bạn phải tháo bỏ dịch vụ IMAP/POP trên hệ thống của bạn.



Sử dụng tính năng của SSL trên IMAP/POP

Thật không may, bởi vì hiện nay chính phủ Mỹ đã giới hạn trong việc xuất khẩu công cụ IMAP với tính năng SSL. Có những gói (packages) của các hãng thứ ba cho phép IMAP và POP3 được dùng thông qua giao thức SSL. Một trong những gói đó là webMail IMP, một giao diện web cho phép đọc những thư tín của bạn qua Internet với một trình duyệt web. WebMail IMP dùng giao thức SSL để mã hoá dữ liệu truyền đến server IMAP/POP. Hãy xem phần IV, "Tham khảo các phần mềm liên quan" trong chương 19 "Phần mềm dành cho server (Những dịch vụ mạng trên Web) để nắm bắt được nhiều thông tin hơn về chủ đề này.



Với WebMail IMP, sự trao đổi thông tin từ mail server đến máy khách được mã hóa bởi giao thức SSL thông qua một trình duyệt web.

Tài liệu bổ sung

Để cho chi tiết hơn có một vài trang man mà bạn có thể đọc:

$ man imapd (8C)	- Internet Message Access Protocol server
$ man ipopd (8C)	- Post Office Protocol server

CÀI ĐẶT CÁC TẬP TIN

> /etc/pam.d/imap
> /etc/pam.d/pop
> /usr/include/imap
> /usr/include/imap/dummy.h
> /usr/include/imap/env.h
> /usr/include/imap/env_unix.h
> /usr/include/imap/fdstring.h
> /usr/include/imap/fistring.h
> /usr/include/imap/fs.h
> /usr/include/imap/ftl.h
> /usr/include/imap/imap4r1.h
> /usr/include/imap/linkage.h
> /usr/include/imap/lockfix.h
> /usr/include/imap/mail.h
> /usr/include/imap/mbox.h
> /usr/include/imap/mbx.h
> /usr/include/imap/mh.h
> /usr/include/imap/misc.h
> /usr/include/imap/mmdf.h
> /usr/include/imap/mtx.h
> /usr/include/imap/nx.h
> /usr/include/imap/netmsg.h
> /usr/include/imap/news.h
> /usr/include/imap/newsrc.h
> /usr/include/imap/nl.h
> /usr/include/imap/nntp.h
> /usr/include/imap/os_a32.h
> /usr/include/imap/os_a41.h
> /usr/include/imap/os_aix.h
> /usr/include/imap/os_aos.h
> /usr/include/imap/os_art.h
> /usr/include/imap/os_asv.h
> /usr/include/imap/os_aux.h
> /usr/include/imap/os_bsd.h
> /usr/include/imap/os_bsi.h
> /usr/include/imap/os_cvx.h
> /usr/include/imap/os_d-g.h
> /usr/include/imap/os_do4.h
> /usr/include/imap/os_drs.h
> /usr/include/imap/os_dyn.h
> /usr/include/imap/os_hpp.h
> /usr/include/imap/os_isc.h
> /usr/include/imap/os_lnx.h
> /usr/include/imap/os_lyn.h
> /usr/include/imap/os_mct.h
> /usr/include/imap/os_mnt.h
> /usr/include/imap/os_nxt.h
> /usr/include/imap/os_os4.h
> /usr/include/imap/os_osf.h
> /usr/include/imap/os_ptx.h
> /usr/include/imap/os_pyr.h
> /usr/include/imap/os_qnx.h
> /usr/include/imap/os_s40.h
> /usr/include/imap/os_sc5.h
> /usr/include/imap/os_sco.h
> /usr/include/imap/os_sgi.h
> /usr/include/imap/os_shp.h
> /usr/include/imap/os_slx.h
> /usr/include/imap/os_sol.h
> /usr/include/imap/os_sos.h
> /usr/include/imap/os_sun.h
> /usr/include/imap/os_sv2.h
> /usr/include/imap/os_sv4.h
> /usr/include/imap/os_ult.h
> /usr/include/imap/os_vu2.h
> /usr/include/imap/osdep.h
> /usr/include/imap/phile.h
> /usr/include/imap/pop3.h
> /usr/include/imap/pseudo.h
> /usr/include/imap/rfc822.h
> /usr/include/imap/smtp.h
> /usr/include/imap/tcp.h
> /usr/include/imap/tcp_unix.h
> /usr/include/imap/tenex.h
> /usr/include/imap/unix.h
> /usr/include/imap/utf8.h
> /usr/include/imap/shortsym.h
> /usr/lib/c-client.a
> /usr/lib/libimap.a
> /usr/man/man8/ipopd.8c
> /usr/man/man8/imapd.8c
> /usr/sbin/ipop2d
> /usr/sbin/ipop3d
> /usr/sbin/imapd

Chương 16:

PHẦN MỀM SERVER

(Dịch vụ mã hóa)

Trong chương này chúng ta sẽ đề cập đến các phần sau:

- Linux OpenSSL Server.

  Cấu hình.

  Các lệnh.

  Bảo vệ OpenSSL.

- Linux FreeS/WAN VPN.

  Sự cấu hình.

  Cấu hình RSA private keys secrets.

  Yêu cầu cài đặt Network cho IPSec.

  Kiểm tra sự cài đặt.



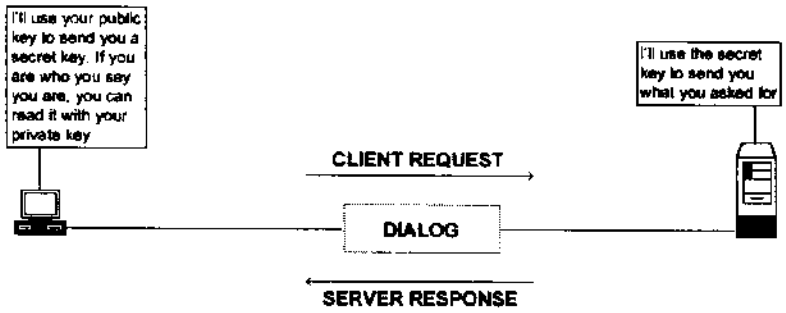
Linux OpenSSL Server

Tổng quan:

Hầu hết các phần mềm như IMAP&POP, Samba, OpenLDAP, FTP, Apache và những phần mềm khác đều yêu cầu công việc kiểm tra tính xác thực (authentication) của người sử dụng trước khi cho phép sử dụng các dịch vụ này, Nhưng mặc định việc truyền tải sự xác minh thông tin người sử dụng và mật khẩu (password) ở dạng văn bản thuần tuý nên có thể được đọc hoặc thay đổi bởi một người khác. Kỹ thuật mã hoá như SSL sẽ đảm bảo tính an toàn và nguyên vẹn của dữ liệu, với kỹ thuật này thông tin truyền trên mạng ở dạng điểm nối điểm được mã hoá. Một khi OpenSSL đã được cài đặt trên Linx server của bạn, bạn có thể sử dụng nó như một công cụ thứ ba cho phép các ứng dụng khác dùng tính năng SSL

Sự diễn giải về OpenSSL:

Dự án OpenSSL là một kết quả của sự cộng tác nhằm phát triển một kỹ thuật bảo mật dạng thương mại, đầy đủ các đặc trưng và là bộ công cụ mã nguồn mở thực thi các giao thức như Secure Sockets Layer (SSL v2/v3) và Transport Layer Security (TLS v1) với những thuật toán mã hoá phức tạp. Dự án được quản lý bởi hiệp hội những người tình nguyện trên thế giới, sử dụng internet để trao đổi thông tin, lập kế hoạch và phát triển công cụ OpenSSL và các tài liệu liên quan khác.



Summary of the Cryptographic Technology

Sự thuận lợi của việc mã hoá:

Những lợi ích chính thu được khi sử dụng công nghệ mã hoá như sau:

Sự tin cẩn dữ liệu

Khi một thư tín (message) được mã hóa, văn bản thuần túy đưa vào được biến đổi bởi một thuật toán thành dạng văn bản được mã hoá, dạng văn bản này che dấu ý nghĩa của message và có thể được gởi thông qua mạng Internet hoặc Intranet. Quá trình này đòi hỏi một khoá riêng (secret key), khoá này dùng để mã hoá dữ liệu và sau đó giải mã giữ liệu. Không có secret key, việc mã hoá dữ liệu là vô nghĩa.

Tính toàn vẹn dữ liệu

Một sự kiểm tra tổng thể mật khẩu được gọi là mã kiểm tra tính xác thực thông điệp (Message Authentication Code) mà trong kỹ thuật mã hoá được gọi là MAC, có thể được tính toán tùy ý trên văn bản người sử dụng cung cấp để bảo vệ tính toàn vẹn của dữ liệu. Kết quả (văn bản và MAC) sau đó được gởi tới người nhận, người nhận kiểm tra nội dung của MAC kèm theo message bằng cách tính toán lại MAC cho message nhận được, sử dụng khoá riêng (secret key) thích hợp và kiểm tra giá trị MAC có bằng chính xác MAC nhận được hay không.

Sự xác thực (Authentication)

Việc xác minh cá nhân là một cách mã hóa khác, ở đây người sử dụng/người gởi biết một secret key, cái mà có thể được dùng để chứng thực nhận dạng của anh ấy/cô ấy.

Chữ ký điện tử (Electronic Signature)

Một chữ ký điện tử đảm bảo người gởi và người nhận rằng message là đáng tin cậy và chỉ có người chủ sở hữu của khoá có thể sinh ra chữ ký điện tử cho người đó.

Việc sáng chế (Patents)

Một vài vấn đề hợp lý sẵn có cho công nghệ SSL. Nếu bạn có ý định sử dụng OpenSSL cho mục đích thương mại bạn cần có một sự cho phép từ tổ chức sáng chế RSA về việc sử dụng các thư viện RSA.

Đây là một đoạn trích trong tập tin README của OpenSSL:

Các công ty khác nhau nắm giữ các bằng sáng chế khác nhau cho những thuật toán khác nhau trong những vị trí khác nhau trên thế giới. Bạn chịu trách nhiệm đảm bảo rằng việc sử dụng bất kỳ thuật toán nào của bạn là hợp lệ bằng cách kiểm tra nếu có bất kỳ bằng sáng chế nào trong quốc gia của bạn hay không. Tập tin này chứa đựng một vài bằng sáng chế mà chúng tôi biết hoặc theo lời giới thiệu sẵn có. Đây không phải là danh sách cuối cùng.

RSA Data Security nắm giữ bằng sáng chế phần mềm trên những thuật toán RSA và RC5. Nếu những thuật toán của họ được dùng bên trong USA và Japan thì phải liên hệ với RSA Data Security cho các điều kiện được phép sử dụng. Trang Web của RSA Data Security có thể truy cập theo địa chỉ http://www.rsa.com/.

RC4 là nhãn hiệu đăng ký của RSA Data Security như vậy việc sử dụng thuật toán này phải được phép của RSA Data Security.

Thuật toán IDEA có bằng sáng chế bởi Ascom ở Áo, Pháp, Đức, Ý, Nhật, Hà Lan, Tật Ban Nha, Thuỵ Điển, Thuỵ Sỹ, Anh và Mỹ. Họ sẽ được liên hệ nếu thuật toán được sử dụng. Tham khảo trang web theo địa chỉ http://www.ascom.ch/.

Hướng dẫn cài đặt

Các dòng lệnh thì tương thích với Unix.

Đường dẫn nguồn sẽ là "/var/tmp" (cũng có thể chọn những đường dẫn khác).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Phiên bản OpenSSL là 0.9.5a.

Dạng lưu trữ Tarballs

Trước hết bạn hãy tạo ra danh sách các tập tin trên hệ thống trước và khi cài đặt OpenSSL, sau đó so sánh chúng dùng lệnh "**diff**" để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh "**find /* > OpenSSL1**" trước và "**find /* > OpenSSL2**" sau khi bạn cài đặt phần mềm, và sử dụng lệnh "**diff OpenSSL1 OpenSSL2 > OpenSSL-Installed**" để lấy một danh sách các thay đổi nào đó.

Các gói cần thiết (Packages)

OpenSSL Homepage Site: http://www.ssh.org.

Bạn phải bảo đảm là tải về gói : openssl-0.9.5a.tar.gz.

Sự biên dịch

Giải nén tập tin dạng tarball (tar.gz).

```
[root@deep /]# cp openssl-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf openssl-version.tar.gz
```

Biên dịch và tối ưu

Chuyển vào thư mục OpenSSL mới và gõ những lệnh sau:

Bước 1:

Hiệu chỉnh tập tin c_rehash (vi +11 tools/c_rehash) và thay đổi dòng.

```
DIR=/usr/local/ssl
```

Trở thành:

```
DIR=/usr
```

Dòng được thay đổi ở trên sẽ xây dựng và cài đặt OpenSSL trong vị trí mặt định "/usr".

Bước 2

Mặc định những tập tin nguồn OpenSSL cho rằng thư mục chương trình Perl của bạn được đặt dưới thư mục "/usr/local/bin/perl". Bạn phải thay đổi dòng

"#!/usr/local/bin/perl" trong tất cả các tập tin script với thư mục perl tương ứng với thư mục perl trên Red Hat Linux "/usr/bin"

```
[root@deep openssl-0.9.5a]# perl util/perlpath.pl /usr/bin (Ở đây chứa đựng chương trình perl của bạn)
```

Bước 3

OpenSSL cần phải biết ở đâu có thể tìm những thư viện nguồn OpenSSL để biên dịch chính xác những tập tin yêu cầu của chúng. Với những lệnh dưới đây chúng ta đặt đường dẫn biến môi trường tới thư mục mặc định, ở đó chúng ta đã giải nén các tập tin nguồn OpenSSL.

```
[root@deep openssl-0.9.5a]# export LD_LIBRARY_PATH=`pwd`
```

Bước 4

Bây giờ chúng ta phải cấu hình OpenSSL cho hệ thống của chúng ta

```
CC="egcs" \
./Configure linux-elf -DSSL_FORBID_ENULL \
--prefix=/usr \
--openssldir=/etc/ssl
```

Chú ý: *Tùy chọn "-DSSL_FORBID_ENULL" yêu cầu không được phép mã hoá rỗng cho những lý do bảo mật.*

Bước 5:

Hiệu chỉnh tập tin Makefile.ssl (vi +50 Makefile.ssl) và thay đổi dòng sau:

```
CC= gcc
```

Trở thành:

```
CC= egcs
```

Hiệu chỉnh tập tin Makefile.ssl (vi +52 Makefile.ssl), thêm/thay đổi những dòng sau:

```
CFLAG= -DTHREADS -D_REENTRANT -DSSL_FORBID_ENULL -DL_ENDIAN -DTERMIO -O9 -funroll-loops-ffast-math -malign-double -mcpu=pentiumpro -march=pentiumpro -fomit-frame-pointer -fno-exceptions-Wall -DSHA1_ASM -DMD5_ASM -DRMD160_ASM
```

Hiệu chỉnh tập tin Makefile.ssl (vi +79 Makefile.ssl) thêm những giá trị sau cho bộ vi xử lý Pentium Pro.

```
PROCESSOR = 686
```

Chú ý: *Cả ba sự thay đổi chúng ta đã làm ở trên sẽ đặt cờ tối ưu cho sự biên dịch của phần mềm OpenSSL trên server. Với sự thay đổi thứ 3 ở trên (PROCESSOR=), nếu bạn đang chạy OpenSSL trên máy Pentium thì hãy đặt giá trị là 586, nếu Pentium Pro/II/III đặt giá trị 686 còn nếu bạn đang chạy trên máy 486 thì giá trị sẽ là 486.*

Bước 6:

Hiệu chỉnh tập tin Makefle.ssl (vi +161 Makefile.ssl) và thay đổi dòng như dưới đây:

```
MANDIR=$(OPENSSLDIR)/man
```

Trở thành:

```
MANDIR=/usr/man
```

Bước này thì cần thiết để thiết đặt thư mục mà chúng ta sẽ cài đặt những trang man của OpenSSL. Với sự thay đổi này chúng ta sẽ cài đặt chúng trong thư mục "/usr/man".

Bước 7:

Bây giờ chúng ta sẽ biên dịch và cài đặt OpenSSL trên server.

```
[root@deep openssl-0.9.5a]# make -f Makefile
[root@deep openssl-0.9.5a]# make test
[root@deep openssl-0.9.5a]# make install
[root@deep openssl-0.9.5a]# mv /etc/ssl/misc/* /usr/bin/
[root@deep openssl-0.9.5a]# rm -rf /etc/ssl/misc/
[root@deep openssl-0.9.5a]# rm -rf /etc/ssl/lib/
[root@deep openssl-0.9.5a]# rm -f /usr/bin/CA.pl
[root@deep openssl-0.9.5a]# rm -f /usr/bin/CA.sh
[root@deep openssl-0.9.5a]# install -m 644 libRSAglue.a /usr/lib/
[root@deep openssl-0.9.5a]# install -m 644 rsaref/rsaref.h /usr/include/openssl/
[root@deep openssl-0.9.5a]# strip /usr/bin/openssl
[root@deep openssl-0.9.5a]# mkdir -p /etc/ssl/crl
```

Lệnh "make –f" sẽ xây dựng những thư viện OpenSSL (libcrypto.a và libssl.a) và tập tin nhị phân OpenSSL "openssl". Các thư viện sẽ được xây dựng trong thư mục gốc của OpenSSL và tập tin nhị phân sẽ được xây dựng trong thư mục "apps" của thư mục gốc của OpenSSL.

Sau khi xây dựng thành công, "make test" sẽ kiểm tra những thư viện, và cuối cùng "make test" sẽ tạo thư mục cài đặt và cài đặt OpenSSL trên server của bạn.

Lệnh "mv" sẽ chuyển tất cả các tập tin trong thư mục "/etc/ssl/misc/" tới thư mục "/usr/bin/". Các tập tin này là các tập tin dạng nhị phân (binary) và phải được đặt trong thư mục "/usr/bin". Mặc định trong hệ thống, tất cả các tập tin nhị phân đều được giữ trong thư mục này. Khi đặt các tập tin trong thư mục này sẽ giữ chúng trong biến môi trường của chúng ta.

Lệnh "rm" sẽ gỡ bỏ các thư mục "/etc/ssl/misc/" và thư mục "/etc/ssl/lib/" từ hệ thống của chúng ta, trước đây những tập tin được đặt trong những thư mục này thì bây giờ nó được đặt trong nơi khác. Cũng như vậy nó sẽ gỡ bỏ các tập tin "CA.pl" và "CA.sh", chúng là những tập tin script nhỏ được sử dụng để tạo các chứng nhận điện tử sở hữu cho chính bạn (CA certificate). Những tập tin script này liên quan đến lệnh "openssl ca" với nhiều yêu cầu phức tạp, và mặc định OpenSSL cấu hình không cho phép một ai dễ dàng sử dụng trực tiếp "openssl ca". Vì vậy chúng ta sẽ tạo tập tin script "sign.sh" để thay thế cho chúng.

Chú ý: *Package bc-1.05a-4.i386.rpm hoặc cao hơn phải được cài đặt sẵn trên Linux Server của bạn hoặc bạn sẽ nhận một thông báo lỗi trong suốt quá trình kiểm tra thư viện của OpenSSL.*

Dọn dẹp sau khi cài đặt:

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf openssl-version/ openssl-version.tar.gz
```

Lệnh "rm" sẽ gỡ bỏ tất cả các tập tin nguồn chúng ta sử dụng để biên dịch và cài đặt OpenSSL. Nó cũng sẽ gỡ bỏ tập tin dạng nén OpenSSL trong thư mục "/var/tmp/".

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Chạy OpenSSL server, những tập tin theo sau được yêu cầu phải tạo mới hoặc được copy tới những thư mục thích hợp trên server của bạn.

  Sao chép tập tin openssl.cnf vào thư mục "/etc/ssl/".

  Sao chép tập tin scriptsign.sh vào thư mục "/usr/bin/".

Bạn có thể thu được những tập tin cấu hình được liệt kê dưới đây từ tập tin dạng nén floppy.tgz. Sao chép những tập tin này từ việc giải nén tập tin floppy.tgz tới những nơi thích hợp hoặc copy hoặc dán trực tiếp chúng từ cuốn sách này tới tập tin có liên quan.

Sự cấu hình của tập tin "/etc/ssl/openssl.cnf"

Đây là tập tin cấu hình chung cho chương trình OpenSSL, ở đó bạn có thể cấu hình ngày hết hạn của khoá của bạn, tên của công ty hoặc tổ chức, địa chỉ ....etc. Những tham số bạn có thể thay đổi sẽ ở trong [CA_default] và đặc biệt phần [req_distinguished_name]

Hiệu chỉnh tập tin openssl.cnf (vi /etc/ssl/openssl.cnf) và thêm hoặc thay đổi:

```
# OpenSSL example configuration file.
# This is mostly being used for generation of certificate requests.
#
RANDFILE = $ENV::HOME/.rnd
oid_file = $ENV::HOME/.oid
oid_section = new_oids
# To use this configuration taäp tin with the "-extfile" option of the
# "openssl x509" utility, name here the section containing the
# X.509v3 extensions to use:
# extensions =
# (Alternatively, use a configuration file that has only
```

```
# X.509v3 extensions in its main [= default] section.)
[ new_oids ]
# We can add new OIDs in here for use by 'ca' and 'req'.
# Add a simple OID like this:
# testoid1=1.2.3.4
# Or use config taäp tin substitution like this:
# testoid2=${testoid1}.5.6
####################################################################
[ ca ]
default_ca = CA_default # The default ca section
####################################################################
[ CA_default ]

dir             = /etc/ssl              # Where everything is kept
certs           = $dir/certs            # Where the issued certs are kept
crl_dir         = $dir/crl              # Where the issued crl are kept
database        = $dir/ca.db.index      # database index file.
new_certs_dir   = $dir/ca.db.certs      # default place for new certs.
certificate     = $dir/certs/ca.crt     # The CA certificate
serial          = $dir/ca.db.serial     # The current serial number
crl             = $dir/crl.pem          # The current CRL
private_key     = $dir/private/ca.key   # The private key
RANDFILE        = $dir/ca.db.rand       # private random number taäp tin

x509_extensions = usr_cert              # The extentions to add to the cert

# Extensions to add to a CRL. Note: Netscape communicator chokes on V2 CRLs
# so this is commented out by default to leave a V1 CRL.
# crl_extensions = crl_ext

# Extensions to add to a CRL. Note: Netscape communicator chokes on V2 CRLs
# so this is commented out by default to leave a V1 CRL.
# crl_extensions = crl_ext
default_days    = 365                   # how long to certify for
default_crl_days= 30                    # how long before next CRL
default_md      = md5                   # which md to use.
Preserve        = no                    # keep passed DN ordering

# A few difference way of specifying how similar the request should look
# For type CA, the listed attributes must be the same, and the optional
# and supplied fields are just that :-)
```

```
policy          = policy_match

# For the CA policy
[ policy_match ]
countryName                 = match
stateOrProvinceName         = match
organizationName            = match
organizationalUnitName      = optional
commonName                  = supplied
emailAddress                = optional
# For the 'anything' policy
# At this point in time, you must list all acceptable 'object'
# types.
[ policy_anything ]
countryName                 = optional
stateOrProvinceName         = optional
localityName                = optional
organizationName            = optional
organizationalUnitName      = optional
commonName                  = supplied
emailAddress                = optional
####################################################################
[ req ]
default_bits                = 1024
default_keyfile             = privkey.pem
distinguished_name          = req_distinguished_name
attributes                  = req_attributes
x509_extensions     = v3_ca     # The extentions to add to the self signed cert

[ req_distinguished_name ]
countryName                 = Country Name (2 letter code)
countryName_default         = CA
countryName_min             = 2
countryName_max             = 2

stateOrProvinceName         = State or Province Name (full name)
stateOrProvinceName_default = Quebec

localityName                = Locality Name (eg, city)
localityName_default        = Montreal
```

```
0.organizationName              = Organization Name (eg, company)
0.organizationName_default      = Open Network Architecture

# we can do this but it is not needed normally :-)
#1.organizationName             = Second Organization Name (eg, company)
#1.organizationName_default     = World Wide Web Pty Ltd

organizationalUnitName          = Organizational Unit Name (eg, section)
organizationalUnitName_default  = Internet Department

commonName                      = Common Name (eg, YOUR name)
commonName_default              = www.openna.com
commonName_max                  = 64
emailAddress                    = Email Address
emailAddress_default            = admin@openna.com
emailAddress_max                = 40

# SET-ex3                       = SET extension number 3

[ req_attributes ]
challengePassword               = A challenge password
challengePassword_min           = 4
challengePassword_max           = 20

unstructuredName                = An optional company name

[ usr_cert ]

# These extensions are added when 'ca' signs a request.

# This goes against PKIX guidelines but some CAs do it and some software
# requires this to avoid interpreting an end user certificate as a CA.

basicConstraints=CA:FALSE
# Here are some examples of the usage of nsCertType. If it is omitted
# the certificate can be used for anything *except* object signing.

# This is OK for an SSL server.
# nsCertType = server

# For an object signing certificate this would be used.
```

```
# nsCertType = objsign

# For normal client use this is typical
# nsCertType = client, email

# and for everything including object signing:
# nsCertType = client, email, objsign

# This is typical in keyUsage for a client certificate.
# keyUsage = nonRepudiation, digitalSignature, keyEncipherment

# This will be displayed in Netscape's comment listbox.
nsComment = "OpenSSL Generated Certificate"

# PKIX recommendations harmless if included in all certificates.
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer:always

# This stuff is for subjectAltName and issuerAltname.
# Import the email address.
# subjectAltName=email:copy

# Copy subject details
# issuerAltName=issuer:copy

#nsCaRevocationUrl        = http://www.domain.dom/ca-crl.pem
#nsBaseUrl
#nsRevocationUrl
#nsRenewalUrl
#nsCaPolicyUrl
#nsSslServerName

[ v3_ca]

# Extensions for a typical CA

# PKIX recommendation.

subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid:always,issuer:always
```

```
# This is what PKIX recommends but some broken software chokes on critical
# extensions.
#basicConstraints = critical,CA:true
# So we do this instead.
basicConstraints = CA:true

# Key usage: this is typical for a CA certificate. However since it will
# prevent it being used as an test self-signed certificate it is best
# left out by default.
# keyUsage = cRLSign, keyCertSign

# Some might want this also
# nsCertType = sslCA, emailCA

# Include email address in subject alt name: another PKIX recommendation
# subjectAltName=email:copy
# Copy issuer details
# issuerAltName=issuer:copy

# RAW DER hex encoding of an extension: beware experts only!

# 1.2.3.5=RAW:02:03
# You can even override a supported extension:
# basicConstraints= critical, RAW:30:03:01:01:FF

[ crl_ext ]
# CRL extensions.
# Only issuerAltName and authorityKeyIdentifier make any sense in a CRL.

# issuerAltName=issuer:copy
authorityKeyIdentifier=keyid:always,issuer:always
```

Chú ý: *File "openssl.cnf" sẵn có trên server của bạn khi bạn biên dịch và cài đặt chương trình OpenSSL và có thể tìm thấy trong thư mục "/etc/ssl". Bạn không cần thay đổi tất cả các tùy chọn mặc định được thiết đặt trong filenày. Sự cấu hình bạn có thể thay đổi chỉ trong phần [CA_default] và [req_distinguished_name]*

Tạo tập tin chương trình sign.sh (touch /usr/bin/sigm.sh) và thêm vào tập tin này:

```
#!/bin/sh
##
## sign.sh -- Sign a SSL Certificate Request (CSR)
```

```
## Copyright (c) 1998-1999 Ralf S. Engelschall, All Rights Reserved.
##

# argument line handling
CSR=$1
if [ $# -ne 1 ]; then
echo "Usage: sign.sign <whatever>.csr"; exit 1
fi
if [ ! -f $CSR ]; then
echo "CSR not found: $CSR"; exit 1
fi
case $CSR in
     *.csr ) CERT="`echo $CSR | sed -e 's/\.csr/.crt/'`" ;;
* ) CERT="$CSR.crt" ;;
esac

# make sure environment exists
if [ ! -d ca.db.certs ]; then
       mkdir ca.db.certs
fi
if [ ! -f ca.db.serial ]; then
       echo '01' >ca.db.serial
fi
if [ ! -f ca.db.index ]; then
       cp /dev/null ca.db.index
fi

# create an own SSLeay config
cat >ca.config <<EOT
[ ca ]
default_ca = CA_own

[ CA_own ]
dir                     = /etc/ssl
certs                   = /etc/ssl/certs
new_certs_dir           = /etc/ssl/ca.db.certs
database                = /etc/ssl/ca.db.index
serial                  = /etc/ssl/ca.db.serial
RANDFILE                = /etc/ssl/ca.db.rand
certificate             = /etc/ssl/certs/ca.crt
private_key             = /etc/ssl/private/ca.key
```

```
default_days            = 365
default_crl_days        = 30
default_md              = md5
preserve                = no
policy                  = policy_anything
[ policy_anything ]
countryName             = optional
stateOrProvinceName     = optional
localityName            = optional
organizationName        = optional
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional
EOT

# sign the certificate
echo "CA signing: $CSR -> $CERT:"
openssl ca -config ca.config -out $CERT -infiles $CSR
echo "CA verifying: $CERT <-> CA cert"
openssl verify -CAfile /etc/ssl/certs/ca.crt $CERT

# cleanup after SSLeay
rm -f ca.config
rm -f ca.db.serial.old
rm -f ca.db.index.old

# die gracefully
exit 0
```

Bây giờ thay đổi quyền mặt định của tập tin này và đặt chúng có quyền có thể thực thi được:

```
[root@deep /]# chmod 755 /usr/bin/sign.sh
```

Chú ý: *Bạn có thể tìm tập tin chương trình này "sign.sh" trong sự phân bố mod_ssl trong thư mục con "mod_ssl-version/pkg.contrib" hoặc trong tập tin nén floppy.tgz. Cần lưu ý rằng phải thay đổi phần [CA_own] để tương ứng với chính môi trường của bạn và cũng đừng quên thay đổi dòng "openssl verify –CAfile /etc/ssl/certs/ca.crt $CERT".*

Các lệnh

Các lệnh được liệt kê bên dưới là một số lệnh mà chúng ta thường dùng, nhưng hiện nay cũng đã được bổ sung thêm nhiều lệnh hơn, và bạn nên kiểm tra trang man và tài liệu để biết thêm chi tiết.

Ví dụ dưới đây sẽ trình bày cho bạn biết làm thế nào để tạo các chứng nhận (Certificate) cho Web Server Apache của bạn và thẩm quyền chứng nhận của chính bạn mà chúng ta thường gọi là CA (Certifying Authority) cho các yêu cầu ký chứng nhận (Certificate Signing Request).

Chú ý: *Tất cả các lệnh được liệt kê dưới đây giả sử được thực hiện trong thư mục "/etc/ssl/".*

1. Tạo một khoá riêng RSA (private key) được bảo vệ với một mật khẩu "passphrase" cho Web Server Apache của bạn.

```
[root@deep ssl]# openssl genrsa -des3 -out server.key 1024
Generating RSA private key, 1024 bit long modulus
.....++++++
.....++++++
e is 65537 (0x10001)
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
```

Sau khi thực hiện xong lưu tập tin server.key tới một nơi an toàn và nhớ pass-phrase bạn vừa mới gõ vào.

2. Sinh ra một yêu cầu ký chứng nhận (CSR) với khoá riêng server RSA.

```
[root@deep ssl]# openssl req -new -key server.key -out server.csr
Using configuration from /etc/ssl/openssl.cnf
Enter PEM pass phrase:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [CA]: VN
State or Province Name (full name) [Quebec]: District3
Locality Name (eg, city) [Montreal]: HCMC
Organization Name (eg, company) [Open Network Architecture]: CyberSoft-VN
Organizational Unit Name (eg, section) [internet Department]: Security Department
Common Name (eg, YOUR name) [www.openna.com]: www.globalcybersoft.com
Email Address [admin@openna.com]: tungt@cybersoft-vn.com
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:.
An optional company name []:.
```

Đảm bảo rằng bạn gõ vào FQDN ("Fully Qualified Domain Name") của server khi OpenSSL nhắc nhở bạn gõ vào "CommonName" (Ví dụ khi bạn tạo ra một CSR cho một web server được phép truy cập thông qua http://www.mydomain.com thì ở đây bạn sẽ gõ vào www.mydomain.com). Như trong trường hợp trên tôi gõ vào www.globalcybersoft.com.

Sau khi sinh ra Certificate Signing Request (CSR), bạn có hai lựa chọn:

Thứ nhất là gởi certificate này đến người có thẩm quyền chứng thực thương mại (commercial Certifying Authority) giống như Verisign hoặc Thawte để ký xác nhận certificate này là của chính bạn. Bạn thường phải gởi thông tin CSR vào trong đơn đăng ký dạng web, trả tiền ký chứng nhận và chờ certificate được ký và lưu nó vào trong tập tin "cerver.crt". Đây là một certificate thật và có thể được sử dụng cho Apache.

Thứ hai, bạn có thể dùng chính CA của chính bạn và bây giờ bạn phải ký (sign) CSR của chính bạn bằng CA này. Giải pháp này thì tiết kiệm và cho phép một công ty, tổ chức quản lý chính server CA của chính họ và tạo ra nhiều certificate khi họ cần cho việc sử dụng bên trong nội bộ mà không trả bất kỳ một khoản tiền nào cho các CA thương mại. Có một điều là dùng chính CA của chính bạn tạo ra các certificate gây nên những vấn đề trong thương mại điện tử bởi vì khách hàng không có sự tin tưởng certificate được ký bởi CA của chính bạn so với khi nó được ký với các CA thương mại. Phần dưới đây trình bày làm thế nào ký một CSR với CA của chính bạn.

3. Tạo một khoá riêng RSA cho CA của bạn.

```
[root@deep ssl]# openssl genrsa -des3 -out ca.key 1024
Generating RSA private key, 1024 bit long modulus
.....++++++
.....++++++
e is 65537 (0x10001)
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
```

Sau khi thực hiện xong lưu tập tin ca.key tới một nơi an toàn và nhớ pass-phrase bạn vừa mới gõ vào.

4. Tạo một certificate được ký bởi chính CA (self-signed certificate) (cấu trúc x509) với khoá RSA của CA.

```
[root@deep ssl]# openssl req -new -x509 -days 365 -key ca.key -out ca.crt
Using configuration from /etc/ssl/openssl.cnf
Enter PEM pass phrase:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
```

```
-----
Country Name (2 letter code) [CA]: VN
State or Province Name (full name) [Quebec]: DISTRICT3
Locality Name (eg, city) [Montreal]: HCMC
Organization Name (eg, company) [Open Network Architecture]: CYBERSOFT
Organizational Unit Name (eg, section) [Internet Department]:CA Marketing
Common Name (eg, YOUR name) [www.openna.com]: www.globalcybersoft.com
Email Address [admin@openna.com]: support@globalcybersoft.com

[root@deep ssl]# mv server.key private/
[root@deep ssl]# mv ca.key private/
[root@deep ssl]# mv ca.crt certs/
```

Chú ý: *Lệnh "req" tạo một chứng nhận được ký bởi chính nó (self-signed certificate) khi tham số –x509 được sử dụng.*

5. Ký một Certificate request

Chuẩn bị script cho việc ký (việc này cần thiết bởi vì lệnh "openssl ca" có một vài tham số phức tạp và mặc định cấu hình OpenSSL không cho phép một ai dễ dàng sử dụng trực tiếp "openssl ca"). tập tin scriptvới tên sign.sh thì đã sẵn có trong thư mục openssl trong đĩa mềm kèm theo sách này.

Bây giờ bạn có thể dùng CA này ký CSR của server để tạo SSL certificate thật sự để dùng trong Webserver Apache.

```
[root@deep ssl]# /usr/bin/sign.sh server.csr
CA signing: server.csr -> server.crt:
Using configuration from ca.config
Enter PEM pass phrase:
Check that the request matches the signature
Signature ok
The Subjects Distinguished Name is as follows
countryName :PRINTABLE:'CA'
stateOrProvinceName :PRINTABLE:'Quebec'
localityName :PRINTABLE:'Montreal'
organizationName :PRINTABLE:'Open Network Architecture'
organizationalUnitName :PRINTABLE:'Internet Department'
commonName :PRINTABLE:'www.openna.com'
emailAddress :IA5STRING:'admin@openna.com'
Certificate is to be certified until Dec 1 14:59:29 2000 GMT (365 days)
Sign the certificate? [y/n]:y
1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
CA verifying: server.crt <-> CA cert
```

server.crt: OK

Lệnh trên ký vào CSR và kết quả sẽ được lưu trong tập tin server.crt

```
[root@deep ssl]# mv server.crt certs/
```

Bây giờ chúng ta có hai file: server.key và server.crt. Thay đổi tập tin cấu hình của Apache sử dụng hai tập tin này như sau:

```
SSLCertificateFile /etc/ssl/certs/server.crt          # Our web server public key
SSLCertificateKeyFile /etc/ssl/private/server.key     # Our web server private key
```

File server.csr thì không cần nữa và bạn có thể xoá

```
[root@deep ssl]# rm -f server.csr
```

Chú ý: *Nếu bạn nhận một thông báo lỗi trong việc ký xác nhận certificate, thì hầu như chắc chắn là bạn đã gõ vào sai FQDN ("Fully Qualified Domain Name") cho server trong khi OpenSSL đợi bạn gõ vào "CommonName". Bạn cần gõ vào chính xác FQDN, ví dụ như my.domain.com không thể gõ vào domain.com. Cũng thế, khi bạn tạo ra hai certificate và CA certificate, điều quan trọng là tối thiểu một phần của thông tin không giống giữa hai tập tin với nhau, hoặc bạn có thể bắt gặp những vấn đề trong lúc ký nhận certificate.*

Bảo vệ OpenSSL

Làm cho các khoá của bạn chỉ có thể "đọc và viết" bởi user root. Điều này quan trọng bởi vì không một ai đụng đến tập tin này.

- Làm cho các khoá của bạn chỉ có thể đọc viết bởi root, dùng những lệnh sau:

```
[root@deep /]# chmod 600 /etc/ssl/certs/ca.crt
[root@deep /]# chmod 600 /etc/ssl/certs/server.crt
[root@deep /]# chmod 600 /etc/ssl/private/ca.key
[root@deep /]# chmod 600 /etc/ssl/private/server.key
```

Việc sử dụng của phần mềm OpenSSL

OpenSSL có thể được sử dụng để:

1. Tạo Certificate Server cho web server của bạn.
2. Cung cấp sự tin cẩn, tính toàn vẹn, tính xác thực của dữ liệu và chữ ký điện tử trong truyền tải thông tin cho các user.
3. Bảo vệ việc thực hiện giao dịch điện tử.

Những tập tin được cài đặt

```
> /etc/ssl                          > /usr/man/man1/verify.1
> /etc/ssl/crl                      > /usr/man/man1/version.1
> /etc/ssl/certs                    > /usr/man/man1/x509.1
> /etc/ssl/private                  > /usr/man/man3/BN_CTX_new.3
> /etc/ssl/openssl.cnf              > /usr/man/man3/BN_CTX_start.3
```

> /usr/bin/openssl
> /usr/bin/c_rehash
> /usr/bin/sign.sh
> /usr/bin/c_hash
> /usr/bin/c_info
> /usr/bin/c_issuer
> /usr/bin/c_name
> /usr/bin/der_chop
> /usr/include/openssl
> /usr/include/openssl/e_os.h
> /usr/include/openssl/e_os2.h
> /usr/include/openssl/crypto.h
> /usr/include/openssl/tmdiff.h
> /usr/include/openssl/opensslv.h
> /usr/include/openssl/opensslconf.h
> /usr/include/openssl/ebcdic.h
> /usr/include/openssl/md2.h
> /usr/include/openssl/md5.h
> /usr/include/openssl/sha.h
> /usr/include/openssl/mdc2.h
> /usr/include/openssl/hmac.h
> /usr/include/openssl/ripemd.h
> /usr/include/openssl/des.h
> /usr/include/openssl/rc2.h
> /usr/include/openssl/rc4.h
> /usr/include/openssl/rc5.h
> /usr/include/openssl/idea.h
> /usr/include/openssl/blowfish.h
> /usr/include/openssl/cast.h
> /usr/include/openssl/bn.h
> /usr/include/openssl/rsa.h
> /usr/include/openssl/dsa.h
> /usr/include/openssl/dh.h
> /usr/include/openssl/buffer.h
> /usr/include/openssl/bio.h
> /usr/include/openssl/stack.h
> /usr/include/openssl/safestack.h
> /usr/include/openssl/lhash.h
> /usr/include/openssl/rand.h
> /usr/include/openssl/err.h
> /usr/include/openssl/objects.h
> /usr/include/openssl/evp.h
> /usr/include/openssl/asn1.h
> /usr/include/openssl/asn1_mac.h
> /usr/include/openssl/pem.h
> /usr/man/man3/BN_add.3
> /usr/man/man3/BN_add_word.3
> /usr/man/man3/BN_bn2bin.3
> /usr/man/man3/BN_cmp.3
> /usr/man/man3/BN_copy.3
> /usr/man/man3/BN_generate_prime.3
> /usr/man/man3/BN_mod_inverse.3
> /usr/man/man3/BN_mod_mul_montgomery.3
> /usr/man/man3/BN_mod_mul_reciprocal.3
> /usr/man/man3/BN_new.3
> /usr/man/man3/BN_num_bytes.3
> /usr/man/man3/BN_rand.3
> /usr/man/man3/BN_set_bit.3
> /usr/man/man3/BN_zero.3
> /usr/man/man3/CRYPTO_set_ex_data.3
> /usr/man/man3/DH_generate_key.3
> /usr/man/man3/DH_generate_parameters.3
> /usr/man/man3/DH_get_ex_new_index.3
> /usr/man/man3/DH_new.3
> /usr/man/man3/DH_set_method.3
> /usr/man/man3/DH_size.3
> /usr/man/man3/DSA_SIG_new.3
> /usr/man/man3/DSA_do_sign.3
> /usr/man/man3/DSA_dup_DH.3
> /usr/man/man3/DSA_generate_key.3
> /usr/man/man3/DSA_generate_parameters.3
> /usr/man/man3/DSA_get_ex_new_index.3
> /usr/man/man3/DSA_new.3
> /usr/man/man3/DSA_set_method.3
> /usr/man/man3/DSA_sign.3
> /usr/man/man3/DSA_size.3
> /usr/man/man3/ERR_GET_LIB.3
> /usr/man/man3/ERR_clear_error.3
> /usr/man/man3/ERR_error_string.3
> /usr/man/man3/ERR_get_error.3
> /usr/man/man3/ERR_load_crypto_strings.3
> /usr/man/man3/ERR_load_strings.3
> /usr/man/man3/ERR_print_errors.3
> /usr/man/man3/ERR_put_error.3
> /usr/man/man3/ERR_remove_state.3
> /usr/man/man3/EVP_DigestInit.3
> /usr/man/man3/EVP_EncryptInit.3
> /usr/man/man3/OPENSSL_VERSION_NUMBER.3
> /usr/man/man3/OpenSSL_add_all_algorithms.3
> /usr/man/man3/RAND_add.3

> /usr/include/openssl/pem2.h
> /usr/include/openssl/x509.h
> /usr/include/openssl/x509_vfy.h
> /usr/include/openssl/x509v3.h
> /usr/include/openssl/conf.h
> /usr/include/openssl/txt_db.h
> /usr/include/openssl/pkcs7.h
> /usr/include/openssl/pkcs12.h
> /usr/include/openssl/comp.h
> /usr/include/openssl/ssl.h
> /usr/include/openssl/ssl2.h
> /usr/include/openssl/ssl3.h
> /usr/include/openssl/ssl23.h
> /usr/include/openssl/tls1.h
> /usr/include/openssl/rsaref.h
> /usr/lib/libcrypto.a
> /usr/lib/libssl.a
> /usr/lib/libRSAglue.a
> /usr/man/man1/CA.pl.1
> /usr/man/man1/asn1parse.1
> /usr/man/man1/ca.1
> /usr/man/man1/ciphers.1
> /usr/man/man1/crl.1
> /usr/man/man1/crl2pkcs7.1
> /usr/man/man1/dgst.1
> /usr/man/man1/dhparam.1
> /usr/man/man1/dsa.1
> /usr/man/man1/dsaparam.1
> /usr/man/man1/enc.1
> /usr/man/man1/gendsa.1
> /usr/man/man1/genrsa.1
> /usr/man/man1/nseq.1
> /usr/man/man1/openssl.1
> /usr/man/man1/pkcs12.1
> /usr/man/man1/pkcs7.1
> /usr/man/man1/pkcs8.1
> /usr/man/man1/req.1
> /usr/man/man1/rsa.1
> /usr/man/man1/s_client.1
> /usr/man/man1/s_server.1
> /usr/man/man1/sess_id.1
> /usr/man/man1/smime.1
> /usr/man/man1/speed.1
> /usr/man/man1/spkac.1
> /usr/man/man3/RAND_bytes.3
> /usr/man/man3/RAND_cleanup.3
> /usr/man/man3/RAND_egd.3
> /usr/man/man3/RAND_load_file.3
> /usr/man/man3/RAND_set_rand_method.3
> /usr/man/man3/RSA_blinding_on.3
> /usr/man/man3/RSA_check_key.3
> /usr/man/man3/RSA_generate_key.3
> /usr/man/man3/RSA_get_ex_new_index.3
> /usr/man/man3/RSA_new.3
> /usr/man/man3/RSA_padding_add_PKCS1_type_1.3
> /usr/man/man3/RSA_print.3
> /usr/man/man3/RSA_private_encrypt.3
> /usr/man/man3/RSA_public_encrypt.3
> /usr/man/man3/RSA_set_method.3
> /usr/man/man3/RSA_sign.3
> /usr/man/man3/RSA_sign_ASN1_OCTET_STRING.3
> /usr/man/man3/RSA_size.3
> /usr/man/man3/blowfish.3
> /usr/man/man3/bn.3
> /usr/man/man3/bn_internal.3
> /usr/man/man3/buffer.3
> /usr/man/man3/crypto.3
> /usr/man/man3/d2i_DHparams.3
> /usr/man/man3/d2i_RSAPublicKey.3
> /usr/man/man3/dh.3
> /usr/man/man3/dsa.3
> /usr/man/man3/err.3
> /usr/man/man3/hmac.3
> /usr/man/man3/lh_stats.3
> /usr/man/man3/lhash.3
> /usr/man/man3/md5.3
> /usr/man/man3/mdc2.3
> /usr/man/man3/rand.3
> /usr/man/man3/rc4.3
> /usr/man/man3/ripemd.3
> /usr/man/man3/rsa.3
> /usr/man/man3/sha.3
> /usr/man/man3/threads.3
> /usr/man/man3/SSL_get_error.3
> /usr/man/man3/ssl.3
> /usr/man/man5/config.5
> /usr/man/man7/des_modes.7

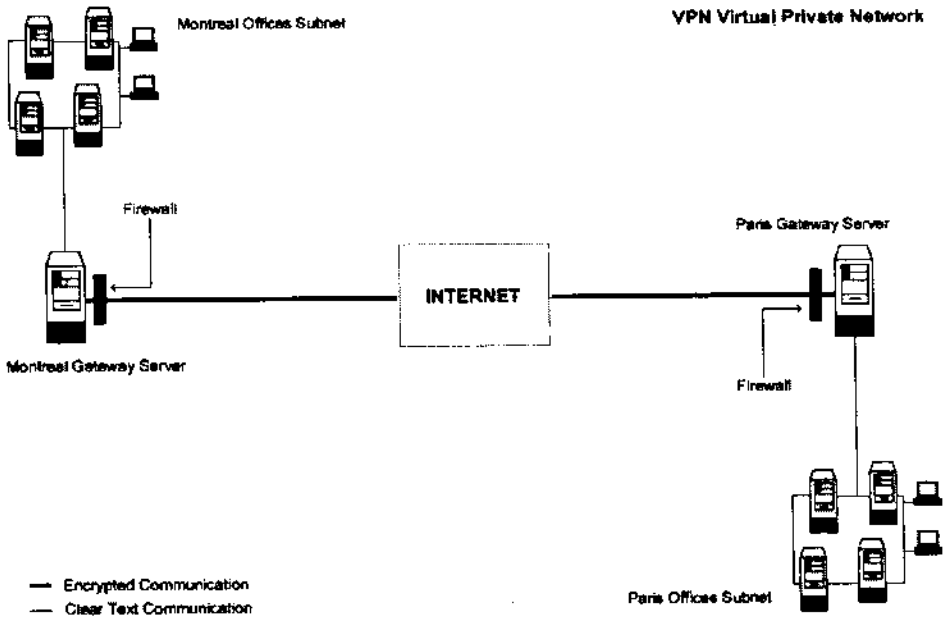
LINUX FreeS/WAN VPN

Tổng quan

Bảo vệ sự trao đổi thông tin giữa client và server với giải pháp SSL là một sự lựa chọn đúng đắn nhưng đôi khi với những môi trường khó khăn để thiết lập kênh trao đổi thông tin bảo mật đảm bảo đầy đủ sự bí mật, tính xác thực, và tính nguyên vẹn của dữ liệu giữa hai bức tường lửa (firewall) trên Internet là vấn đề quan trọng. Vì lý do này IPSEC được đưa ra.

IPSec là Internet Protocol SECurity. Nó sử dụng mật khẩu phức tạp để cung cấp hai dịch vụ kiểm tra tính xác thực (authetication) và mã hoá (encryption). Dịch vụ Authentication bảo đảm rằng những gói dữ liệu (package) đến từ đúng người gởi và không bị thay đổi trong quá trình truyền tải. Dịch vụ encryption ngăn cấm những người không có quyền đọc được nội dung gói dữ liệu. IPSec có thể bảo vệ bất kỳ protocol đang chạy trên IP và bất kỳ sự trung gian sử dụng IP. IPSec cũng có thể cung cấp một vài dịch vụ bảo mật ở phía sau "background" không nhìn thấy được tác động tới người sử dụng. IPSec cũng có thể bảo vệ một sự hỗn hợp của các protocol đang chạy trên một sự kết hợp phức tạp của nhiều môi trường (ví dụ IMAP/POP3) không phải thay đổi chúng trong bất kỳ đường đi nào, từ khi sự mã hoá xuất hiện ở tầng IP.

Những dịch vụ IPSec cho phép bạn xây dựng những đường hầm bảo mật thông qua mạng không an toàn. Mọi thứ đi xuyên qua mạng không an toàn Internet thì được mã hoá bởi máy gateway IPSEC và được giải mã bởi gateway khác ở đầu cuối. Kết quả là Virtual Private Network hoặc VPN được thiết lập. Đây là một mạng máy tính, một mạng riêng cho dù nó bao gồm các máy tính ở những nơi khác nhau được kết nối bởi mạng không an toàn internet.



Hướng dẫn cài đặt

- Các lệnh thì tương thích với Unix.
- Đường dẫn nguồn là "/usr/src".
- Sự cài đặt đã được kiểm tra trên Red Hat Linux 6.1 và 6.2.
- Tất cả các bước trong tiến trình cài đặt sẽ xảy ra trong supper-user root.
- Version của Kernel là 2.2.14.
- Version của kernel là 1.3 (khi cuốn sách này đến tay bạn thì version hiện tại là 1.9.4. Bạn đừng bận tâm về vấn đề này, với version 1.9.4 thì yêu cầu version của kernel là 2.4.5.

Các gói dữ liệu

Kernel Homepage: http://www.kernelnotes.org/.

Bạn phải đảm bảo tải về: linux-2_2_14_tar.gz.

FreeS/WAN VPN Homepage Site: http://www.freeswan.org/.

FreeS/WAN VPN FTP Site: 194.109.6.26.

Bạn phải đảm bảo tải về: freeswan-1.91.tar.gz.

Dạng lưu trữ Tarballs

Trước hết hãy tạo ra danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt FreeS/WAN, và sau đó so sánh chúng dùng lệnh "**diff**" để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh "**find /* >** Freeswan1" trước và "**find /* >** Freeswan2" sau khi bạn cài đặt phần mềm, và sử dụng lệnh "**diff** Freeswan1 Freeswan**2 >** Freeswan-**Installed**" để lấy một danh sách các thay đổi nào đó.

Yêu cầu ban đầu

Sự cài đặt phần mềm mạng riêng ảo (Virtual Private Network) IPSec FreeS/WAN yêu cầu một vài sự thay đổi của kernel nguyên thủy của bạn. Trước khi sử dụng được Freeswan nó phải được đưa vào và được kết hợp chặt chẽ trong kernel của bạn. Với lý do này bước đầu tiên trong tiến trình cài đặt phần mềm FreeS/WAN là xem lại phần Linux kernel trong cuốn sách này và theo sự chỉ dẫn làm thế nào cài đặt Linux Kernel trên hệ thống của bạn (ngay cả bạn đã làm điều này trước đây) và quay trở lại phần FreeS/WAN VPN sau khi bạn đã thực thi những lệnh "make dep; male clean" nhưng trước lệnh "make bzImage" trong phần Linux Kernel

Cảnh báo: *Một lời khuyên rằng bạn không biên dịch bất kỳ những gì trong kernel với cờ tối ưu nếu bạn có dự định cài đặt phần mềm FreeSWAN trên hệ thống của bạn. Bất kỳ cờ tối ưu được thêm vào Linux kernel sẽ đưa ra những thông báo lỗi trong phần mềm FreeSWAN IPSEC khi nó thử*

chạy; đây là một cảnh báo quan trọng bạn phải chú ý hoặc không cái gì sẽ làm việc với FreeSWAN. Những cờ tối ưu được mô tả trong chương 5 "Cấu hình và xây dựng một kernel bảo mật, tối ưu" áp dụng không có bất kỳ vấn đề gì đối với những phần và những chương của cuốn sách này ngoại trừ đối với phần mềm FreeSWAN IPSEC. Một lần nữa nhắc lại, đừng sử dụng hoặc thêm bất kỳ tùy chọn hoặc những cờ tối ưu nào vào trong Linux kenel của bạn khi biên dịch và chỉnh sửa kernel để hỗ trợ FreeSWAN.

Sự biên dịch của FreeS/WAN

Giải nén tập tin lưu trữ dạng tar (tar.gz).

```
[root@deep /]# cp freeswan-version.tar.gz /usr/src/
[root@deep /]# cd /usr/src
[root@deep src]# tar xzpf freeswan-version.tar.gz
[root@deep src]# chown -R 0.0 /usr/src/freeswan-version
```

Biên dịch và đưa FreeS/WAN vào Kernel

Bạn phải thay đổi "Makefile" dưới thư mục nguồn FreeS/WAN và các thư mục con như "utils", "klips/utils", "Pluto" và "lib" theo các đường dẫn tiến trình cài đặt chỉ định. Chúng phải thay đổi những tập tin này theo với cấu trúc hệ thống tập tin của Red Hat và cài đặt những tập tin dưới biến môi trường PATH của chúng ta.

Bước 1:

Chuyển vào thư mục gốc của FreeS/WAN và gõ vào những lệnh sau:

Hiệu chỉnh tập tin Makefile (vi Makefile) và thay đổi những dòng sau:

```
PUBDIR=/usr/local/sbin
```

Trở thành

```
PUBDIR=/usr/sbin
```

```
PRIVDIR=/usr/local/lib/ipsec
```

Trở thành

```
PRIVDIR=/usr/lib/ipsec
```

```
FINALPRIVDIR=/usr/local/lib/ipsec
```

Trở thành

```
FINALPRIVDIR=/usr/lib/ipsec
```

```
MANTREE=/usr/local/man
```

Trở thành

```
MANTREE=/usr/man
```

Bước 2

Hiệu chỉnh tập tin Makefile của thư mục con "utils" (vi utils/Makefile) và thay đổi những dòng sau:

```
PUBDIR=/usr/local/sbin
```

Trở thành

```
PUBDIR=/usr/sbin
```

```
PRIVDIR=/usr/local/lib/ipsec
```

Trở thành

```
PRIVDIR=/usr/lib/ipsec
```

```
FINALPRIVDIR=/usr/local/lib/ipsec
```

Trở thành

```
FINALPRIVDIR=/usr/lib/ipsec
```

```
MANTREE=/usr/local/man
```

Trở thành

```
MANTREE=/usr/man
```

Bước 3

Hiệu chỉnh tập tin Makefile của thư mục con "klips/utils" (vi klips/utils/Makefile) và thay đổi những dòng sau:

```
BINDIR=/usr/local/lib/ipsec
```

Trở thành

```
BINDIR=/usr/lib/ipsec
```

```
MANTREE=/usr/local/man
```

Trở thành

```
MANTREE=/usr/man
```

Bước 4

Hiệu chỉnh tập tin Makefile của thư mục con "pluto" (vi pluto/Makefile) và thay đổi những dòng sau:

```
BINDIR=/usr/local/lib/ipsec
```

Trở thành

```
BINDIR=/usr/lib/ipsec
```

```
MANTREE=/usr/local/man
```

Trở thành
MANTREE=/usr/man

Bước 5

Hiệu chỉnh tập tin Makefile của thư mục con "lib" (vi lib/Makefile) và thay đổi những dòng sau:

MANTREE=/usr/local/man

Trở thành
MANTREE=/usr/man

Bước 6

Hiệu chỉnh tập tin Makefile của thư mục con "libdes" (vi libdes/Makefile) và thay đổi những dòng sau:

LIBDIR=/usr/local/lib

Trở thành
LIBDIR=/usr/lib

BINDIR=/usr/local/bin

Trở thành
BINDIR=/usr/bin

INCDIR=/usr/local/include

Trở thành
INCDIR=/usr/include

MANDIR=/usr/local/man

Trở thành
MANDIR=/usr/man

Các thay đổi trên từ bước 1 đến bước 6 sẽ đặt tất cả các tập tin liên quan đến phần mềm FreeS/WAN đến các thư mục đích thích hợp mà chúng ta đã chọn trong thứ tự tuân theo kiến trúc tập tin hệ thống của Red Hat Linux.

Bước 7

Bây giờ chúng ta sẽ biên dịch và cài đặt FreeSWAN trên server:

```
[root@deep freeswan-1.3]# make insert
[root@deep freeswan-1.3]# make programs
[root@deep freeswan-1.3]# make install
```

Lệnh "make insert" tạo một biểu tượng liên kết "/usr/src/linux/net/ipsec" trỏ tới thư mục nguồn KLIPS, nó ráp một vài tập tin kernel ở những nơi cần thiết, biết về KLIPS và/hoặc sửa lỗi. Nó cũng thêm cấu hình mặc định cho tập tin cấu hình kernel và cuối cùng nó tạo tập tin truyền đạt thông tin KLIPS "/dev/ipsec", nếu nó không sẵn có ở đây.

Lệnh "make programs" xây dựng các thư viện, Pluto và những tiện ích mức user. Lệnh "make install" sẽ cài đặt chương trình deamon Pluto và những tiện ích user-level và thiết đặt những công việc cho quá trình startup.

Định lại cấu hình và cài đặt kernel để hỗ trợ FreeS/WAN

Bây giờ chúng ta quay trở lại thư mục "/usr/src/linux" và thực thi những lệnh sau để định lại cấu hình kernel với FreeS/WAN hỗ trợ:

```
[root@deep freeswan-1.3]# cd /usr/src/linux
[root@deep linux]# make config
```

Chú ý: *Khác với lệnh "make config" chúng ta đã sử dụng trước đây và bây giờ phần mới liên quan tới FreeS/WAN được đưa vào trong cấu hình kernel của chúng ta và do lý do này chúng ta phải định hình lại kernel để tùy chỉnh các tùy chọn IPSec của kernel của bạn.*

Điều đầu tiên bạn cần làm là bảo đảm kernel của bạn được xây dựng với hỗ trợ FreeS/WAN. Trong kernel version 2.2.14, một phần mới liên quan tới freeswan VPN với tên là "IPSec options (FreeS/WAN)" sẽ xuất hiện trong cấu hình kernel của bạn sau khi bạn ráp kernel với chương trình FreeS/WAN như được diễn giải ở trên. Bạn cần bảo đảm rằng bạn đã trả lời Y cho những câu hỏi dưới đây (IPSec options (FreeS/WAN)).

IPSec options (FreeS/WAN)
IP Security Protocol (FreeS/WAN IPSEC) (CONFIG_IPSEC) [Y/n/?]
IPSEC: IP-in-IP encapsulation (CONFIG_IPSEC_IPIP) [Y/n/?]
IPSEC: PF_KEYv2 kernel/user interface (CONFIG_IPSEC_PFKEYv2) [Y/n/?]
IPSEC: Enable ICMP PMTU messages (CONFIG_IPSEC_ICMP) [Y/n/?]
IPSEC: Authentication Header (CONFIG_IPSEC_AH) [Y/n/?]
HMAC-MD5 authentication algorithm (CONFIG_IPSEC_AUTH_HMAC_MD5) [Y/n/?]
HMAC-SHA1 authentication algorithm (CONFIG_IPSEC_AUTH_HMAC_SHA1) [Y/n/?]
IPSEC: Encapsulating Security Payload (CONFIG_IPSEC_ESP) [Y/n/?]
3DES encryption algorithm (CONFIG_IPSEC_ENC_3DES) [Y/n/?]
IPSEC Debugging Option (DEBUG_IPSEC) [Y/n/?]

Chú ý: *Tất cả các tùy biến bạn đã làm với kernel lần đầu tiên mà bạn đã chạy các lệnh "make config", "make dep" và "make clean" vẫn duy trì, như vậy bạn không cần phải cấu hình lại mọi thứ trong kernel của bạn, chỉ phần mới được thêm FresS/WAN với tên "IPSec options (FreeS/WAN)" thì được yêu cầu như trình bày ở trên.*

Một vài tùy chọn về Network sẽ được mở một cách tự động ngay cả chúng đã bị tắc ở lần trước đó bởi vì IPSEC cần chúng. Bất cứ cấu hình nào bạn đang sử dụng, bạn nên cẩn thận chú ý đến một vài vấn đề: theo kinh nghiệm cá nhân bạn không nên làm mất tác dụng tùy chọn "Networking Options" của cấu hình kernel của bạn.

```
Kernel/User netlink socket (CONFIG_NETLINK) [Y/n/?]
Netlink device emulation (CONFIG_NETLINK_DEV) [Y/n/?]
```

Biên dịch và cài đặt kernel mới với FreeS/WAN

Chúng ta đã làm cho kernel của chúng ta hỗ trợ FreeS/WAN, bây giờ chúng ta có thể biên dịch và cài đặt kernel mới.

- Quay trở lại thư mục "/usr/src/linux" và chạy những lệnh sau:

```
[root@deep linux]# make dep; make clean; make bzImage
```

Sau khi thực thi các lệnh ở trên, đi theo phần còn lại của sự chỉ dẫn trong phần Nhân Linux (Linux Kernel) của quyển sách này (Chương "Cấu hình và xây dựng một kernel bảo mật, tối ưu") như để cài đặt kernel. Ở thời điểm này giả sử bạn đã copy và cài đặt tập tin kernel mới (bzImage), system.map, hoặc các module (nếu cần thiết) của bạn tới vị trí thích hợp và hiệu chỉnh tập tin lilo.conf để làm việc với kernel mới, bạn cũng cần chỉnh sửa các tập tin cấu hình có liên quan tới FreeS/WAN như "ipsec.conf" và "ipsec.secrets" trước khi khởi động lại hệ thống của bạn.

Dọn dẹp sau khi cài đặt

```
[root@deep /]# cd /usr/src
[root@deep src]# rm -rf freeswan-version/ freeswan-version.tar.gz
```

Lệnh "rm" sẽ loại bỏ tất cả các tập tin nguồn chúng ta đã sử dụng để biên dịch và cài đặt FreeS/WAN. Nó cũng sẽ loại bỏ tập tin nén của FreeS/WAN từ thư mục "/usr/src".

Sự cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy FreeS/WAN, những tập tin theo sau được yêu cầu và phải được tạo hoặc copy tới những thư mục thích hợp trên server.

  Sao chép tập tin ipsec.conf tới thư mục "/etc/".

  Sao chép tập tin ipsec.secrets tới thư mục "/etc/"

Cấu hình tập tin "/etc/ipsec.conf"

File cấu hình "/etc/ipsec.conf" của FreeS/WAN cho phép bạn cấu hình IPSEC, điều khiển thông tin và các kiểu kết nối. Hiện tại IPSec hỗ trợ hai kiểu kết nối: Khoá điều khiển bằng tay (Manually keyed) và khoá tự động

(Automatically keyed). Phương cách làm việc của hai kiểu kết nối này hoàn toàn khác nhau, sự kết nối Manually keyed sử dụng những khoá được lưu trữ trong tập tin "/etc/ipsec.conf". Kiểu của sự kết nối này bảo mật kém hơn Automatically keyed. Sự kết nối Automatically keyed sử dụng những khoá được sinh ra một cách tự động bởi chương trình thỏa thuận khoá Pluto. Giao thức thoả thuận khoá được sử dụng mặc định với tên là IKE, kiểm tra tính xác thực với hệ thống khác sử dụng một khoá riêng biệt được lưu trữ trong tập tin "/etc/ipsec.secrets". Với lý do này chúng tôi muốn thông báo rằng bạn nên sử dụng sự kết nối Automatically keyed sẽ có nhiều tính năng bảo mật hơn so với việc sử dụng Manually keyed. Và đây cũng là phương cách được sử dụng nhiều trong việc thiết lập VPN.

Trong ví dụ về cấu hình dưới đây, chúng ta sẽ cấu hình một đường hầm (tunnel) xuyên qua hai firewall được thiết lập giữa hai đầu bên trái và bên phải. Tôi chọn trình bày cấu hình này cho các bạn vì tôi cho rằng đây là cấu hình mà nhiều người và các công ty sẽ sử dụng, nó cũng cho phép chúng ta có nhiều tùy chọn trong tập tin cấu hình "ipsec.conf" cho các kết nối automatically keyed. Các ví dụ cấu hình khác nhau cũng có sẵn và bạn cũng có thể tham khảo trong thư mục "doc/examples" trong thư mục nguồn Frees/WAN.

SubnetDeep=====Deep·····Deepgate.....Mailgate·····Mail=====SubnetMail

Mạng không an toàn (Internet)

leftsubnet = **SubnetDeep** (192.168.1.0/24)

left = **Deep** (deep.openna.com)

leftnexthop = **Deepgate** (the first router in the direction or ISP router for deep.openna.com)

Internet = **Untrusted net**

rightnexthop = **Mailgate** (the first router in the direction or ISP router for mail.openna.com)

right = **Mail** (mail.openna.com)

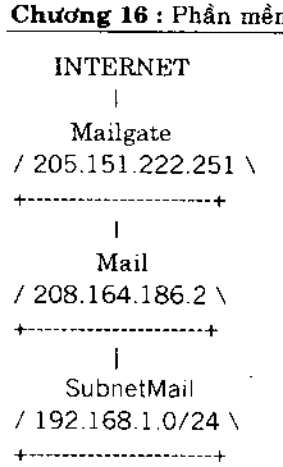
rightsubnet = **SubnetMail** (192.168.1.0/24)

```
  SubnetDeep
\ 192.168.1.0/24 /
+------------------+
        |
      Deep
\ 208.164.186.1 /
        |
    Deepgate
\ 205.151.222.250 /
+-----------------+
        |
```

SubnetDeep là địa mạng IP của mạng riêng của bạn trên gateway thứ nhất, eth1 được gắn bên trong network.

Deep là địa chỉ IP của gateway thứ nhất, eth0 được gắn vào Internet.

Deepgate là địa chỉ IP của Router thứ nhất theo hướng của gatewate thứ hai của bạn (mail.openna.com) hoặc router ISP của bạn.



Internet là mạng không an toàn.

Mailgate là địa chỉ IP của Router thứ hai theo hướng của gatewate thứ nhất của bạn (deep.openna.com) hoặc router ISP của bạn.

Mail là địa chỉ IP của gateway thứ hai của bạn, eth0 được gắn vào Internet.

SubnetMail là địa chỉ mạng IP mạng riêng của bạn trên gateway thứ hai, eth1 được gắn bên trong mạng.

Chúng ta phải hiệu chỉnh tập tin ipsec.conf (vi /etc/ipsec.conf) và thay đổi những giá trị mặt định để chỉnh sửa sự chỉ định của chúng ta cho cấu hình IPSec và sự trao đổi thông tin. Hiện tại chúng ta có kiểu của section trong tập tin (/etc/ipsec.conf): phần "config" chỉ rõ thông tin cấu hình chung cho IPSec và phần "conn" chỉ rõ sự kết nối IPSec. Nội dung của chúng phải tuyệt đối bảo mật trừ khi manual keying được dùng (nhắc lại, manual keying không nên dùng cho thông tin cần bảo mật)

Section đầu tiên có tên *config setup* chứa đựng toàn bộ những tham số cài đặt cho IPSEC mà chúng sẽ được áp dụng cho tất cả các connection và thông tin được dùng khi software khởi động.

Section thứ hai có tên conn chứa đựng một sự chỉ định kết nối mạng dùng IPSec. Tên chúng ta đặt cho nó là tùy ý.

Xem ví dụ sau:

```
# /etc/ipsec.conf - FreeS/WAN IPSEC configuration file

# More elaborate and more varied sample configurations can be found
# in doc/examples.

# basic configuration
config setup
        interfaces="ipsec0=eth0"
        klipsdebug=none
        plutodebug=none
        plutoload=%search
        plutostart=%search

# sample connection
conn deep-mail
```

```
left=208.164.186.1
leftsubnet=192.168.1.0/24
leftnexthop=205.151.222.250
right=208.164.186.2
rightsubnet=192.168.1.0/24
rightnexthop=205.151.222.251
keyingtries=0
auth=ah
auto=start
```

Tập tin ipsec.conf thiết đặt cấu hình với những thông số như sau:

interfaces="ipsec0=eth0"

Tùy chọn này chỉ rõ interface vật lý và ảo nào dành riêng cho IPSec sử dụng. Sự thiết đặt mặc định "interfaces=%defaultroute" sẽ tìm sự kết nối mặc định của bạn tới internet hoặc kết nối trong mạng cho ipsec. Bạn có thể chỉ định một hoặc nhiều interface được dùng cho Freeswan ví dụ như:

```
interfaces="ipsec0=eth0"
interfaces="ipsec0=eth0 ipsec1=ppp0"
```

Cả hai ví dụ trên thiết đặt interface eth0 như ipsec0, trong ví dụ thứ hai cho biết IPSec cũng được hỗ trợ trên PPP. Nếu việc cài đặt mặt định "interfaces=%defaultroue" không được dùng thì chỉ những interface được chỉ định trên một gateway có thể sử dụng để trao đổi thông tin với các gateway khác.

klipsdebug=none

Tùy chọn này chỉ rõ đưa ra màn hình thông tin giúp ta gỡ rối chương trình KLIPS. Giá trị mặc định none, ý nghĩa rằng không có đưa ra màn hình thông tin gỡ rối và giá trị all ý nghĩa đưa ra đầy đủ.

plutodebug=none

Tùy chọn này chỉ rõ đưa ra màn hình thông tin giúp ta gỡ rối chương trình Pluto key. Giá trị mặc định none, ý nghĩa rằng không có đưa ra màn hình thông tin gỡ rối và giá trị all ý nghĩa đưa ra đầy đủ.

plutoload=%search

Tùy chọn này chỉ rõ connection nào (bằng tên) được tải một cách tự động vào trong bộ nhớ khi Pluto khởi động. Mặc định là none và giá trị %search tải tất cả các connection với auto=add hoặc auto=start.

plutostart=%search

Tùy chọn này chỉ rõ connection nào (bằng tên) được dàn sếp một cách tự động khi Pluto khởi động. Mặc định là none và giá trị %search khởi động tất cả các connection với auto=start.

Conn deep-mail

Tùy chọn này chỉ rõ tên đưa ra để nhận biết connection nào dùng IPSec. Bạn nên có một qui ước trong việc đánh tên để tránh sự nhầm lẫn. Ví dụ liên kết giữa hai gateway server deep.openna.com và mail.openna.com có thể được đặt tên là deep-mail hoặc liên kết giữa các văn phòng ở Saigon và Hanoi có thể đặt tên là saigon-hanoi.

Chú ý: *Tên deep-mail hoặc bất cứ tên nào bạn chọn nên giống nhau trong tập tin ipsec.conf trên cả hai gateway, nếu có thay đổi thì nên làm trong tập tin /etc/ipsec.conf trên gateway thứ hai là thay đổi dòng "interface=" cho phù hợp với interface mà gateway thứ hai sử dụng cho sự kết nối IPSEC. Tất nhiên bạn sẽ không cần làm gì hết nếu không có sự khác biệt nào. Ví dụ nếu interface eth0 được sử dụng trên cả hai gateway cho sự trao đổi thông tin IPSEC bạn không cần thay đổi dòng interface= trên gateway thứ hai. Trong trường hợp nếu gateway thứ nhất dùng eth0 và gateway thứ hai sử dụng eth1 bạn phải thay đổi dòng "interface=" trên gateway thứ hai cho phù hợp với interface eth1.*

left=208.164.186.1

Tùy chọn này chỉ rõ địa chỉ IP của interface bên ngoài của gateway được dùng để nói chuyện với các gateway khác.

leftsubnet=192.168.1.0/24

Tùy chọn này chỉ rõ địa chỉ mạng hoặc địa chỉ của mạng riêng phía sau gateway.

leftnexthop=205.151.222.250

Tùy chọn này chỉ rõ địa chỉ IP của router đầu tiên hoặc IP của router của ISP.

right=208.164.186.2

Tùy chọn này chỉ rõ địa chỉ IP của interface bên ngoài của gateway bên phải được dùng để nói chuyện với các gateway khác.

rightsubnet=192.168.1.0/24

Tùy chọn này chỉ rõ địa chỉ mạng hoặc địa chỉ của mạng riêng phía sau gateway.

rightnexthop=205.151.222.251

Tùy chọn này chỉ rõ địa chỉ IP của router đầu tiên hoặc IP của router của ISP.

keyingtries=0

Tùy chọn này chỉ rõ bao nhiêu lần hai gateway sẽ cố gắng để đưa ra một khóa cho ipsec. Giá trị mặc định 0 là một gợi ý (sẽ retry mãi mãi).

auth=ah

Tùy chọn này chỉ rõ việc kiểm tra tính xác thực (authentication) sẽ được dùng là AH (Authenticatication Header), ngoài ra chúng ta có thể dùng ESP (Encapsulated Security Payload).

auto=start

Tùy chọn này chỉ rõ có hay không quá trình khởi động (startup) IPSec sẽ được làm một cách tự động.

Lưu ý: *Một sự nhầm lẫn ở bất kỳ nơi nào trong tập tin cấu hình ipsec.conf sẽ là nguyên nhân dẫn đến việc thông báo các lỗi khác nhau trong log file.*

Cấu hình tập tin "/etc/ipsec.secrets"

File ipsec.secrets lưu trữ khóa bảo mật được sử dụng bởi chương trình pluto để kiểm tra tính xác thực giữa hai gateway. Có hai loại khóa bảo mật có thể được configure trong tập tin này là preshared secrets key và RSA private keys. Bạn phải kiểm tra mode và permission của tập tin này để đảm bảo rằng super-user "root" là sở hữu của tập tin này và permission trên tập tin này thì bị khóa không cho truy cập bởi những user khác.

Bước 1:

Mặc định một ví dụ về secret key được cung cấp trong tập tin ipsec.secrets, bạn nên thay đổi nó bằng cách tạo ra secret key cho chính bạn. Ipsec có thể support cho bạn tạo ra khóa 256bit và dùng khóa này trong suốt quá trình trao đổi khóa để đảm bảo rằng các tin tặc không thể tìm thấy trong thời gian ngắn.

- Để tạo một shared secret key mới dùng những lệnh sau:

```
[root@deep /]# ipsec ranbits 256 > temp
```

Khóa mới được tạo với trình tiện ích ranbits(8) và được lưu trong tập tin temp. Khóa mới được tạo sau một vài giây và nó được chọn ngẫu nhiên 256bit.

Chú ý: *Đừng quên xóa tập tin temp sau khi bạn đã làm xong với nó.*

Bước 2:

Bây giờ bạn đã có khóa shared secret mới được lưu trong tập tin temp, bạn phải đặt nó trong tập tin "/etc/ipsec.secrets" Khi hiệu chỉnh tập tin ipsec.secrets bạn sẽ thấy một vài điều giống như xuất hiện dưới đây trên màn hình của bạn. Mỗi dòng có địa chỉ IP của cả hai gateway cộng với khóa bảo mật. Nó trông giống như trình bày dưới đây:

```
# This file holds shared secrets which are currently the only inter-Pluto
# authentication mechanism. See ipsec_pluto(8) manpage. Each secret is
# (oversimplifying slightly) for one pair of negotiating hosts.
# The shared secrets are arbitrary character strings and should be both
```

long and hard to guess.

Note that all secrets must now be enclosed in quotes, even if they have
no white space inside them.

10.0.0.1 11.0.0.1 "jxVS1kVUTTulkVRRTnTujSm444jRuU1mlkkIku2nkW3nnVu V2WjjRRnuImlkmU1Run5VSnnRT"

Hiệu chỉnh tập tin ipsec.secrets (vi /etc/ipsec.secrets) và thay đổi khóa

10.0.0.1 11.0.0.1 " jxVS1kVUTTulkVRRTnTujSm444jRuU1mlkkIku2nkW3nnVu V2WjjRRnuImlkmU1Run5VSnnRT "

Trở thành:

208.164.186.1 208.164.186.2

"0x9748cc31_2e99194f_d230589b_cd846b57_dc070b01_74b66f34_19c40a1a_804906ed"

Ở đây "208.164.186.1" và "208.164.186.2" là địa chỉ IP của hai gateway và "0x9748cc31_2e99194f_d230589b_cd846b57_dc070b01_74b66f34_19c40a1a_804906ed" (Chú ý rằng yêu cầu có dấu ngoặc kép) là khóa bảo mật mà chúng ta đã tạo ra ở trên với lệnh "ipsec ranbits 256 > temp" trong tập tin temp

Bước 3:

Cả hai tập tin "ipsec.conf" và "ipsec.secrets" phải được copy tới máy gateway thứ hai để đảm bảo tính giống hệt trên cả hai, chỉ ngoại trừ tập tin ipsec.conf ở dòng config setup cần chỉnh sửa chính xác interface cài đặt cho gateway thứ hai nếu chúng khác so với gateway thứ nhất. Khác với secret key, RSA private key có cùng shared key trên cả hai gateway.

Chú ý: *File "/etc/ipsec.secrets" sẽ có permission rw-------(600) và được sở hữu bởi super-user root. tập tin "/etc/ipsec.conf được cài đặt với quyền rw-r--r--(644) và cũng được sở hữu bởi root.*

Cấu hình RSA private keys secrets

Xin nhắc lại hiện tại FreesWAN hỗ trợ hai loại khóa là: preshares secrets key và RSA key. Về preshared secrets là những gì chúng ta đã cấu hình (configure) trong các tập tin ipsec.conf và ipsec.secrets ở trên. Một vài người thích dùng RSA private key cho việc kiểm tra tính xác thực bằng chương trình Pluto daemon của những host khác. Nếu bạn ở trong trường hợp này bạn sẽ phải làm một vài sự thay đổi nhỏ trên cả hai tập tin ipsec.conf và ipsec.secrets như được diễn giải trong những bước dưới đây:

Bạn cần tạo một RSA riêng lẻ cho mỗi gateway, mỗi gateway đặt private key của nó trong chính tập tin ipsec.secrets của gateway đó và public key sẽ được

đặt trong các tham số "leftrsasigkey" và "rightrsasigkey" trong phần conn của tập tin "ipsec.conf". Các bước tiến hành như sau:

Bước 1:

Tạo một khóa RSA riêng lẻ cho mỗi gateway:

- Trên gateway thứ nhất (ví dụ deep) sử dụng những lệnh sau:

```
[root@deep /]# cd /
[root@deep /]# ipsec rsasigkey --verbose 1024 > deep-keys
computing primes and modulus...
getting 64 random bytes from /dev/random
looking for a prime starting there
found it after 30 tries
getting 64 random bytes from /dev/random
looking for a prime starting there
found it after 230 tries
swapping primes so p is the larger
computing (p-1)*(q-1)...
computing d...
computing exp1, exp1, coeff...
output...
```

- Trên gateway thứ hai (ví dụ mail) dùng những lệnh sau:

```
[root@mail /]# cd /
[root@mail /]# ipsec rsasigkey --verbose 1024 > mail-keys
computing primes and modulus...
getting 64 random bytes from /dev/random
looking for a prime starting there
found it after 30 tries
getting 64 random bytes from /dev/random
looking for a prime starting there
found it after 230 tries
swapping primes so p is the larger
compu ting (p-1)*(q-1)...
computing d...
computing exp1, exp1, coeff...
output...
```

Trình tiện ích rsasigkey sinh ra một cặp khóa private và public key với một chữ ký 1024 bit và đặt nó trong tập tin deep-keys (mail-keys cho lệnh thứ hai trên gateway thứ hai). Private key có thể được đưa vào đúng nguyên văn trong tập tin "ipsec.secrets" và public key được đưa vào trong tập tin "ipsec.conf".

Chú ý: *Đừng quên xóa các tập tin tạm sau khi đã làm việc với nó.*

Bước 2:

Thay đổi tập tin /etc/ipsec.conf của bạn để dùng RSA public keys trong mỗi gateway:

Hiệu chỉnh tập tin ipsec.conf nguyên thủy của bạn (vi /etc/ipsec.conf) và thêm vào những tham số liên quan đến RSA trong phần conn của tập tin ipsec.conf trên cả hai gateway:

```
# sample connection
conn deep-mail
left=208.164.186.1
leftsubnet=192.168.1.0/24
leftnexthop=205.151.222.250
right=208.164.186.2
rightsubnet=192.168.1.0/24
rightnexthop=205.151.222.251
keyingtries=0
auth=ah
authby=rsasig
leftrsasigkey=<Public key of deep>
rightrsasigkey=<Public key of mail>
auto=start
```

authby=rsasig

Tham số này chỉ rõ làm thế nào hai gateway bảo mật sẽ kiểm tra tính xác thực với nhau. Giá trị mặc định là secret cho shared secrets. Bạn phải chỉ rõ rsasig cho RSA sau khi chúng ta đã quyết định dùng chữ ký điện tử RSA.

leftrsasigkey=<Public key of deep>

Tham số này chỉ ra public key của gateway hay người tham dự bên trái cho việc kiểm tra tính xác thực RSA. Trong ví dụ của chúng ta left là 208.164.186.1 và đại diện cho deep.openna.com như vậy chúng ta phải đặt RSA public key cho deep trên dòng này.

rightrsasigkey=<Public key of mail>

Tham số này chỉ ra public key của gateway hay người tham dự bên phải cho việc kiểm tra tính xác thực RSA. Trong ví dụ của chúng ta right là 208.164.186.2 và đại diện cho mail.openna.com như vậy chúng ta phải đặt RSA public key cho mail trên dòng này.

Chúng ta có thể nhận được public key của deep trong tập tin RSA key tên "deep-keys" và public key của mail cho tập tin RAS key tên "mail-keys" chúng ta đã tạo trong bước 1 ở trên. Các tập tin này trông giống như sau:

RAS keys cho gateway deep (deep-keys)

[root@deep /]# **cd /**

[root@deep /]# **vi deep-keys**

1024 bits, Fri Feb 4 05:05:19 2000

for signatures only, UNSAFE FOR ENCRYPTION

#pubkey=**0x010395daee1be05f3038ae529ef2668afd79f5ff1b16203c9ceaef801cea9cb74bcfb51a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b5427e04cf4a38ef98a7f29edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f084186a2a54f4b4cec12cca1a5deac3b19d561c16a76bab772888f1fd71aa08f08502a141b611f**

Modulus:

0x95daee1be05f3038ae529ef2668afd79f5ff1b16203c9ceaef801cea9cb74bcfb51a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b5427e04cf4a38ef98a7f29edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f084186a2a54f4b4cec12cca1a5deac3b19d561c16a76bab772888f1fd71aa08f08502a141b611f

PublicExponent: 0x03

everything after this point is secret

PrivateExponent:

0x63e74967eaea2025c98c69f6ef0753a6a3ff6764157dbdf1f50013471324dd352366f48805b0b37f232384b2b52ce2ee85d173468b62eaa052381a9588a317b3a1324d01a531a41fa7add6c5efbdd88f4718feed2bc0246be924e81bb90f03e49ceedf7af0dd48f06f265b519600bd082c6e6bd27eaa71cc0288df1ecc3b062b

Prime1:

0xc5b471a88b025dd09d4bd7b61840f20d182d9b75bb7c11eb4bd78312209e3aee7ebfe632304db6df5e211d21af7fee79c5d45546bea3ccc7b744254f6f0b847f

Prime2:

0xc20a99feeafe79767122409b693be75f15e1aef76d098ab12579624aec708e85e2c5dd62080c3a64363f2f45b0e96cb4aef8918ca333a326d3f6dc2c72b75361

Exponent1:

0x83cda11b0756e935be328fcebad5f6b36573bcf927a80bf2328facb6c0697c9eff2a9976cade79ea3ec0be1674fff4512e8d8e2f29c2888524d818df9f5d02ff

Exponent2:

0x815c66a9f1fefba44b6c2b124627ef94b9411f4f9e065c7618fb96dc9da05f03ec83e8ec055d7c42ced4ca2e75f0f3231f5061086ccd176f37f9e81da1cf8ceb

Coefficient:

0x10d954c9e2b8d11f4db1b233ef37ff0a3cecfffad89ba5d515449b007803f577e3bd7f0183ceddfd805466d62f767f3f5a5731a73875d30186520f1753a7e325

RSA keys cho gateway mail (mail-keys):

[root@mail /]# **cd /**

[root@mail /]# **vi mail-keys**

```
        # 1024 bits, Fri Feb 4 04:46:59 2000
        # for signatures only, UNSAFE FOR ENCRYPTION
        #pubkey=0x01037631b81f00d5e6f888c542d44dbb784cd3646f084ed96
f942d341c7c4686cbd405b805dc728f8697475f11e8b1dd797550153a3f0d4ff0f2
b274b70a2ebc88f073748d1c1c8821dc6be6a2f0064f3be7f8e4549f8ab9af64944
f829b014788dd202cf7d2e320cab666f5e7a197e64efe0bfee94e92ce4dad82d52
30c57b89edf
        Modulus:
0x7631b81f00d5e6f888c542d44dbb784cd3646f084ed96f942d341c7c4686cbd4
05b805dc728f8697475f11e8b1dd797550153a3f0d4ff0f2b274b70a2ebc88f0737
48d1c1c8821dc6be6a2f0064f3be7f8e4549f8ab9af64944f829b014788dd202cf7
d2e320cab666f5e7a197e64efe0bfee94e92ce4dad82d5230c57b89edf
        PublicExponent: 0x03
        # everything after this point is secret
        PrivateExponent:
0x4ecbd014ab3944a5b08381e2de7cfadde242f4b03490f50d737812fd8459dd38
03d003e84c5faf0f84ea0bf07693a64e35637c2a08dff5f721a324b1747db09f62c8
71d5e11711251b845ae76753d4ef967c494b0def4f5d0762f65da603bc04c41b4c
6cab4c413a72c633b608267ae2889c162a3d5bc07ee083b1c6e038400b
        Prime1:
0xc7f7cc8feaaac65039c39333b878bffd8f95b0dc22995c553402a5b287f341012
253e9f25b83983c936f6ca512926bebee3d5403bf9f4557206c6bbfd9aac899
        Prime2:
0x975015cb603ac1d488dc876132d8bc83079435d2d3395c03d5386b5c004ead
d4d7b01b3d86aad0a2275d2d6b791a2abe50d7740b7725679811a32ca22db976
37
        Exponent1:
0x854fddb5471c84357bd7b777d0507ffe5fb92092c1bb92e37801c3cc5aa22b56
16e29bf6e7ad1028624a486e0c619d47f428e2ad2a6a2e3a159d9d2a911c85bb
        Exponent2:
0x64e00e87957c81385b3daf9621e5d302050d7937377b92ad38d04792aadf1e8
de52012290471e06c1a3e1e47a61171d435e4f807a4c39a6561177316c9264ecf
        Coefficient:
0x6f087591becddc210c2ee0480e30beeb25615a3615203cd3cef65e5a1d476fd9
602ca0ef10d9b858edb22db42c975fb71883a470b43433a7be57df7ace4a0a3f
```

Định rõ và copy các public key của deep và mail trên cả hai tập tin deep-keys và mail-keys vào tập tin ipsec.conf như được trình bày ở dưới đây. Bạn có thể xác định dòng liên quan tới public key bởi một câu bắt đầu với dòng "#pubkey="

```
# sample connection
conn deep-mail
        left=208.164.186.1
        leftsubnet=192.168.1.0/24
```

leftnexthop=205.151.222.250
right=208.164.186.2
rightsubnet=192.168.1.0/24
rightnexthop=205.151.222.251
keyingtries=0
auth=ah
authby=rsasig

leftrsasigkey=0x010395daee1be05f3038ae529ef2668afd79f5ff1b16203c9ceaef801cea9cb74bcfb51a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b5427e04cf4a38ef98a7f29edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f084186a2a54f4b4cec12cca1a5deac3b19d561c16a76bab772888f1fd71aa08f08502a141b611f

rightrsasigkey=0x01037631b81f00d5e6f888c542d44dbb784cd3646f084ed96f942d341c7c4686cbd405b805dc728f8697475f11e8b1dd797550153a3f0d4ff0f2b274b70a2ebc88f073748d1c1c8821dc6be6a2f0064f3be7f8e4549f8ab9af64944f829b014788dd202cf7d2e320cab666f5e7a197e64efe0bfee94e92ce4dad82d5230c57b89edf

auto=start

Chú ý: *Xin nhắc lại rằng trong ví dụ trên tham số "leftrsasigkey=" chứa đựng public key của deep và "rightrsasigkey=" chứa đựng public key của mail.*

Bước 3:

Thay đổi tập tin "/etc/ipsec.secrets" để dùng private key trong mỗi gateway:

Hiệu chỉnh tập tin ipsec.secrets ban đầu (vi /etc/ipsec.secrets) và thêm RSA private key cho việc kiểm tra tính xác thực trên cả hai gateway

File ipsec.secrets cho gateway deep:

[root@deep /]# **vi /etc/ipsec.secrets**

208.164.186.1 208.164.186.2

"0x9748cc31_2e99194f_d230589b_cd846b57_dc070b01_74b66f34_19c40a1a_804906ed"

Bạn phải thay đổi tập tin ipsec.secrets của bạn trông giống như được trình bày ở trên cả trên hai gateway. Có một điều quan trọng là private key thì không giống nhau trên cả hai gateway deep và mail. Private key cho deep được lấy từ tập tin RSA key "deep-keys" trong khi private key cho mail được lấy từ tập tin RSA key "mail-keys"

208.164.186.1 208.164.186.2: RSA {

Modulus:

0x95daee1be05f3038ae529ef2668afd79f5ff1b16203c9ceaef801cea9cb74bcfb51a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b5427e04cf4a38ef98a7f2

9edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f084186a2a54f4b4cec12cca
1a5deac3b19d561c16a76bab772888f1fd71aa08f08502a141b611f

PublicExponent: 0x03

everything after this point is secret

PrivateExponent:

0x63e74967eaea2025c98c69f6ef0753a6a3ff6764157dbdf1f50013471324dd352
366f48805b0b37f232384b2b52ce2ee85d173468b62eaa052381a9588a317b3a1
324d01a531a41fa7add6c5efbdd88f4718feed2bc0246be924e81bb90f03e49ceedf
7af0dd48f06f265b519600bd082c6e6bd27eaa71cc0288df1ecc3b062b

Prime1:

0xc5b471a88b025dd09d4bd7b61840f20d182d9b75bb7c11eb4bd78312209e3a
ee7ebfe632304db6df5e211d21af7fee79c5d45546bea3ccc7b744254f6f0b847f

Prime2:

0xc20a99feeafe79767122409b693be75f15e1aef76d098ab12579624aec708e85e
2c5dd62080c3a64363f2f45b0e96cb4aef8918ca333a326d3f6dc2c72b75361

Exponent1:

0x83cda11b0756e935be328fcebad5f6b36573bcf927a80bf2328facb6c0697c9eff
2a9976cade79ea3ec0be1674fff4512e8d8e2f29c2888524d818df9f5d02ff

Exponent2:

0x815c66a9f1fefba44b6c2b124627ef94b9411f4f9e065c7618fb96dc9da05f03ec
83e8ec055d7c42ced4ca2e75f0f3231f5061086ccd176f37f9e81da1cf8ceb

Coefficient:

0x10d954c9e2b8d11f4db1b233ef37ff0a3cecfffad89ba5d515449b007803f577e3
bd7f0183ceddfd805466d62f767f3f5a5731a73875d30186520f1753a7e325

}

File "ipsec.secrets" cho gateway mail

[root@mail /]# **vi /etc/ipsec.secrets**

208.164.186.1 208.164.186.2: RSA {

Modulus:

0x95daee1be05f3038ae529ef2668afd79f5ff1b16203c9ceaef801cea9cb74bcfb51
a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b5427e04cf4a38ef98a7f2
9edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f084186a2a54f4b4cec12cca
1a5deac3b19d561c16a76bab772888f1fd71aa08f08502a141b611f

PublicExponent: 0x03

everything after this point is secret

PrivateExponent:

0x63e74967eaea2025c98c69f6ef0753a6a3ff6764157dbdf1f50013471324dd352
366f48805b0b37f232384b2b52ce2ee85d173468b62eaa052381a9588a317b3a1
324d01a531a41fa7add6c5efbdd88f4718feed2bc0246be924e81bb90f03e49ceedf
7af0dd48f06f265b519600bd082c6e6bd27eaa71cc0288df1ecc3b062b

Prime1:

0xc5b471a88b025dd09d4bd7b61840f20d182d9b75bb7c11eb4bd78312209e3a
ee7ebfe632304db6df5e211d21af7fee79c5d45546bea3ccc7b744254f6f0b847f

Prime2:

0xc20a99feeafe79767122409b693be75f15e1aef76d098ab12579624aec708e85e
2c5dd62080c3a64363f2f45b0e96cb4aef8918ca333a326d3f6dc2c72b75361

Exponent1:

0x83cda11b0756e935be328fcebad5f6b36573bcf927a80bf2328facb6c0697c9eff
2a9976cade79ea3ec0be1674fff4512e8d8e2f29c2888524d818df9f5d02ff

Exponent2:

0x815c66a9f1fefba44b6c2b124627ef94b9411f4f9e065c7618fb96dc9da05f03ec
83e8ec055d7c42ced4ca2e75f0f3231f5061086ccd176f37f9e81da1cf8ceb

Coefficient:

0x10d954c9e2b8d11f4dbfb233ef37ff0a3cecfffad89ba5d515449b007803f577e3
bd7f0183ceddfd805466d62f767f3f5a5731a73875d30186520f1753a7e325

}

Kiểm tra tính xác thực bằng chữ ký RSA yêu cầu mỗi host có private key của chính nó. Phần khóa có thể bắt đầu với một dấu hiệu chỉ rõ loại của khóa. "RAS" biểu hiện cho RSA private key và "PSK" (mặc định) biểu hiện cho PreShared Key. Vì "PSK" được dùng mặc định bạn phải chỉ rõ "RSA". Như vậy chúng ta có thể sử dụng RSA private key trong tập tin này (ipsec.secrets). Super-user "root" là người chủ sở hữu tập tin này và permission sẽ bị khóa cho việc truy cập của các user khác.

Yêu cầu cài đặt Network cho IPSec

Có một vài sự nghiên cứu bạn phải bảo đảm là chính xác trước khi chạy phần mềm FreeS/WAN. Những nghiên cứu này rất quan trọng nếu bạn không muốn nhận messages lỗi trong quá trình startup VPN của bạn. Theo sau là những bước cần tiến hành.

Bước 1

Bạn sẽ cần làm cho có tác dụng chức năng TCP/IP forwarding trên cả hai gateway server. Trong Red Hat Linux, việc này thực hiện bằng cách thay đổi hoặc thêm những dòng sau, phụ thuộc trên version Red Hat bạn đang sử dụng:

Dưới Red Hat Linux 6.1

- Hiệu chỉnh tập tin network (vi /etc/sysconfig/network) và thay đổi dòng sau:

  FORWARD_IPV4="false"

  Trở thành:

  FORWARD_IPV4="yes"

- Bạn phải khởi động lại network của bạn để việc thay đổi có hiệu lực:

  [root@deep /]# /etc/rc.d/init.d/network restart

  Bringing up interface lo [OK]

```
Bringing up interface eth0 [ OK ]
Bringing up interface eth1 [ OK ]
```

Dưới Red Hat Linux 6.2

- Làm cho có hiệu lực IPv4 forworking trên hệ thống RH6.2, dùng lệnh sau:

Hiệu chỉnh tập tin /etc/sysctl.conf và thêmvào những dòng sau:

```
# Enable packet forwarding
net.ipv4.ip_forward = 1
```

Bạn phải khởi động lại network của bạn để việc thay đổi có hiệu lực. Lệnh để restart network như sau:

- Khởi động lại tất cả các thiết bị bằng tay trên hệ thống của bạn dùng những lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters [ OK ]
Bringing up interface lo [ OK ]
Bringing up interface eth0 [ OK ]
Bringing up interface eth1 [ OK ]
```

Bước 2:

Nhắc lại rằng các kết nối automatically keyed sử dụng các khóa được sinh ra một cách tự động bởi chương trình Pluto key negotiation daemon. Chương trình pluto daemon khi khởi động sẽ thử connect tới chương trình Pluto daemon khác ở cuối tunnel và thiết lập một connection. Do nguyên nhân này một gateway IPSEC nên có các qui tắc lọc package (trong tập tin script firewall) cho phép những protocol sau đi qua gateway khi nói chuyện với IPSEC gateway khác.

- UDP port 500 cho IKE được thực thi bởi Pluto daemon.
- Protocol 50 cho ESP mã hoá và kiểm tra tính xác thực.
- Protocol 51 cho việc kiểm tra tính xác thực mức package.

Hiệu chỉnh tập tin script firewall (vi /etc/rc.d/init.d/firewall) trên cả hai máy gateway và thêm vào những dòng sau cho phép các package IPSEC đi qua gateway từ xa vào gateway của bạn.

```
# FreeS/WAN IPSec VPN
# .....

# If you are using the FreeSWAN IPSec VPN, you will need to fill in the
# addresses of the gateways in the IPSECSG and the virtual interfaces for
# FreeS/Wan IPSEC in the FREESWANVI parameters. Look at the beginning of
# this firewall script rules taäp tin to set the parameters.
```

```
# IPSECSG is a Space separated list of remote gateways. FREESWANVI is a
# Space separated list of virtual interfaces for FreeS/Wan IPSEC
# implementation. Only include those that are actually used.

# Allow IPSEC protocol from remote gateways on external interface
# IPSEC uses three main types of packet:
# IKE uses the UDP protocol and port 500,
# ESP use the protocol number 50, and
# AH use the protocol number 51

ipchains -A input -i $EXTERNAL_INTERFACE -p udp \
-s $IPSECSG -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p udp \
-d $IPSECSG -j ACCEPT

ipchains -A input -i $EXTERNAL_INTERFACE -p 50 \
-s $IPSECSG -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p 50 \
-d $IPSECSG -j ACCEPT

ipchains -A input -i $EXTERNAL_INTERFACE -p 51 \
-s $IPSECSG -j ACCEPT

ipchains -A output -i $EXTERNAL_INTERFACE -p 51 \
-d $IPSECSG -j ACCEPT

# Allow all traffic to FreeS/WAN Virtual Interface
ipchains -A input -i $FREESWANVI \
        -s $ANYWHERE \
        -d $ANYWHERE -j ACCEPT

ipchains -A output -i $FREESWANVI \
        -s $ANYWHERE \
        -d $ANYWHERE -j ACCEPT

# Forward anything from the FreeS/WAN virtual interface IPSEC tunnel
ipchains -A forward -i $FREESWANVI \
```

```
-s $ANYWHERE \
-d $ANYWHERE -j ACCEPT
```

```
Ở đây EXTERNAL_INTERFACE="eth0"      # Interface kết nối với bên ngoài
Ở đây ANYWHERE="any/0"               # Ý nghĩa là 0.0.0.0/0
Ở đây IPSECSG="208.164.186.2"        # Địa chỉ IP của VPN gateway từ xa
Ở đây FREESWANVI="ipsec0"            # Virtual interface cho FresS/Wan
```

Chú ý: *Xem chương Netwoking firewall cho nhiều thông tin. Đừng quên thêm hoặc kiểm tra những qui tắc firewall này trong những gateway khác.*

Bước 3:

rp_filer (liên quan đến sự bảo vệ IP) phải được tắt trên cả hai gateway để IPSEC làm việc chính xác. Việc này được thực hiện bằng cách kiểm tra nếu giá trị 0 (off) được set trong các tập tin tương ứng

"/proc/sys/net/ipv4/conf/ipsec0/rp_filter và

"/proc/sys/net/ipv4/conf/eth0/rp_filter"

- Kiểm tra nếu giá trị 0 (off) được set trong các tập tin rp_filer dùng lệnh:

```
[root@deep /]# cat /proc/sys/net/ipv4/conf/ipsec0/rp_filter
0
[root@deep /]# cat /proc/sys/net/ipv4/conf/eth0/rp_filter
0
```

Chú ý: *Thư mục con "ipec0" trong ví dụ của chúng ta sẽ được tạo chỉ sau khi khởi động lại máy tính. Như vậy bạn có thể kiểm tra giá trị của tập tin "rp_filer" trong thư mục ipsec0 sau khi hệ thống bạn được khởi động lại.*

- Thiết lập giá trị 0 (off) trong cả hai tập tin rp_filter bằng tay dùng lệnh:

```
[root@deep /]# echo 0 > /proc/sys/net/ipv4/conf/ipsec0/rp_filter
[root@deep /]# echo 0 > /proc/sys/net/ipv4/conf/eth0/rp_filter
```

Bạn cũng có thể đặt những dòng như dưới đây trong các tập tin script fileware (/etc/rc.d/init.d/firewall) trên cả hai gateway để tự động thiết đặt những giá trị này tới 0 (off) và khỏi phải làm việc này bằng tay.

```
# Disable IP spoofing protection to allow IPSEC to work properly
echo 0 > /proc/sys/net/ipv4/conf/ipsec0/rp_filter
echo 0 > /proc/sys/net/ipv4/conf/eth0/rp_filter
```

Chú ý: *Trong ví dụ của tập tin script firewall trên, chúng ta giả sử rằng eth0 là interface bạn sử dụng cho sự kết nối của bạn. Dĩ nhiên nếu bạn dùng eth1 bạn phải thay đổi eth0 thành eth1.*

Nếu bạn quên bước này bạn sẽ nhận message lỗi trên màn hình của bạn như được trình bày dưới đây trong quá trình khởi động FreeSWAN IPSEC.

```
ipsec_setup: WARNING: ipsec0 has route filtering turned on, KLIPS may not work
ipsec_setup: (/proc/sys/net/ipv4/conf/ipsec0/rp_filter = `1', should be 0)
ipsec_setup: WARNING: eth0 has route filtering turned on, KLIPS may not work
ipsec_setup: (/proc/sys/net/ipv4/conf/eth0/rp_filter = `1', should be 0)
```

Bước 4:

Điều quan trọng cần thông báo rằng bất kỳ những qui tắc giả dạng (masquerading) cho các client bên trong network IPSEC phải đặt sau những qui tắc cho phép lưu thông liên quan tới IPSEC (bước 2 và 3 ở trên) hoặc máy sẽ thử masquerade những package thay vì chúng được chuyển đi trên IPSEC.

Hiệu chỉnh tập tin scriptfirewall (vi /etc/rc.d/init.d/firewall) trên cả hai gateway và thêm những dòng sau cho phép masquerade những package đi qua gateway mạng từ xa tới tới gateway mạng của bạn:

```
# Masquerade internal traffic.
# All internal traffic is masqueraded externally.
ipchains -A forward -i $EXTERNAL_INTERFACE -s $LOCALNET_1 -j MASQ
```

Ở đây EXTERNAL_INTERFACE="eth0" # Interface bên ngoài nối với Internet.

Ở đây LOCALNET_1=" 192.168.1.0/24" # bất cứ khoảng riêng lẻ nào bạn sử dụng.

Xem chương "Liên kết firewall với hỗ trợ Masquerading và Forwarding" cho nhiều thông tin.

Bây giờ bạn có thể khởi động lại hệ thống của bạn và những máy trên gateway A có thể nói chuyện với những máy trên gateway B và sẽ không có những vấn đề nào cả.

Kiểm tra sự cài đặt

- Khởi động lại cả hai gateway.
- Xem xét tập tin /var/log/messages cho nhiều thông tin về trạng thái làm việc của IPSEC. Nếu tất cả làm việc tốt bạn sẽ thấy giống như sau trong tập tin /var/log/messages hiện lên trên màn hình của bạn.

```
Feb 2 05:22:35 deep ipsec_setup: Starting FreeS/WAN IPSEC
snap2000jan31b...
Feb 2 05:22:35 deep ipsec_setup: KLIPS debug `none'
Feb 2 05:22:35 deep ipsec_setup: KLIPS ipsec0 on eth0
192.168.1.1/255.255.255.0 broadcast 192.168.1.255
Feb 2 05:22:36 deep ipsec_setup: Disabling core dumps:
Feb 2 05:22:36 deep ipsec_setup: Starting Pluto (debug `none'):
```

Feb 2 05:22:37 deep ipsec_setup: Loading Pluto database `deep-mail':

Feb 2 05:22:37 deep ipsec_setup: Enabling Pluto negotiation:

Feb 2 05:22:37 deep ipsec_setup: Routing for Pluto conns `deep-mail':

Feb 2 05:22:37 deep ipsec_setup: Initiating Pluto tunnel `deep-mail':

Feb 2 05:22:39 deep ipsec_setup: 102 "deep-mail" #1: STATE_MAIN_I1: initiate

Feb 2 05:22:39 deep ipsec_setup: 104 "deep-mail" #1: STATE_MAIN_I2: from STATE_MAIN_I1;sent MI2, expecting MR2

Feb 2 05:22:39 deep ipsec_setup: 106 "deep-mail" #1: STATE_MAIN_I3: from STATE_MAIN_I2;sent MI3, expecting MR3

Feb 2 05:22:39 deep ipsec_setup: 004 "deep-mail" #1: STATE_MAIN_I4: SA established

Feb 2 05:22:39 deep ipsec_setup: 110 "deep-mail" #2: STATE_QUICK_I1: initiate

Feb 2 05:22:39 deep ipsec_setup: 004 "deep-mail" #2: STATE_QUICK_I2: SA established

Feb 2 05:22:39 deep ipsec_setup: ...FreeS/WAN IPSEC started

- Khám xét tập tin "/var/log/secure" cho nhiều dấu hiệu để sửa lỗi, Nếu tất cả làm việc tốt bạn sẽ thấy giống như sau trong tập tin /var/log/secure hiện lên trên màn hình của bạn.

Feb 21 14:45:42 deep Pluto[432]: Starting Pluto (FreeS/WAN Version 1.3)

Feb 21 14:45:43 deep Pluto[432]: added connection description "deep-mail"

Feb 21 14:45:43 deep Pluto[432]: listening for IKE messages

Feb 21 14:45:43 deep Pluto[432]: adding interface ipsec0/eth0 192.168.1.1

Feb 21 14:45:43 deep Pluto[432]: loading secrets from "/etc/ipsec.secrets"

Feb 21 14:45:43 deep Pluto[432]: "deep-mail" #1: initiating Main Mode

Feb 21 14:45:44 deep Pluto[432]: "deep-mail" #1: ISAKMP SA established

Feb 21 14:45:44 deep Pluto[432]: "deep-mail" #2: initiating Quick Mode

POLICY_RSASIG+POLICY_ENCRYPT+POLICY_AUTHENTICATE+POLICY_TUNNEL+POLICY_PFS

Feb 21 14:45:46 deep Pluto[432]: "deep-mail" #2: sent QI2, IPsec SA established

Feb 21 14:45:47 deep Pluto[432]: "deep-mail" #3: responding to Main Mode

Feb 21 14:45:49 deep Pluto[432]: "deep-mail" #3: sent MR3, ISAKMP SA established

Feb 21 14:45:49 deep Pluto[432]: "deep-mail" #4: responding to Quick Mode

Feb 21 14:45:50 deep Pluto[432]: "deep-mail" #4: IPsec SA established

- Trên cả hai gateway toàn bộ những tập tin sau sẽ có trong thư mục "/proc/net/"

```
[root@deep /]# ls -l /proc/net/ipsec_*
-r--r--r-- 1 root root 0 Feb 2 05:30 /proc/net/ipsec_eroute
-r--r--r-- 1 root root 0 Feb 2 05:30 /proc/net/ipsec_klipsdebug
-r--r--r-- 1 root root 0 Feb 2 05:30 /proc/net/ipsec_spi
-r--r--r-- 1 root root 0 Feb 2 05:30 /proc/net/ipsec_spigrp
-r--r--r-- 1 root root 0 Feb 2 05:30 /proc/net/ipsec_spinew
-r--r--r-- 1 root root 0 Feb 2 05:30 /proc/net/ipsec_tncfg
-r--r--r-- 1 root root 0 Feb 2 05:30 /proc/net/ipsec_version
```

- Các interface IPSEC sẽ được gắn trên đầu của các interface vật lý được chỉ định.

Xác thực chúng với lệnh:

```
[root@deep /]# cat /proc/net/ipsec_tncfg
ipsec0 -> eth0 mtu=16260 -> 1500
ipsec1 -> NULL mtu=0 -> 0
ipsec2 -> NULL mtu=0 -> 0
ipsec3 -> NULL mtu=0 -> 0
```

- Bây giờ thực thi lệnh sau để trình bày thông tin debug nhỏ nhất và sẽ thấy thông tin xuất hiện trên màn hình giống như sau:

```
[root@deep /]# ipsec look
deep.openna.com Fri Feb 4 17:25:17 EST 2000

============
192.168.1.1/32 -> 192.168.1.2/32 => tun0x106@192.168.1.2
esp0x4450894d@192.168.1.2 ah0x4450894c@192.168.1.2
------------
ah0x3350f551@192.168.1.1 AH_HMAC_MD5: dir=in ooowin=32 seq=115
bit=0xffffffff alen=128 aklen=16
life(c,s,h)=bytes(16140,0,0)add(51656,0,0)use(54068,0,0)packets(115,0,0)
idle=499
ah0x4450894c@192.168.1.2 AH_HMAC_MD5: dir=out ooowin=32 seq=2828
alen=128 aklen=16
life(c,s,h)=bytes(449488,0,0)add(51656,0,0)use(51656,0,0)packets(2828,0,0
) idle=6
esp0x3350f552@192.168.1.1 ESP_3DES: dir=in ooowin=32 seq=115
bit=0xffffffff eklen=24
life(c,s,h)=bytes(13380,0,0)add(51656,0,0)use(54068,0,0)packets(115,0,0)
idle=499
esp0x4450894d@192.168.1.2 ESP_3DES: dir=out ooowin=32 seq=2828
eklen=24
```

life(c,s,h)=bytes(381616,0,0)add(51656,0,0)use(51656,0,0)packets(2828,0,0
) idle=6

tun0x105@192.168.1.1 IPIP: dir=in 192.168.1.2 -> 192.168.1.1
life(c,s,h)=add(51656,0,0)

tun0x106@192.168.1.2 IPIP: dir=out 192.168.1.1 -> 192.168.1.2

life(c,s,h)=bytes(327581,0,0)add(51656,0,0)use(51656,0,0)packets(2828,0,0
) idle=6

192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0

192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 ipsec0

192.168.1.1 0.0.0.0 255.255.255.255 UH 0 0 0 eth0

192.168.1.2 192.168.1.2 255.255.255.255 UGH 0 0 0 ipsec0

Destination Gateway Genmask Flags MSS Window irtt Iface

- Thử ping 192.168.1.2 từ client 192.168.1.1. Nếu cái này làm việc thì khi đó bạn đã cài đặt chính xác. Nếu chúng không làm việc hãy kiểm tra lại network của bạn và bảo đảm rằng 208.164.186.1 có thể gởi đến 208.164.186.2 và đảm bảo rằng TCP/IP forwarding đã được làm cho có hiệu lực và cũng bảo đảm rằng không có những qui tắc khóa package cần cho IPSEC hoặc thử masquerade chúng trước những qui tắc cho phép lưu thông liên quan tới IPSec. Để kiểm tra việc này dùng lệnh ping từ một subnet này đến một subnet khác ví dụ như:

  Kiến trúc mạng như sau:

  208.164.186.1 ---- 205.151.222.250 ---- 205.151.222.251 ---- 208.164.186.2
  |
  192.168.1.0/24
  192.168.1.0/24
  |
  192.168.1.1

  192.168.1.2

Một chú ý cuối cùng về việc kiểm tra tiến trình cài đặt FreeSWAN IPSEC, nếu bạn phát hiện một vấn đề mà bạn không thể giải quyết, bạn có thể dùng lệnh sau để ghi lại toàn bộ thông tin (bao gồm nội dung của các file, logfile ...)liên quan tới IPSEc gởi tới Linux-IPSec Mailing List (linux-ipsec@clinet.fi) để có sự giúp đỡ.

- Dùng lệnh sau để ghi lại thông tin như nói ở trên:

```
[root@deep /]# ipsec barf > result
```

Lệnh này được cung cấp như một tiện ích cho việc sửa lỗi từ xa, nó ghi lại tất cả các thông tin liên quan để chẩn đoán một vấn đề trong IPSEC.

Các lệnh cần tham khảo thêm

Chúng tôi không có ý định diễn giải tất cả các lệnh dưới đây, bạn có thể tham khảo bằng cách dùng lệnh man với những lệnh bạn cần biết.

$ man ipsec (8) - invoke IPSEC utilities

$ man ipsec atoaddr, addrtoa (3) - convert Internet addresses to and from ASCII
$ man ipsec atoasr (3) - convert ASCII to Internet address, subnet, or range
$ man ipsec atobytes, bytestoa (3) - convert binary data bytes from and to ASCII formats
$ man ipsec atodata, datatoa (3) - convert binary data from and to ASCII formats
$ man ipsec atosa, satoa (3) - convert IPSEC Security Association IDs to and from ASCII
$ man ipsec atosubnet, subnettoa (3) - convert subnet/mask ASCII form to and from addresses
$ man ipsec atoul, ultoa (3) - convert unsigned-long numbers to and from ASCII
$ man ipsec auto (8) - control automatically-keyed IPSEC connections
$ man ipsec barf (8) - spew out collected IPSEC debugging information
$ man ipsec bitstomask (3) - convert bit count to Internet subnet mask
$ man ipsec eroute (8) - manipulate IPSEC extended routing tables
$ man ipsec goodmask (3) - is this Internet subnet mask a valid one?
$ man ipsec hostof (3) - given Internet address and subnet mask, return host part
$ man ipsec klipsdebug (8) - set Klips (kernel IPSEC support) debug features and level
$ man ipsec look (8) - show minimal debugging information
$ man ipsec manual (8) - take manually-keyed IPSEC connections up and down
$ man ipsec masktobits (3) - convert Internet subnet mask to bit count
$ man ipsec optionsfrom (3) - read additional ``command-line'' options from file
$ man ipsec pluto (8) - IPsec IKE keying daemon
$ man ipsec ranbits (8) - generate random bits in ASCII form
$ man ipsec rangetoa (3) - convert Internet address range to ASCII
$ man ipsec rsasigkey (8) - generate RSA signature key
$ man ipsec setup (8) - control IPSEC subsystem
$ man ipsec spi (8) - manage IPSEC Security Associations
$ man ipsec spigrp (8) - group/ungroup IPSEC Security Associations
$ man ipsec subnetof (3) - given Internet address and subnet mask, return subnet number
$ man ipsec tncfg (8) - associate IPSEC virtual interface with real interface
$ man ipsec whack (8) - control interface for IPSEC keying daemon
$ man ipsec.conf (5) - IPSEC configuration and connections
$ man ipsec.secrets (5) - secrets for IKE/IPsec authentication
$ man ipsec (8) - invoke IPSEC utilities
$ man ipsec atoaddr, addrtoa (3) - convert Internet addresses to and from ASCII
$ man ipsec atoasr (3) - convert ASCII to Internet address, subnet, or range
$ man ipsec atobytes, bytestoa (3) - convert binary data bytes from and to ASCII formats
$ man ipsec atodata, datatoa (3) - convert binary data from and to ASCII formats
$ man ipsec atosa, satoa (3) - convert IPSEC Security Association IDs to and from ASCII
$ man ipsec atosubnet, subnettoa (3) - convert subnet/mask ASCII form to and from addresses

$ man ipsec atoul, ultoa (3)	- convert unsigned-long numbers to and from ASCII
$ man ipsec auto (8)	- control automatically-keyed IPSEC connections
$ man ipsec barf (8)	- spew out collected IPSEC debugging information
$ man ipsec bitstomask (3)	- convert bit count to Internet subnet mask
$ man ipsec eroute (8)	- manipulate IPSEC extended routing tables
$ man ipsec goodmask (3)	- is this Internet subnet mask a valid one?
$ man ipsec hostof (3)	- given Internet address and subnet mask, return host part
$ man ipsec klipsdebug (8)	- set Klips (kernel IPSEC support) debug features and level
$ man ipsec look (8)	- show minimal debugging information
$ man ipsec manual (8)	- take manually-keyed IPSEC connections up and down
$ man ipsec masktobits (3)	- convert Internet subnet mask to bit count
$ man ipsec optionsfrom (3)	- read additional "command-line" options from file
$ man ipsec pluto (8)	- IPsec IKE keying daemon
$ man ipsec ranbits (8)	- generate random bits in ASCII form
$ man ipsec rangetoa (3)	- convert Internet address range to ASCII
$ man ipsec rsasigkey (8)	- generate RSA signature key
$ man ipsec setup (8)	- control IPSEC subsystem
$ man ipsec spi (8)	- manage IPSEC Security Associations
$ man ipsec spigrp (8)	- group/ungroup IPSEC Security Associations
$ man ipsec subnetof (3)	- given Internet address and subnet mask, return subnet number
$ man ipsec tncfg (8)	- associate IPSEC virtual interface with real interface
$ man ipsec whack (8)	- control interface for IPSEC keying daemon
$ man ipsec.conf (5)	- IPSEC configuration and connections
$ man ipsec.secrets (5)	- secrets for IKE/IPsec authentication

Các tập tin được cài đặt

> /etc/rc.d/init.d/ipsec
> /etc/rc.d/rc0.d/K68ipsec
> /etc/rc.d/rc1.d/K68ipsec
> /etc/rc.d/rc2.d/S47ipsec
> /etc/rc.d/rc3.d/S47ipsec
> /etc/rc.d/rc4.d/S47ipsec
> /etc/rc.d/rc5.d/S47ipsec
> /etc/rc.d/rc6.d/K68ipsec
> /etc/ipsec.conf
> /etc/ipsec.secrets
> /usr/lib/ipsec
> /usr/lib/ipsec/spi
> /usr/lib/ipsec/eroute
> /usr/lib/ipsec/spigrp
> /usr/lib/ipsec/tncfg
> /usr/lib/ipsec/klipsdebug
> /usr/man/man3/ipsec_atoasr.3
> /usr/man/man3/ipsec_rangetoa.3
> /usr/man/man3/ipsec_atodata.3
> /usr/man/man3/ipsec_atobytes.3
> /usr/man/man3/ipsec_bytestoa.3
> /usr/man/man3/ipsec_datatoa.3
> /usr/man/man3/ipsec_atosa.3
> /usr/man/man3/ipsec_satoa.3
> /usr/man/man3/ipsec_atoul.3
> /usr/man/man3/ipsec_ultoa.3
> /usr/man/man3/ipsec_goodmask.3
> /usr/man/man3/ipsec_masktobits.3
> /usr/man/man3/ipsec_bitstomask.3
> /usr/man/man3/ipsec_optionsfrom.3
> /usr/man/man3/ipsec_subnetof.3
> /usr/man/man3/ipsec_hostof.3

> /usr/lib/ipsec/pluto
> /usr/lib/ipsec/whack
> /usr/lib/ipsec/ipsec
> /usr/lib/ipsec/barf
> /usr/lib/ipsec/manual
> /usr/lib/ipsec/auto
> /usr/lib/ipsec/look
> /usr/lib/ipsec/showdefaults
> /usr/lib/ipsec/_include
> /usr/lib/ipsec/_confread
> /usr/lib/ipsec/_keycensor
> /usr/lib/ipsec/_secretcensor
> /usr/lib/ipsec/_updown
> /usr/lib/ipsec/ranbits
> /usr/lib/ipsec/rsasigkey
> /usr/lib/ipsec/setup
> /usr/man/man3/ipsec_atoaddr.3
> /usr/man/man3/ipsec_addrtoa.3
> /usr/man/man3/ipsec_atosubnet.3
> /usr/man/man3/ipsec_subnettoa.3
> /usr/man/man3/ipsec_broadcastof.3
> /usr/man/man5/ipsec.secrets.5
> /usr/man/man5/ipsec.conf.3
> /usr/man/man8/ipsec_spi.8
> /usr/man/man8/ipsec.8
> /usr/man/man8/ipsec_eroute.8
> /usr/man/man8/ipsec_spigrp.8
> /usr/man/man8/ipsec_tncfg.8
> /usr/man/man8/ipsec_klipsdebug.8
> /usr/man/man8/ipsec_pluto.8
> /usr/man/man8/ipsec_whack.8
> /usr/man/man8/ipsec_barf.8
> /usr/man/man8/ipsec_look.8
> /usr/man/man8/ipsec_manual.8
> /usr/man/man8/ipsec_auto.8
> /usr/man/man8/ipsec_setup.8
> /usr/man/man8/ipsec_ranbits.8
> /usr/man/man8/ipsec_rsasigkey.8
> /usr/sbin/ipsec

Chương 17:

PHẦN MỀM SERVER
(Dịch vụ về cơ sở dữ liệu)

Trong chương này chúng ta sẽ đề cập đến các phần sau:

- Linux OpenLDAP Server.

  Sự cấu hình.

  Bảo mật OpenLDAP.

  Các công cụ bảo trì và khởi tạo OpenLDAP.

  Các công cụ cho các người sử dụng OpenLDAP.

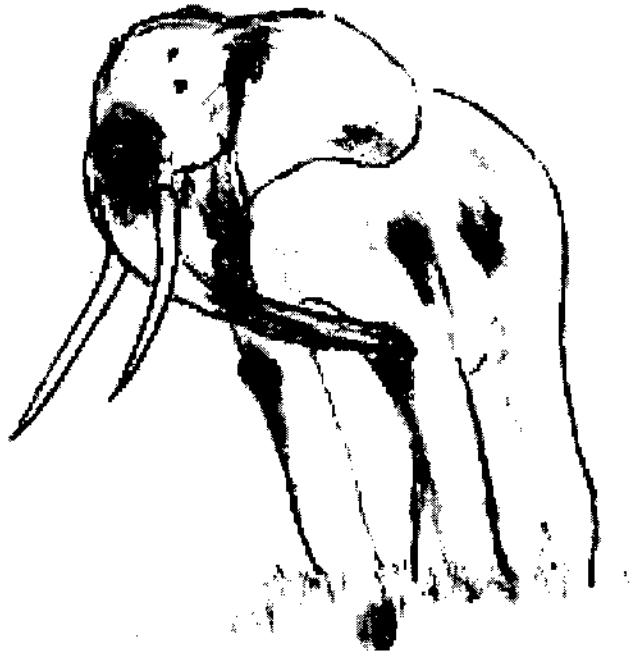
  Đặc trưng Address Book client của Netscape cho LDAP.

- Linux PostgreSQL Database Server.

  Cài đặt cơ sở dữ liệu từ tài khoản super-user PostgreSQL của bạn.

  Công việc cấu hình.

  Các lệnh.



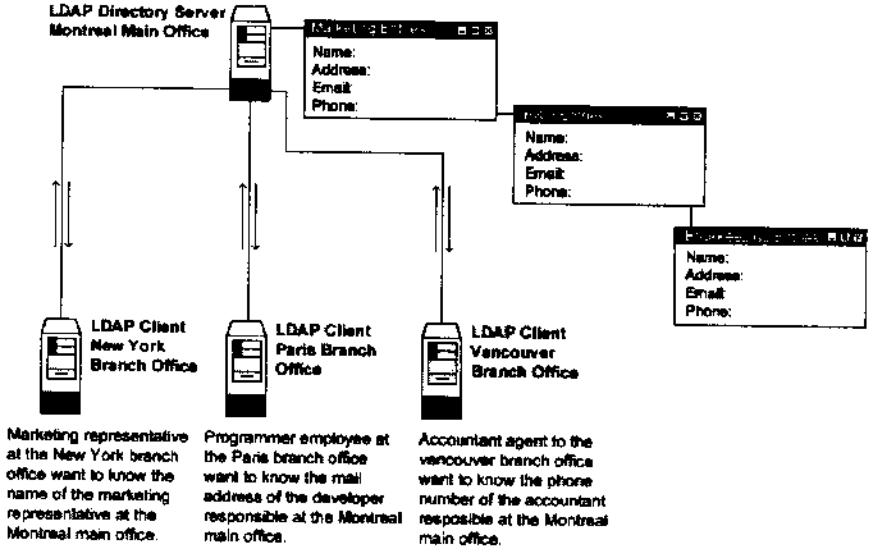
Linux OpenLDAP Server

Giới thiệu tổng quan

Cho đến giờ, chúng ta đã nói nhiều về việc bảo mật và sự tối ưu hóa trong quyển sách này, vì vậy tại sao chúng ta muốn nói về OpenLDAP? Vâng, server directory OpenLDAP sẽ mở rộng tầm nhìn của chúng ta thông qua nhiều khả năng của nó. Chúng có thể sử dụng khả năng nhân bản của nó để tập trung và củng cố các thông tin khác nhau trên một máy server cho tất cả các máy khác trong cùng hệ thống mạng của chúng ta. Hãy tưởng tượng rằng có một khả năng thêm vào hay vô hiệu hóa một tài khoản sử dụng Unix hay NT, thiết đặt truy cập tới một Web server bị hạn chế, thêm vào một địa chỉ mail hay bí danh (alias), tất cả chỉ với một thao tác đơn giản sẵn có như một dịch vụ NIS, thêm tính bảo mật của sự mã hóa SSL và tốc độ của việc phân cấp hướng đối tượng. Một việc thú vị khác được sử dụng là tạo một một danh sách các nhân viên có hiệu lực trên một hay nhiều các server LDAP có thể được truy cập từ mạng riêng của bạn, hay trên mạng internet.

Như được giải thích trong web sire OpenLDAP:

LDAP (Lightweight Directory Access Protocol) là một giao thức chuẩn-mở dành cho việc truy cập các dịch vụ thông tin. Giao thức này chạy trên các giao thức truyền internet, chẳng hạn như TCP, và có thể được dùng để truy cập các directory server hay các X.500 directory.



- Directory service cung cấp một nơi lưu trữ những thông tin về các thực thể network chẳng hạn như các ứng dụng, các file, các máy in và thông tin về con người. Nó cho phép chúng ta truy cập, quản lý và bảo vệ các thông tin trên cũng như các tài nguyên riêng rẽ.

- X.500 directory cũng là một hệ thống cho phép lưu trữ tên và địa chỉ mail của nhiều hệ thống thư tín. X500 directory cũng có thể được dùng để tạo một directory service rộng lớn không phụ thuộc. Nhiều công ty lớn đã sử dụng X500 directory xây dựng các directory cực lớn của mình.

Tóm tắt các hướng dẫn cài đặt này

Các dòng lệnh là tương thích với Unix.

Đường dẫn nguồn là "/var/tmp" (các đường dẫn khác là cũng có thể).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Phiên bản OpenLDAP là 1.2.10

Các gói cần thiết (Packages)

OpenLDAP Homepage Site: http://www.openldap.org

OpenLDAP FTP Site: 204.152.186.57

Bạn phải bảo đảm là tải về gói : openldap-1.2.10.tgz

Dạng lưu trữ Tarballs

Trước hết hãy tạo ra danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt OpenLDAP, và sau đó so sánh chúng với nhau bằng cách dùng lệnh **"diff"** để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh **"find /* > OpenLDAP1"** trước và **"find /* > OpenLDAP2"** sau khi bạn cài đặt phần mềm, và sử dụng lệnh **"diff OpenLDAP1 OpenLDAP2 > OpenLDAP-Installed"** để lấy một danh sách các thay đổi nào đó.

Sự biên dịch

Giải nén dạng lưu trữ tarball (tar.gz)

```
[root@deep /]# cp openldap-version.tgz /var/tmp
[root@deep /]# cd /var/tmp/
[root@deep tmp]# tar xzpf openldap-version.tgz
```

Biên dịch và tối ưu

Chuyển vào thư mục OpenLDAP mới và gõ vào các lệnh sau trên máy terminal của bạn:

Bước 1

Điều quan trọng phải lưu ý rằng bạn có thể cấu hình ba loại cơ sở dữ liệu hỗ trợ với LDAP. Một tính thực hiện của hệ thống cao, cơ sở dữ liệu dựa trên cơ sở đĩa được đặt tên là "LDBM"; một cơ sở dữ liệu giao tiếp với các lệnh UNIX tùy ý hoặc các shell script đặt tên là "SHELL"; và một tập tin dữ liệu mật khẩu đơn giản "PASSWD".

Việc cài đặt mặc nhiên của OpenLDAP chiếm giữ một cơ sở dữ liệu phụ trợ (backend database) LDBM, vì vậy nếu bạn muốn cấu hình một kiểu khác của cơ sở dữ liệu backend (backend database), thì bạn phải chỉ rõ nó trong suốt thời gian cấu hình và biên dịch. Đối với một SHELL backend database, bạn phải thêm vào tùy chọn "—enable-shell" và đối với một PASSWD backend database (được dùng như việc thay thế cho dịch vụ NIS) bạn phải thêm vào tùy chọn "—enable-passwd" trong vài dòng cấu hình của bạn.

```
CC="egcs" \
CFLAGS="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomit-frame-
pointer -fno-exceptions -D_REENTRANT" \
./configure \
--prefix=/usr \
--libexecdir=/usr/sbin \
--localstatedir=/var/run \
--sysconfdir=/etc \
--enable-dns \
--enable-shared \
--with-gnu-ld \
--disable-debug
```

Việc này bảo OpenLDAP cài đặt chính nó dành cho việc cài đặt phần cứng quen thuộc như sau:

- Enable dns support.
- Build shared libraries.
- **Assume the C compiler uses GNU ld.**

***Ghi chú**: Các tùy chọn biên dịch ở trên giả sử rằng bạn muốn cài đặt một cơ sở dữ liệu backend LDBM. Đối với loại cơ sở dữ liệu backend khác, bạn phải thêm vào tùy chọn được yêu cầu trong các dòng cấu hình của bạn ở trên.*

Bước 2

Bây giờ, chúng ta phải biên dịch và cài đặt OpenLDAP trong máy server:

```
[root@deep openldap-1.2.10]# make depend
[root@deep openldap-1.2.10]# make
[root@deep openldap-1.2.10]# cd tests/
[root@deep tests]# make
[root@deep tests]# cd ..
[root@deep openldap-1.2.10]# make install
```

Lệnh "make depend" sẽ xây dựng và tạo các sự phụ thuộc cần thiết cho các tập tin khác nhau, "make" biên dịch tất cả các tập tin nguồn sang các dạng nhị phân có thể thi hành được, và sau đó "make install" cài đặt các tập tin nhị phân và bất kỳ các tập tin hỗ trợ vào các vị trí phù hợp.

Lệnh "make" trong thư mục con "test" sẽ thực hiện một số các thử nghiệm quan trọng để kiểm tra các chức năng của máy server LDAP của bạn trước khi cài đặt. Nếu một số thử nghiệm thất bại, thì bạn sẽ cần sửa các lỗi trước khi tiếp tục cài đặt.

```
[root@deep openldap-1.2.10]# install -d -m 700 /var/ldap
[root@deep openldap-1.2.10]# echo localhost > /etc/openldap/ldapserver
[root@deep openldap-1.2.10]# strip /usr/lib/liblber.so.1.0.0
[root@deep openldap-1.2.10]# strip /usr/lib/libldap.so.1.0.0
[root@deep openldap-1.2.10]# strip /usr/lib/libldap.a
[root@deep openldap-1.2.10]# strip /usr/lib/liblber.a
[root@deep openldap-1.2.10]# strip /usr/sbin/in.xfingerd
[root@deep openldap-1.2.10]# strip /usr/sbin/go500
[root@deep openldap-1.2.10]# strip /usr/sbin/go500gw
[root@deep openldap-1.2.10]# strip /usr/sbin/mail500
[root@deep openldap-1.2.10]# strip /usr/sbin/rp500
[root@deep openldap-1.2.10]# strip /usr/sbin/rcpt500
[root@deep openldap-1.2.10]# strip /usr/sbin/fax500
[root@deep openldap-1.2.10]# strip /usr/sbin/slapd
[root@deep openldap-1.2.10]# strip /usr/sbin/slurpd
[root@deep openldap-1.2.10]# strip /usr/sbin/ldif
[root@deep openldap-1.2.10]# strip /usr/sbin/ldif2ldbm
[root@deep openldap-1.2.10]# strip /usr/sbin/ldif2index
[root@deep openldap-1.2.10]# strip /usr/sbin/ldif2id2entry
[root@deep openldap-1.2.10]# strip /usr/sbin/ldif2id2children
[root@deep openldap-1.2.10]# strip /usr/sbin/ldbmcat
[root@deep openldap-1.2.10]# strip /usr/sbin/ldbmtest
[root@deep openldap-1.2.10]# strip /usr/sbin/centipede
[root@deep openldap-1.2.10]# strip /usr/bin/ud
[root@deep openldap-1.2.10]# strip /usr/bin/ldapadd
[root@deep openldap-1.2.10]# strip /usr/bin/ldapsearch
[root@deep openldap-1.2.10]# strip /usr/bin/ldapmodify
[root@deep openldap-1.2.10]# strip /usr/bin/ldapmodrdn
[root@deep openldap-1.2.10]# strip /usr/bin/ldappasswd
[root@deep openldap-1.2.10]# strip /usr/bin/ldapdelete
```

Lệnh "install" ở trên sẽ tạo một thư mục mới đặt tên là "ldap" trong thư mục "/var" và sẽ đặt mode là chỉ có thể đọc (readable), có thể ghi (writable), và có thể thi hành (executable) bởi super-user "root" (700) dành cho các lý do bảo mật.

Lệnh "strip" sẽ loại bỏ tất cả các ký hiệu từ các tập tin đối tượng. Điều này có nghĩa là các tập tin nhị phân của chúng ta sẽ nhỏ hơn về kích thước. Điều này

sẽ cải tiến việc thực thi vì có một ít dòng hơn sẽ được đọc bởi hệ thống khi nó thi hành tập tin nhị phân.

Dọn dẹp sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf openldap-version/ openldap-version.tgz
```

Lệnh "rm" sẽ xoá tất cả các tập tin nguồn mà chúng ta phải được dùng để biên dịch và cài đặt OpenLDAP. Nó cũng sẽ xóa dạng lưu trữ nén OpenLDAP từ thư mục "/var/tmp".

Sự cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách

- Để chạy OpenLDAP server, các tập tin sau được yêu cầu và phải được tạo hay được sao chép vào các thư mục thích hợp trên máy server của bạn:

  Sao chép tập tin slapd.conf vào trong thư mục "/etc/openldap/".

  Sao chép tập tin script lapd vào trong thư mục "/etc/rc.d/init.d".

Cấu hình cho tập tin "/etc/ldap/slapd.conf"

Tập tin "/etc/ldap/slapd.conf" là tập tin cấu hình chính cho tiến trình stand-alone LDAP daemon. Có các tùy chọn như: permission, password, database type, database location và v.v..có thể được cấu hình trong tập tin này và sẽ kết gắn hoàn toàn cho trình slapd daemon. Trong ví dụ bên dưới chúng ta cấu hình tập tin "slapd.conf" cho một cơ sở dữ liệu LDBM backend database.

Soạn thảo tập tin slapd.conf (vi etc/openldap/slapd.conf) và thêm/ điều chỉnh phù hợp thông tin sau:

```
#
# See slapd.conf(5) for details on configuration options.
# This file should NOT be world readable.
#
include         /etc/openldap/slapd.at.conf
include         /etc/openldap/slapd.oc.conf
schemacheck     off
#referral        ldap://ldap.itd.umich.edu

pidfile         /var/run/slapd.pid
argsfile        /var/run/slapd.args

########################################################################
# ldbm database definitions
########################################################################
```

```
database        ldbm
suffix          "o=openna, c=com"
directory       /var/ldap
rootdn          "cn=admin, o=openna, c=com"
rootpw          secret
# cleartext passwords, especially for the rootdn, should
    # be avoid. See slapd.conf(5) for details.

# ldbm indexed attribute definitions
index cn,sn,uid
index objectclass pres,eq
index default none
# ldbm access control definitions
defaultaccess read
access to attr=userpassword
        by self write
        by dn="cn=admin, o=openna, c=com" write
            by * compare
```

Bạn nên bảo đảm rằng đã cài đặt các tùy chọn sau trong tập tin "slapd.conf" của bạn ở trên trước khi khởi động chương trình daemon slapd:

suffix ***"o=openna, c=com"***

Tùy chọn này chỉ rõ DN của root của cây thư mục con mà bạn đã tạo. Nói cách khác, nó chỉ ra các entry nào được nắm giữ bởi cơ sở dữ liệu.

directory ***/var/ldap***

Tùy chọn này chỉ ra thư mục nào mà cơ sở dữ liệu và các tập tin chỉ mục của LDAP có liên quan. Chúng ta phải đặt tùy chọn này là "/var/ldap" bởi vì chúng ta tạo được thư mục này sớm hơn trong giai đoạn cài đặt để xử lý cơ sở dữ liệu backend của LDAP.

rootdn ***"cn=admin, o=openna, c=com"***

Tùy chọn này chỉ rõ DN của một entry được phép thực hiện bất cứ việc gì trên thư mục LDAP. Tên được nhập vào ở đây có thể là một tên mà nó không thật sự tồn tại trong tập tin chứa thông tin mật khẩu "/etc/passwd" của bạn.

rootpw ***secret***

Tùy chọn này chỉ rõ mật khẩu có thể được sử dụng kiểm tra tính xác thực "super-user" để vào cơ sở dữ liệu. Đây là mật khẩu cho tùy chọn rootdn ở trên. Điều quan trọng là không được dùng clear text password ở đây, và thay vào đó là phải dùng một crypto password.

index cn,sn,uid | index objectclass pres,eq | index default none

Các tùy chọn này chỉ rõ các định nghĩa index mà bạn muốn xây dựng và bảo trì trong việc định nghĩa cơ sở dữ liệu. Các tùy chọn mà chúng ta chỉ ra trong tập tin ví dụ "slapd.conf" ở trên, mục đích các index được duy trì cho các thuộc tính cho cn, sn và uid (index cn,sn,uid), sự hiện diện của các index là bình đẳng như nhau cho thuộc tính objectclass (index objectclass pres,eq), và không có index nào cho tất cả các thuộc tính còn lại (index default none). Xem phần hướng dẫn người sử dụng để biết thêm nhiều thông tin hơn.

Các tùy chọn sau cùng trong tập tin "slapd.conf" liên quan với việc điều khiển truy cập trong thư mục LDAP.

```
defaultaccess read
access to attr=userpassword
        by self write
        by dn="cn=admin, o=openna, c=com" write
            by * compare
```

Ví dụ này gắn liền các mục(entry) trong mô hình cây "o=openna, c=com". Quyền truy cập Read là được gán cho mọi người, và entry của chính nó có thể được ghi ra tất cả các thuộc tính, ngoại trừ userpassword. Thuộc tính userpassword chỉ có thể được ghi bởi entry cn (admin), và có thể được so sánh bởi mọi người khác. Xem phần hướng dẫn người dùng của bạn để biết thêm thông tin.

Cấu hình cho tập tin script "/etc/rc.d/init.d/ldap"

Tập tin script "/etc/rc.d/init.d/ldap" để khởi động và dừng LDAP Server.

Tạo tập tin ldap (touch /etc/rc.d/init.d/ldap) và thêm vào:

```
#!/bin/sh
#
# ldap This shell script takes care of starting and stopping
# ldap servers (slapd and slurpd).
#
# chkconfig: - 70 40
# description: LDAP stands for Lightweight Directory Access Protocol, used \
#          for implementing the industry standard directory services.
# processname: slapd
# config: /etc/openldap/slapd.conf
    # pidfile: /var/run/slapd.pid

# Source function library.
. /etc/rc.d/init.d/functions
# Source networking configuration.
. /etc/sysconfig/network
```

```
# Check that networking is up.
    [ ${NETWORKING} = "no" ] && exit 0

[ -f /usr/sbin/slapd ] || exit 0
[ -f /usr/sbin/slurpd ] || exit 0

    RETVAL=0

    # See how we were called.
case "$1" in
  start)
    # Start daemons.
    echo -n "Starting ldap: "
    daemon slapd
    RETVAL=$?
    if [ $RETVAL -eq 0 ]; then
      if grep -q "^replogfile" /etc/openldap/slapd.conf; then
        daemon slurpd
        RETVAL=$?
        [ $RETVAL -eq 0 ] && pidof slurpd | cut -f 1 -d " " > /var/run/slurpd
      fi
    fi
    echo
    [ $RETVAL -eq 0 ] && touch /var/lock/subsys/ldap
    ;;
  stop)
# Stop daemons.
    echo -n "Shutting down ldap: "
    killproc slapd
    RETVAL=$?
    if [ $RETVAL -eq 0 ]; then
      if grep -q "^replogfile" /etc/openldap/slapd.conf; then
        killproc slurpd
        RETVAL=$?
      fi
    fi
    echo
    if [ $RETVAL -eq 0 ]; then
```

```
        rm -f /var/lock/subsys/ldap
        rm -f /var/run/slapd.args
    fi
    ;;
    status)
    status slapd
    RETVAL=$?
    if [ $RETVAL -eq 0 ]; then
      if grep -q "^replogfile" /etc/openldap/slapd.conf; then
        status slurpd
        RETVAL=$?
      fi
    fi
    ;;
restart)
    $0 stop
    $0 start
    RETVAL=$?
    ;;
reload)
    killproc -HUP slapd
    RETVAL=$?
    if [ $RETVAL -eq 0 ]; then
      if grep -q "^replogfile" /etc/openldap/slapd.conf; then
        killproc -HUP slurpd
        RETVAL=$?
      fi
    fi
    ;;
*)
        echo "Usage: $0 start|stop|restart|status}"
        exit 1
    esac

    exit $RETVAL
```

Và bây giờ, tạo script có thể thi hành này và thay đổi các quyền mặc định của nó:

```
[root@deep /]# chmod 700 /etc/rc.d/init.d/ldap
```

Tạo các mối liên kết bằng hình ảnh rc.d cho OpenLDAP với lệnh:

```
[root@deep /]# chkconfig --add ldap
```

Script OpenLDAP sẽ không được tự động khởi động chương trình daemon slapd khi bạn khởi động lại máy server. Bạn có thể thay đổi các mặc định của nó bằng cách thi hành lệnh sau:

```
[root@deep /]# chkconfig --level 345 ldap on
```

Khởi động OpenLDAP Server thực hiện bằng tay với lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/ldap start
Starting ldap:                                [ OK ]
```

Bảo mật OpenLDAP

Bảo vệ các tập tin cấu hình quan trọng.

Bit không thể thay đổi được dùng để ngăn chặn nó từ sự cố xóa hay chép chồng một tập tin mà nó phải được bảo vệ. Việc này cũng ngăn chặn mọi người từ việc tạo một mối liên kết hình ảnh tới tập tin này. Một khi tập tin "slapd.conf" của bạn đã được cấu hình, một ý tưởng hay để phòng ngừa nó với lệnh như sau:

```
[root@deep /]# chattr +i /etc/openldap/slapd.conf
```

Tài liệu bổ sung

Để cho chi tiết hơn có một vài trang man mà bạn có thể đọc:

$ man ldapd (8)	· LDAP X.500 Protocol Daemon
$ man ldapdelete (1)	· ldap delete entry tool
$ man ldapfilter.conf (5)	· configuration file for LDAP get filter routines
$ man ldapfriendly (5)	· data file for LDAP friendly routines
$ man ldapmodify, ldapadd (1)	· ldap modify entry and ldap add entry tools
$ man ldapmodrdn (1)	· ldap modify entry RDN tool
$ man ldappasswd (1)	· change the password of an LDAP entry
$ man ldapsearch (1)	· ldap search tool
$ man ldapsearchprefs.conf (5)	· configuration taäp tin for LDAP search preference routines
$ man ldaptemplates.conf (5)	· configuration taäp tin for LDAP display template routines
$ man ldif (5)	· LDAP Data Interchange Format
$ man slapd (8)	· Stand-alone LDAP Daemon
$ man slapd.conf (5)	· configuration taäp tin for slapd, the stand alone LDAP daemon
$ man slurpd (8)	· Standalone LDAP Update Replication Daemon
$ man ud (1)	· interactive LDAP Directory Server query program

Các công cụ bảo trì và khởi tạo OpenLDAP

Việc tạo LDMB backend database

Có hai phương pháp để tạo một cơ sở dữ liệu cho LDAP, đầu tiên là phương pháp off-line với tiện ích lệnh "ldif2ldbm" và một phương pháp khác là on-line với tiện ích lệnh "ldapadd". Thông thường thì bạn dùng phương pháp off-line khi bạn có hàng ngàn entry chèn vào trong cơ sở dữ liệu của bạn và phương pháp on-line khi bạn chỉ có một số nhỏ entry đưa vào trong cơ sở dữ liệu của bạn. Một điều cũng quan trọng đáng chú ý mà phương pháp off-line yêu cầu là trình daemon "slapd" của bạn không được chạy còn phương pháp on-line thì yêu cầu trình daemon "slapd" của LDAP là đang chạy.

Việc tạo LDMB backend database off-line với công cụ "ldif2ldbm"

Việc đầu tiên để làm là tạo một tập tin đưa vào (input) LDIF chứa đựng một văn bản đại diện cho các entry của bạn. Tập tin văn bản này được đặt tên là "my-data-file" bên dưới có thể được dùng như một tập tin ví dụ (dĩ nhiên, tập tin input LDIF thật sự sẽ xử lý các thông tin nhiều hơn ví dụ này). Khi bạn cài đặt OpenLDAP trong lần đầu tiên và có các entry lớn để nhập vào trong cơ sở dữ liệu backend database, đó là một ý tưởng hay để đưa vào tất cả thông tin này vào trong một tập tin văn bản và thêm chúng vào backend database của bạn với tiện ích lệnh "ldif2ldbm".

Bước 1

Tạo tập tin my-data-file (touch /tmp/my-data-file) và thêm vào như một ví dụ trong tập tin này các dòng như sau:

```
dn: o=openna, c=com
o: openna
objectclass: organization

dn: cn=Gerhard Mourani, o=openna, c=com
cn: Gerhard Mourani
sn: Mourani
mail: gmourani@videotron.ca
title: Author
objectclass: person

dn: cn=Anthony Bay, o=openna, c=com
cn: Anthony Bay
sn: Bay
homephone: (444) 111-2233
mobile: (444) 555-6677
```

mail: abay@openna.com
objectclass: person

dn: cn=George Parker, o=openna, c=com
cn: George Parker
sn: Parker
telephonenumber: (555) 234-5678
fax: (543) 987-6543
mobile: (543) 321-4354
description: E-Commerce
objectclass: person

Ví dụ trên trình bày cho bạn thấy cách chuyển đổi các thông tin của bạn như thế nào sang các tập tin LDIF trước khi thêm chúng vào thư mục backend mới. Tham khảo các tư liệu hay sách vở OpenLDAP của bạn để biết thêm thông tin chi tiết.

Bước 2

Một khi tập tin input LDIF chứa đựng các entry của chúng ta đã được tạo, chúng ta phải chèn nó vào trong LDAP directory server.

- Để chèn tập tin input LDIF và tạo cơ sở dữ liệu off-line, dùng lệnh sau:

```
[root@deep tmp]# ldif2ldbm -i <inputfile> -f <slapdconfigfile>
[root@deep tmp]# ldif2ldbm -i my-data-file -f /etc/openldap/slapd.conf
```

Tùy chọn "-i" với tùy chọn <inputfile> chỉ ra vị trí của tập tin input LDIF chứa đựng các entry trong dạng văn bản để thêm vào. Tùy chọn <slapdconfigfile> chỉ ra vị trí của tập tin cấu hình slapd mà chỉ ra nơi tạo các chỉ mục (index), và các index nào được tạo, v.v...

Ghi chú: *Điều quan trọng đáng lưu ý là trình daemon "slapd" của LDAP không được khởi động trong chế độ tạo (mode of creation) này.*

Việc tạo LDMB backend database on-line với công cụ "ldapadd"

Nếu các mục (entry) trong directory server của bạn đã được tạo sẵn hoặc nếu bạn chỉ có một lượng thông tin nhỏ để chèn vào backend database, bạn sẽ thích dùng tiện ích lệnh "ldapadd" hơn để thực hiện công việc on-line của bạn. Lấy ví dụ, để thêm entry "Europe Mourani" dùng công cụ "ldapadd", bạn có thể tạo một tập tin được gọi là "newentry" trong thư mục "/tmp".

Bước 1

Tạo tập tin newentry (touch /tmp/newentry) và thêm vào trong tập tin này các nội dung sau:

cn=Europe Mourani, o=openna, c=com
cn=Europe Mourani
sn=Mourani
mail=emourani@old.com
description=Marketing relation
objectClass=person

Bước 2

Một khi tập tin "newentry" đã được tạo, chúng ta phải thêm entry này vào trong LDAP directory server.

- Hiện tại để tạo entry on-line trong backend database, dùng lệnh sau:
  [root@deep /]# ldapadd -f /tmp/newentry -D "cn=admin, o=openna, c=com" -W
  Enter LDAP Password :

Lệnh trên hàm ý rằng là bạn đã thiết đặt rootdn với giá trị "cn=admin, o=openna, c=com" và rootpw là "secret". Bạn sẽ được nhắc để nhập vào mật mã.

Ghi chú: *Điều quan trọng đáng lưu ý là trình daemon "slapd" của LDAP được khởi động trong chế độ tạo (mode of creation) này.*

ldapmodify

Ngược lại với cơ sở dữ liệu quan hệ nơi mà dữ liệu liên tục thay đổi, directory server chứa đựng thông tin mà nó rất hiếm khi được hiệu chỉnh khi được chèn vào. Nhưng, đôi khi thì bạn cũng cần phải hiệu chỉnh thông tin, và công cụ "ldapmodify" sẽ giúp bạn trong các công việc thay đổi này của bạn. Lệnh "ldapmodify" cho phép bạn thêm hay hiệu chỉnh các entry trên backend directory server.

Giả sử rằng, chúng ta muốn thay thế nội dung của thuộc tính mail của entry "Europe Mourani" với giá trị mới emourani@new.com, các bước sau sẽ được yêu cầu:

Bước 1

Tạo tập tin modifyentry (touch /tmp/modifyentry) và thêm vào trong tập tin này nội dung như sau:

cn=Europe Mourani, o=openna, c=com
- mail=emourani@old.com # will delete the old mail address for Europe Mourani in the database.
+mail=emourani@new.com # will add the new mail address for Europe Mourani in the database.

Bước 2

Một khi tập tin "modifyentry" đã được tạo, chúng ta phải thay thế entry trong LDAP directory server với nội dung được chứa đựng trong tập tin này (modifyentry).

- Để hiệu chỉnh nội dung của backend database, dùng lệnh sau:
  [root@deep /]# ladpmodify -D 'cn=Admin, o=openna, c=com' -W -f *<inputfile>*

```
[root@deep /]# ladpmodify -D 'cn=Admin, o=openna, c=com' -W -f modifyentry
```

Trong đó *<inputfile>* là tên của tập tin "modifyentry" mà chúng ta tạo trong bước 1 ở trên.

Các công cụ cho các người sử dụng OpenLDAP

Tìm kiếm các entry trên LDAP directory server

Tiện ích ldapsearch tìm kiếm thông qua backend database của LDAP directory các thông tin mà bạn yêu cầu.

- Để tìm kiếm các entry trên LDAP directory, dùng lệnh sau:

```
[root@deep /]# ldapsearch -b 'dn' 'attrs'
[root@deep /]# ldapsearch -b 'o=openna, c=com' 'cn=europe*'
cn=Europe Mourani, o=openna, c=com
cn=Europe Mourani
sn=Mourani
mail=emourani@old.com
description=Marketing relation
objectClass=person
```

Lệnh này sẽ gọi ra tất cả các entry và các giá trị cho tên **europe** và sẽ in kết quả xuất ra trên máy terminal của bạn.

Một số công việc có thể dùng phần mềm OpenLDAP

OpenLDAP có thể được dùng như sau:

1. A Web Catalogue Server.
2. A White Pages Server.
3. A Certificate Server.
4. An Access Control Server.
5. A Network Name Server.

Đặc trưng Address Book client của Netscape cho LDAP

Nếu bạn đã cài đặt Nestcape trên một máy trạm Linux (Linux Workstation), hay thậm chí là một loại hệ điều hành khác, bạn có thể sử dụng các tính năng Address Book của nó để truy cập LDAP directory Server mà bạn vừa mới cài đặt trên Linux và truy vấn các thông tin từ directory server của bạn giống như bạn thực hiện với công cụ lệnh "ldapsearch" trên Linux.

Nếu bạn yêu thích công việc này, thì theo các bước đơn giản bên dưới:

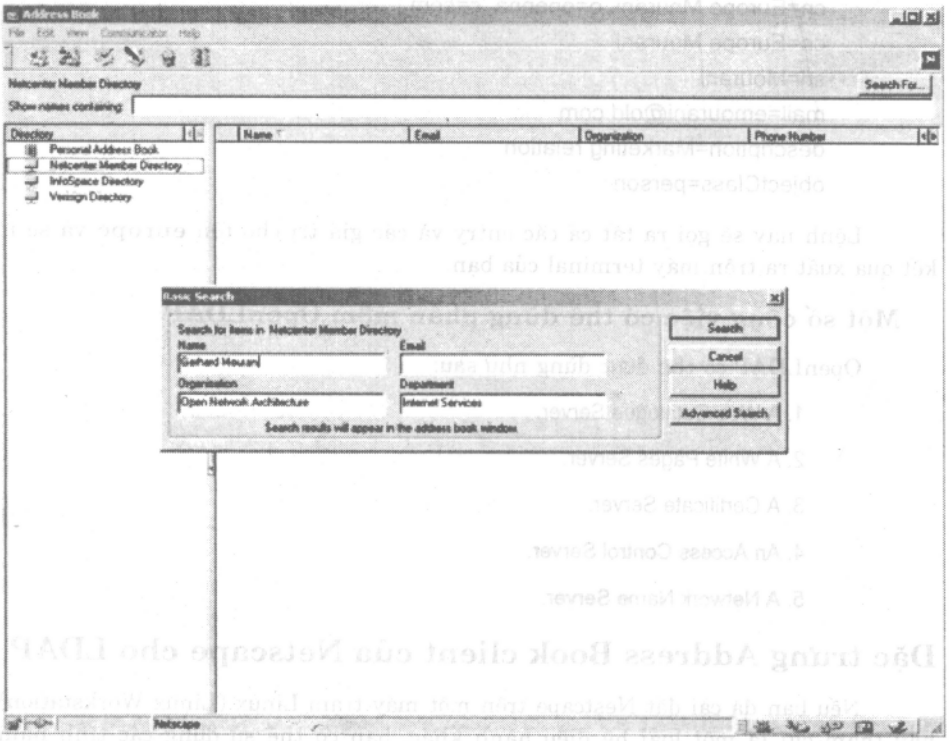
1. Mở trình duyệt Netscape Communication.
2. Vào menu Communication.
3. Mở Address Book.

4. Vào menu File.
5. Click trên New Directory...
6. Điền vào các ô nhập liệu với thông tin server của bạn.

Lấy ví dụ:

- Description: Open Network Architecture
- LDAP Server: 208.164.186.3
- Server Root: o=openna, c=com

Bây giờ tất cả các bạn phải thực hiện một số truy vấn tới LDAP Directory Server của bạn trên Linux, bằng cách dùng ô nhập liệu có tên "Show names Containing:" để khởi động việc tìm kiếm của bạn, và click trên nút "Search For:" để lấy các kết quả.



Netscape Address Book Client Program.

Các tập tin cài đặt

> /etc/openldap

> /etc/openldap/ldap.conf

> /usr/man/man3/ldap_open.3

> /usr/man/man3/ldap_errlist.3

> /etc/openldap/ldap.conf.default
> /etc/openldap/ldapfilter.conf
> /etc/openldap/ldapfilter.conf.default
> /etc/openldap/ldaptemplates.conf
> /etc/openldap/ldaptemplates.conf.default
> /etc/openldap/ldapsearchprefs.conf
> /etc/openldap/ldapsearchprefs.conf.default
> /etc/openldap/slapd.conf
> /etc/openldap/slapd.conf.default
> /etc/openldap/slapd.at.conf
> /etc/openldap/slapd.at.conf.default
> /etc/openldap/slapd.oc.conf
> /etc/openldap/slapd.oc.conf.default
> /etc/openldap/ldapserver
> /etc/rc.d/init.d/ldap
> /etc/rc.d/rc0.d/K40ldap
> /etc/rc.d/rc1.d/K40ldap
> /etc/rc.d/rc2.d/K40ldap
> /etc/rc.d/rc3.d/S70ldap
> /etc/rc.d/rc4.d/S70ldap
> /etc/rc.d/rc5.d/S70ldap
> /etc/rc.d/rc6.d/K40ldap
> /usr/bin/ud
> /usr/bin/ldapsearch
> /usr/bin/ldapmodify
> /usr/bin/ldapdelete
> /usr/bin/ldapmodrdn
> /usr/bin/ldappasswd
> /usr/bin/ldapadd
> /usr/include/ldap.h
> /usr/include/lber.h
> /usr/man/man3/ldap_err2string.3
> /usr/man/man3/ldap_first_attribute.3
> /usr/man/man3/ldap_next_attribute.3
> /usr/man/man3/ldap_first_entry.3
> /usr/man/man3/ldap_next_entry.3
> /usr/man/man3/ldap_count_entries.3
> /usr/man/man3/ldap_friendly.3
> /usr/man/man3/ldap_friendly_name.3
> /usr/man/man3/ldap_free_friendlymap.3
> /usr/man/man3/ldap_get_dn.3
> /usr/man/man3/ldap_explode_dn.3
> /usr/man/man3/ldap_explode_dns.3
> /usr/man/man3/ldap_dn2ufn.3
> /usr/man/man3/ldap_is_dns_dn.3
> /usr/man/man3/ldap_get_values.3
> /usr/man/man3/ldap_get_values_len.3
> /usr/man/man3/ldap_value_free.3
> /usr/man/man3/ldap_value_free_len.3
> /usr/man/man3/ldap_count_values.3
> /usr/man/man3/ldap_count_values_len.3
> /usr/man/man3/ldap_getfilter.3
> /usr/man/man3/ldap_init_getfilter.3
> /usr/man/man3/ldap_init_getfilter_buf.3
> /usr/man/man3/ldap_getfilter_free.3
> /usr/man/man3/ldap_getfirstfilter.3
> /usr/man/man3/ldap_getnextfilter.3
> /usr/man/man3/ldap_setfilteraffixes.3
> /usr/man/man3/ldap_build_filter.3
> /usr/man/man3/ldap_modify.3
> /usr/man/man3/ldap_modify_s.3
> /usr/man/man3/ldap_mods_free.3

> /usr/include/ldap_cdefs.h
> /usr/include/disptmpl.h
> /usr/include/srchpref.h
> /usr/lib/liblber.so.1.0.0
> /usr/lib/liblber.so.1
> /usr/lib/liblber.so
> /usr/lib/liblber.la
> /usr/lib/liblber.a
> /usr/lib/libldap.so.1.0.0
> /usr/lib/libldap.so.1
> /usr/lib/libldap.so
> /usr/lib/libldap.la
> /usr/lib/libldap.a
> /usr/man/man1/ud.1
> /usr/man/man1/ldapdelete.1
> /usr/man/man1/ldapmodify.1
> /usr/man/man1/ldapadd.1
> /usr/man/man1/ldapmodrdn.1
> /usr/man/man1/ldappasswd.1
> /usr/man/man1/ldapsearch.1
> /usr/man/man3/cldap_close.3
> /usr/man/man3/cldap_open.3
> /usr/man/man3/cldap_search_s.3
> /usr/man/man3/cldap_setretryinfo.3
> /usr/man/man3/lber-decode.3
> /usr/man/man3/lber-encode.3
> /usr/man/man3/ldap.3
> /usr/man/man3/cldap.3
> /usr/man/man3/ldap_abandon.3
> /usr/man/man3/ldap_add.3
> /usr/man/man3/ldap_add_s.3
> /usr/man/man3/ldap_modrdn.3
> /usr/man/man3/ldap_modrdn_s.3
> /usr/man/man3/ldap_modrdn2.3
> /usr/man/man3/ldap_modrdn2_s.3
> /usr/man/man3/ldap_init.3
> /usr/man/man3/ldap_result.3
> /usr/man/man3/ldap_msgfree.3
> /usr/man/man3/ldap_search.3
> /usr/man/man3/ldap_search_s.3
> /usr/man/man3/ldap_search_st.3
> /usr/man/man3/ldap_searchprefs.3
> /usr/man/man3/ldap_init_searchprefs.3
>/usr/man/man3/ldap_init_searchprefs_buf.3
> /usr/man/man3/ldap_free_searchprefs.3
> /usr/man/man3/ldap_first_searchobj.3
> /usr/man/man3/ldap_next_searchobj.3
> /usr/man/man3/ldap_sort.3
> /usr/man/man3/ldap_sort_entries.3
> /usr/man/man3/ldap_sort_values.3
> /usr/man/man3/ldap_sort_strcasecmp.3
> /usr/man/man3/ldap_ufn.3
> /usr/man/man3/ldap_ufn_search_s.3
> /usr/man/man3/ldap_ufn_search_c.3
> /usr/man/man3/ldap_ufn_search_ct.3
> /usr/man/man3/ldap_ufn_setprefix.3
> /usr/man/man3/ldap_ufn_setfilter.3
> /usr/man/man3/ldap_ufn_timeout.3
> /usr/man/man3/ldap_url.3
> /usr/man/man3/ldap_is_ldap_url.3
> /usr/man/man3/ldap_url_parse.3
> /usr/man/man3/ldap_free_urldesc.3

> /usr/man/man3/ldap_bind.3
> /usr/man/man3/ldap_bind_s.3
> /usr/man/man3/ldap_simple_bind.3
> /usr/man/man3/ldap_simple_bind_s.3
> /usr/man/man3/ldap_kerberos_bind_s.3
> /usr/man/man3/ldap_kerberos_bind1.3
> /usr/man/man3/ldap_kerberos_bind1_s.3
> /usr/man/man3/ldap_kerberos_bind2.3
> /usr/man/man3/ldap_kerberos_bind2_s.3
> /usr/man/man3/ldap_unbind.3
> /usr/man/man3/ldap_unbind_s.3
> /usr/man/man3/ldap_set_rebind_proc.3
> /usr/man/man3/ldap_cache.3
> /usr/man/man3/ldap_enable_cache.3
> /usr/man/man3/ldap_disable_cache.3
> /usr/man/man3/ldap_destroy_cache.3
> /usr/man/man3/ldap_flush_cache.3
> /usr/man/man3/ldap_uncache_entry.3
> /usr/man/man3/ldap_uncache_request.3
> /usr/man/man3/ldap_set_cache_options.3
> /usr/man/man3/ldap_charset.3
> /usr/man/man3/ldap_set_string_translators.3
> /usr/man/man3/ldap_enable_translation.3
> /usr/man/man3/ldap_translate_from_t61.3
> /usr/man/man3/ldap_translate_to_t61.3
> /usr/man/man3/ldap_t61_to_8859.3
> /usr/man/man3/ldap_8859_to_t61.3
> /usr/man/man3/ldap_compare.3
> /usr/man/man3/ldap_compare_s.3
> /usr/man/man3/ldap_delete.3
> /usr/man/man3/ldap_delete_s.3
> /usr/man/man3/ldap_url_search.3
> /usr/man/man3/ldap_url_search_s.3
> /usr/man/man3/ldap_url_search_st.3
> /usr/man/man5/ldap.conf.5
> /usr/man/man5/ldapfilter.conf.5
> /usr/man/man5/ldapfriendly.5
> /usr/man/man5/ldapsearchprefs.conf.5
> /usr/man/man5/ldaptemplates.conf.5
> /usr/man/man5/ldif.5
> /usr/man/man5/slapd.conf.5
> /usr/man/man5/slapd.replog.5
> /usr/man/man5/ud.conf.5
> /usr/man/man8/centipede.8
> /usr/man/man8/chlog2replog.8
> /usr/man/man8/edb2ldif.8
> /usr/man/man8/go500.8
> /usr/man/man8/go500gw.8
> /usr/man/man8/in.xfingerd.8
> /usr/man/man8/ldapd.8
> /usr/man/man8/ldbmcat.8
> /usr/man/man8/ldif.8
> /usr/man/man8/ldif2ldbm.8
> /usr/man/man8/ldif2index.8
> /usr/man/man8/ldif2id2entry.8
> /usr/man/man8/ldif2id2children.8
> /usr/man/man8/mail500.8
> /usr/man/man8/fax500.8
> /usr/man/man8/rcpt500.8
> /usr/man/man8/slapd.8
> /usr/man/man8/slurpd.8
> /usr/sbin/ldif

> /usr/man/man3/ldap_disptmpl.3
> /usr/man/man3/ldap_init_templates.3
> /usr/man/man3/ldap_init_templates_buf.3
> /usr/man/man3/ldap_free_templates.3
> /usr/man/man3/ldap_first_disptmpl.3
> /usr/man/man3/ldap_next_disptmpl.3
> /usr/man/man3/ldap_oc2template.3
> /usr/man/man3/ldap_tmplattrs.3
> /usr/man/man3/ldap_first_tmplrow.3
> /usr/man/man3/ldap_next_tmplrow.3
> /usr/man/man3/ldap_first_tmplcol.3
> /usr/man/man3/ldap_next_tmplcol.3
> /usr/man/man3/ldap_entry2text.3
> /usr/man/man3/ldap_entry2text_search.3
> /usr/man/man3/ldap_vals2text.3
> /usr/man/man3/ldap_entry2html.3
> /usr/man/man3/ldap_entry2html_search.3
> /usr/man/man3/ldap_vals2html.3
> /usr/man/man3/ldap_error.3
> /usr/man/man3/ldap_perror.3
> /usr/man/man3/ld_errno.3
> /usr/man/man3/ldap_result2error.3
> /usr/sbin/in.xfingerd
> /usr/sbin/go500
> /usr/sbin/go500gw
> /usr/sbin/mail500
> /usr/sbin/rp500
> /usr/sbin/fax500
> /usr/sbin/xrpcomp
> /usr/sbin/rcpt500
> /usr/sbin/slapd
> /usr/sbin/ldif2ldbm
> /usr/sbin/ldif2index
> /usr/sbin/ldif2id2entry
> /usr/sbin/ldif2id2children
> /usr/sbin/ldbmcat
> /usr/sbin/centipede
> /usr/sbin/ldbmtest
> /usr/sbin/slurpd
> /usr/share/openldap
> /usr/share/openldap/ldapfriendly
> /usr/share/openldap/go500gw.help
> /usr/share/openldap/rcpt500.help
> /var/ldap

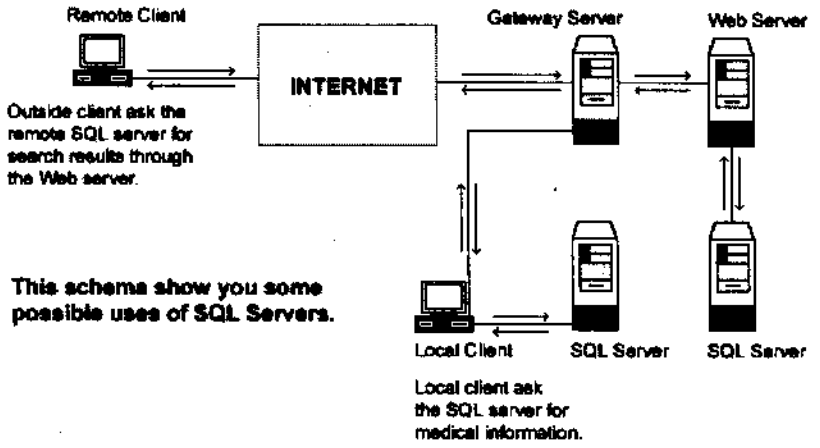
Linux PostgreSQL Database Server

Giới thiệu tổng quan

Một khi bạn bắt đầu phục vụ, và cung cấp các dịch vụ cho khách hàng của bạn, chắc hẳn bạn sẽ tìm kiếm những thông tin mà bạn cần phải nắm bắt về họ bằng một dạng lưu trữ để có thể truy cập và có thể hiệu chỉnh tại mọi thời điểm bạn muốn. Các công việc này có thể được hoàn thành với việc dùng một cơ sở dữ liệu. Có rất nhiều cơ sở dữ liệu đã có sẵn trên Linux; bạn có thể chọn một cơ sở dữ liệu đã được biên dịch, cũng như nó hẳn nhiên phải hỗ trợ một số ngôn ngữ lập trình về các tính năng đặc trưng và chuẩn. PostgreSQL là một sự lựa chọn thích hợp, PostgreSQL nguyên được phát triển từ UC Berkeley Computer Science Department (Khoa máy tính của đại học Berkeley), dựa trên rất nhiều các khái niệm về đối tượng – quan hệ (object-relational) và hiện nay đã trở thành một cơ sở dữ liệu thương mại hoá. Nó cung cấp sự hỗ trợ ngôn ngữ SQL92/SQL3, hỗ trợ tích hợp giao dịch, và sự mở rộng kiểu.

Như được giải thích trên web sire PostgreSQL:

PostgreSQL là một hệ quản trị cơ sở dữ liệu đối tượng – quan hệ tinh vi (Object-Relational DBMS), hỗ trợ hầu hết tất cả các lệnh SQL, bao gồm các subselect, các transaction, các hàm và kiểu do người dùng định nghĩa. Nó là một cơ sở dữ liệu có mã nguồn cao cấp nhất đã có sẵn ở mọi nơi.



Hướng dẫn cài đặt

Các dòng lệnh là tương thích với Unix.

Đường dẫn nguồn sẽ là "/var/tmp" (cũng có thể chọn những đường dẫn khác).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Phiên bản PostgreSQL là 6.5.3

Các gói cần thiết (Packages)

PostgreSQL Homepage Site: http://www.postgresql.org

PostgreSQL FTP Site: 216.126.84.28

Bạn phải bảo đảm là tải về gói : postgresql-6.5.3.tar.gz

Các yêu cầu cần thiết ban đầu

Trước khi biên dịch chương trình PostgreSQL, bạn phải kiểm tra rằng package egcs-c++-version.i386.rpm đã được cài đặt trên hệ thống của bạn. Package egcs-c++-version.i386.rpm sẵn có trong Red Hat Linux CD-ROM trong thư mục con "Red Hat/RPMS" của bạn. Sau khi biên dịch và cài đặt PostgreSQL bạn có thể xóa package egcs-c++-version.i386.rpm từ hệ thống của bạn.

- Để kiểm tra package egcs-c++-version.i386.rpm đã được cài đặt chưa, dùng lệnh sau:

```
[root@deep /]# rpm -q egcs-c++
```

- Để cài đặt package egcs-c++-version.i386.rpm, dùng lệnh sau:

```
[root@deep /]# mount /dev/cdrom /mnt/cdrom
[root@deep /]# cd /mnt/cdrom/Red Hat/RPMS
[root@deep RPMS]# rpm -Uvh egcs-c++-version.i386.rpm
egcs-c++
##################################################
```

Dạng lưu trữ Tarballs

Trước hết hãy tạo ra danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt PostgreSQL, và sau đó so sánh chúng với nhau bằng cách dùng lệnh **"diff"** để tìm ra các tập tin nào là được đặt ở đâu. Hãy chạy lệnh **"find /* > PostgreSQL1"** trước và **"find /* > PostgreSQL2"** sau khi bạn cài đặt phần mềm, và sử dụng lệnh **"diff PostgreSQL1 PostgreSQL2 > PostgreSQL-Installed"** để lấy một danh sách các thay đổi nào đó.

Sự biên dịch

Giải nén dạng lưu trữ tarball (tar.gz)

```
[root@deep /]# cp postgresql-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf postgresql-version.tar.gz
```

Biên dịch và tối ưu

Bước 1

Trước tiên, để tránh các rủi ro bảo mật, chúng ta phải tạo một tài khoản người sử dụng không có nhiều quyền hạn có tên "postgres" được gán quyền và sở hữu riêng các tập tin Postgres.

- Để tạo tài khoản Postgres, dùng lệnh sau:

```
[root@deep /]# useradd -M -o -r -d /var/lib/pgsql -s /bin/bash -c "PostgreSQL
Server" -u 40 postgres >/dev/null 2>&1 || :
```

Bước 2

Chuyển vào trong thư mục PostgreSQL mà chúng ta đã giải nén tar trước đó, và khi đó chuyển vào trong thư mục con của nó có tên là "src". Gõ các lệnh sau vào máy terminal của bạn:

```
[root@deep /]# cd /var/tmp/postgresql-6.5.3
[root@deep postgresql-6.5.3]# cd src
CC="egcs" \
./configure \
--prefix=/usr \
--enable-locale
```

Việc này bảo PostgreSQL tự thiết lập cấu hình cho việc cài đặt với các phần cứng riêng biệt:

- Enable locale support

Hiệu chỉnh tập tin **Makefile.global** (vi +210 Makefile.global) và thay đổi dòng:

```
CFLAGS= -I$(SRCDIR)/include -I$(SRCDIR)/backend
```

Chuyển thành :

```
CFLAGS= -I$(SRCDIR)/include -I$(SRCDIR)/backend -O9 -funroll-loops -
ffast-math -malign-double -mcpu=pentiumpro -march=pentiumpro -fomit-
frame-pointer -fno-exceptions
```

Có các cờ hiệu tối ưu hoá của chúng ta dành cho PostgreSQL Server. Dĩ nhiên, bạn phải hiệu chỉnh chúng để phù hợp với hệ thống và cấu trúc CPU của bạn.

Bước 3

Và bây giờ, chúng ta phải biên dịch và cài đặt PostgreSQL trên máy server:

```
[root@deep src]# make all
[root@deep src]# cd ..
[root@deep postgresql-6.5.3]# make -C src install
[root@deep postgresql-6.5.3]# make -C src/man install
[root@deep postgresql-6.5.3]# mkdir -p /usr/include/pgsql
```

```
[root@deep postgresql-6.5.3]# mv /usr/include/access /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/commands /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/executor /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/lib /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/libpq /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/libpq++ /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/port /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/utils /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/fmgr.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/os.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/config.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/c.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/postgres.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/postgres_ext.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/libpq-fe.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/libpq-int.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/ecpgerrno.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/ecpglib.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/ecpgtype.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/sqlca.h /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/include/libpq++.H /usr/include/pgsql/
[root@deep postgresql-6.5.3]# mkdir -p /usr/lib/pgsql
[root@deep postgresql-6.5.3]# mv /usr/lib/*source /usr/lib/pgsql/
[root@deep postgresql-6.5.3]# mv /usr/lib/*sample /usr/lib/pgsql/
[root@deep postgresql-6.5.3]# mkdir -p /var/lib/pgsql
[root@deep postgresql-6.5.3]# chown -R postgres.postgres /var/lib/pgsql/
[root@deep postgresql-6.5.3]# chmod 755 /usr/lib/libpq.so.2.0
[root@deep postgresql-6.5.3]# chmod 755 /usr/lib/libecpg.so.3.0.0
[root@deep postgresql-6.5.3]# chmod 755 /usr/lib/libpq++.so.3.0
[root@deep postgresql-6.5.3]# strip /usr/bin/postgres
[root@deep postgresql-6.5.3]# strip /usr/bin/postmaster
[root@deep postgresql-6.5.3]# strip /usr/bin/ecpg
[root@deep postgresql-6.5.3]# strip /usr/bin/pg_id
[root@deep postgresql-6.5.3]# strip /usr/bin/pg_version
[root@deep postgresql-6.5.3]# strip /usr/bin/pg_dump
[root@deep postgresql-6.5.3]# strip /usr/bin/pg_passwd
[root@deep postgresql-6.5.3]# strip /usr/bin/psql
[root@deep postgresql-6.5.3]# rm -f /usr/lib/global1.description
[root@deep postgresql-6.5.3]# rm -f /usr/lib/local1_template1.description
```

Lệnh "make" biên dịch tất cả các tập tin nguồn sang dạng nhị phân có thể thi hành, và lệnh "make install" cài đặt các tập tin nhị phân này và bất kỳ các

tập tin hỗ trợ vào các nơi thích hợp. Lệnh "mkdir" sẽ tạo một thư mục mới có tên là "pqsql" trong các thư mục "/usr/include" và "usr/lib", và sau đó chúng ta chuyển tất cả các thư mục con và các tập tin có liên quan với PostgreSQL trong các thư mục "/usr/include" và "/usr/lib" vào các thư mục "/usr/include/pgsql" và "usr/lib/pgsql" từng cái một với lệnh "**mv**".

Lệnh "**chown**" sẽ đặt chính xác các quyền sở hữu riêng (owner) và nhóm người dùng (group) cho thư mục "/var/lib/pgsql". Lệnh "strip" sẽ loại bỏ tất cả các biểu tượng từ các tập tin đối tượng (object). Điều này có nghĩa là tập tin nhị phân của bạn sẽ có kích thước nhỏ hơn. Việc này sẽ cải tiến việc thực hiện các chương trình bởi vì chỉ có vài dòng được đọc bởi hệ thống khi nó thi hành tập tin nhị phân.

Lệnh "**rm**" sẽ xoá các tập tin "global1.description" và "local1_template1.description" không cần thiết bởi chương trình PostgreSQL.

Cài đặt cơ sở dữ liệu từ tài khoản super-user PostgreSQL của bạn

Một khi PostgreSQL được cài đặt trên Linux Server của bạn, điều quan trọng là tạo cài đặt cơ sở dữ liệu trước khi khởi động PostgreSQL Server của bạn.

- Để tạo cài đặt cơ sở dữ liệu, dùng lệnh sau:

```
[root@deep /]# su postgres
[postgres@deep /]$ initdb --pglib=/usr/lib/pgsql --
pgdata=/var/lib/pgsql

We are initializing the database system with username postgres (uid=40).
This user will own all the files and must also own the server process.

Creating Postgres database system directory /var/lib/pgsql/base

Creating template database in /var/lib/pgsql/base/template1

Creating global classes in /var/lib/pgsql/base

Adding template1 database to pg_database...

Vacuuming template1
Creating public pg_user view
Creating view pg_rules
Creating view pg_views
Creating view pg_tables
Creating view pg_indexes
```

```
Loading pg_description

[postgres@deep /]$ chmod 640 /var/lib/pgsql/pg_pwd
[postgres@deep /]$ exit
exit
[root@deep /]#
```

Lệnh "--pglib" chỉ rõ nơi mà thư mục thư viện (library) của PostgreSQL trong hệ thống, và lệnh "--pgdata" chỉ rõ nơi các tập tin cơ sở dữ liệu phải nằm trong việc cài đặt Linux này.

Ghi chú: *Không tạo cài đặt cơ sở dữ liệu như "root" ! Điều này sẽ trở thành một lỗ hổng bảo mật chính yếu.*

Sự dọn dẹp sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf postgresql-version/ postgresql-version.tar.gz
```

Xoá package egcs-c++-version.i386.rpm để tiết kiệm không gian trống của đĩa.

```
[root@deep /]# rpm -e egcs-c++
```

Lệnh "**rm**" sẽ xoá tất cả các tập tin nguồn mà chúng ta dùng để biên dịch và cài đặt PostgreSQL. Nó cũng sẽ xoá dạng lưu trữ nén PostgreSQL từ thư mục "/var/tmp".

Lệnh "**rpm -e**" sẽ xoá package egcs-c++ mà chúng ta đã cài đặt để biên dịch PostgreSQL Server. Chú ý rằng package egcs-c++ chỉ được yêu cầu cho các chương trình biên dịch như PostgreSQL và có thể tháo gỡ một cách an toàn sau khi việc biên dịch PostgreSQL thành công.

Công việc cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy PostgreSQL Database server, các tập tin sau được yêu cầu và phải được tạo hay được sao chép vào các thư mục thích hợp trên máy server của bạn:

Sao chép tập tin postgresql vào trong thư mục "/etc/rc.d/init.d".

Cấu hình cho tập tin script "/etc/rc.d/init.d/postgresql"

Cấu hình cho tập tin script "/etc/rc.d/init.d/postgresql" của bạn để khởi động và dừng PostgreSQL Server.

Tạo tập tin script postgresql (touch /etc/rc.d/init.d/postgresql) và thêm vào:

```
#! /bin/sh
# postgresql This is the init script for starting up the PostgreSQL
# server

# chkconfig: 345 85 15
# description: Starts and stops the PostgreSQL backend daemon that handles \
# all database requests.
# processname: postmaster
# pidfile: /var/run/postmaster.pid
    #

# Source function library.
. /etc/rc.d/init.d/functions

# Get config.
    . /etc/sysconfig/network

# Check that networking is up.
# Pretty much need it for postmaster.
[ ${NETWORKING} = "no" ] && exit 0

    [ -f /usr/bin/postmaster ] || exit 0

# This script is slightly unusual in that the name of the daemon (postmaster)
# is not the same as the name of the subsystem (postgresql)

# See how we were called.
case "$1" in
start)
    echo -n "Checking postgresql installation: "
    # Check for the PGDATA structure
    if [ -f /var/lib/pgsql/PG_VERSION ] && [ -d /var/lib/pgsql/base/template1 ]
    then
    # Check version of existing PGDATA

        if [ `cat /var/lib/pgsql/PG_VERSION` != '6.5' ]
        then
            echo "old version. Need to Upgrade."
            echo "See /usr/doc/postgresql-6.5.2/README.rpm for more
    information."
```

```
        exit 1
    else
        echo "looks good!"
    fi

# No existing PGDATA! Initdb it.

else
    echo "no database files found."
    if [ ! -d /var/lib/pgsql ]
    then
        mkdir -p /var/lib/pgsql
        chown postgres.postgres /var/lib/pgsql
    fi
    su -l postgres -c '/usr/bin/initdb --pglib=/usr/lib/pgsql --
pgdata=/var/lib/pgsql'
fi

# Check for postmaster already running...
pid=`pidof postmaster`
if [ $pid ]
then
    echo "Postmaster already running."
else
    #all systems go -- remove any stale lock files
    rm -f /tmp/.s.PGSQL.* > /dev/null
    echo -n "Starting postgresql service: "
    su -l postgres -c '/usr/bin/postmaster -i -S -D/var/lib/pgsql'
    sleep 1
    pid=`pidof postmaster`
    if [ $pid ]
    then
        echo -n "postmaster [$pid]"
        touch /var/lock/subsys/postgresql
        echo $pid > /var/run/postmaster.pid
        echo
    else
        echo "failed."
    fi
fi
```

```
    ;;
stop)
    echo -n "Stopping postgresql service: "
    killproc postmaster
    sleep 2
    rm -f /var/run/postmaster.pid
        rm -f /var/lock/subsys/postgresql
    echo
    ;;
status)
    status postmaster
    ;;
restart)
    $0 stop
    $0 start
    ;;
*)
    echo "Usage: postgresql {start|stop|status|restart}"
    exit 1
esac

    exit 0
```

Và bây giờ, tạo script có thể thực thi được và thay đổi các quyền mặc định của nó:

```
[root@deep /]# chmod 700 /etc/rc.d/init.d/postgresql
```

Tạo các mối liên kết trong rc.d cho PostgreSQL với lệnh:

```
[root@deep /]# chkconfig --add postgresql
```

Khởi động PostgreSQL Server mới thực hiện bằng tay với lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/postgresql start
Checking postgresql installation: looks good!
Starting postgresql service: postmaster [22401]
```

Các lệnh

Các lệnh được liệt kê bên dưới là một số lệnh mà chúng ta thường dùng, nhưng hiện nay cũng đã có nhiều lệnh hơn. Kiểm tra trang man để biết thêm thông tin chi tiết.

- Để định nghĩa một người dùng mới trong cơ sở dữ liệu của bạn, chạy trình tiện ích createuser:

```
[root@deep /]# su postgres
[postgres@deep /]$ createuser
Enter name of user to add ---> admin
Enter user's postgres ID or RETURN to use unix user ID: 500 ->
Is user "admin" allowed to create databases (y/n) y
Is user "admin" a superuser? (y/n) y
createuser: admin was successfully added
```

- Để xoá một người dùng mới trong cơ sở dữ liệu của bạn, chạy trình tiện ích destroyuser:

```
[root@deep /]# su postgres
[postgres@deep /]$ destroyuser
Enter name of user to delete ---> admin
destroyuser: delete of user admin was successful.
```

- Để tạo một cơ sở dữ liệu mới, chạy trình tiện ích createdb:

```
[root@deep /]# su postgres
[postgres@deep /]$ createdb dbname (dbname is the name of the database).

or with the Postgres terminal monitor program (psql)
[root@deep /]# su admin
[admin@deep /]$ psql template1
Welcome to the POSTGRESQL interactive sql monitor:
  Please read the taap tin COPYRIGHT for copyright terms of
POSTGRESQL
[PostgreSQL 6.5.3 on i686-pc-linux-gnu, compiled by egcs ]

  type \? for help on slash commands
  type \q to quit
  type \g or terminate with semicolon to execute query
You are currently connected to the database: template1

template1 → create database foo;
CREATEDB
```

Ghi chú: *Các kết nối client có thể bị hạn chế bởi địa chỉ IP và/hoặc user name thông qua tập tin "pg_hba.conf" trong PG_DATA.*

Một số lệnh của chương trình theo dõi terminal Postgres tiện ích khác là:

- Để kết nối tới cơ sở dữ liệu mới, dùng lệnh:

```
template1 → \c foo
connecting to new database: foo
foo →
```

- Để tạo một bảng(table), dùng lệnh.

```
foo → create table bar (i int4, c char(16));
CREATE
foo →
```

- Để kiểm tra bảng mới, dùng lệnh:

```
foo _ \d bar
Table = bar
+------------------------------+------------------------------+----------+
| Field                        | Type                         | Length   |
+------------------------------+------------------------------+----------+
| i                            | int4                         |    4     |
| c                            | char()                       |   16     |
+------------------------------+------------------------------+----------+
foo →
```

- Để loại bỏ một table, index, view, dùng lệnh:

```
foo → drop table table_name;
foo → drop index index_name;
foo → drop view view_name;
```

- Để chèn vào: (once a table is created, it can be filled using the command...)

```
foo → insert into table_name (name_of_attr1, name_of_attr2, name_of_attr3)
foo → values (value1, value2, value3);
```

Các tập tin cài đặt

```
> /etc/rc.d/init.d/postgresql
> /etc/rc.d/rc0.d/K15postgresql
> /etc/rc.d/rc1.d/K15postgresql
> /etc/rc.d/rc2.d/K15postgresql
> /etc/rc.d/rc3.d/S85postgresql
> /etc/rc.d/rc4.d/S85postgresql
> /etc/rc:d/rc5.d/S85postgresql
> /etc/rc.d/rc6.d/K15postgresql
> /usr/bin/postgres
> /usr/bin/postmaster
> /usr/bin/ecpg
> /usr/bin/pg_id
> /usr/bin/pg_version
> /usr/bin/psql
> /usr/man/manl/begin.l
> /usr/man/manl/close.l
> /usr/man/manl/cluster.l
> /usr/man/manl/commit.l
> /usr/man/manl/copy.l
> /usr/man/manl/create_aggregate.l
> /usr/man/manl/create_database.l
> /usr/man/manl/create_function.l
> /usr/man/manl/create_index.l
> /usr/man/manl/create_language.l
> /usr/man/manl/create_operator.l
> /usr/man/manl/create_rule.l
> /usr/man/manl/create_sequence.l
> /usr/man/manl/create_table.l
```

> /usr/bin/pg_dump
> /usr/bin/pg_dumpall
> /usr/bin/pg_upgrade
> /usr/bin/pg_passwd
> /usr/bin/cleardbdir
> /usr/bin/createdb
> /usr/bin/createlang
> /usr/bin/createuser
> /usr/bin/destroydb
> /usr/bin/destroylang
> /usr/bin/destroyuser
> /usr/bin/initdb
> /usr/bin/vacuumdb
> /usr/bin/initlocation
> /usr/bin/ipcclean
> /usr/include/lrb
> /usr/include/lib/dllist.h
> /usr/include/pgsql
> /usr/include/pgsql/access
> /usr/include/pgsql/access/attnum.h
> /usr/include/pgsql/commands
> /usr/include/pgsql/commands/trigger.h
> /usr/include/pgsql/executor
> /usr/include/pgsql/executor/spi.h
> /usr/include/pgsql/libpq
> /usr/include/pgsql/libpq/pqcomm.h
> /usr/include/pgsql/libpq/libpq-fs.h
> /usr/include/pgsql/libpq++
>/usr/include/pgsql/libpq++/pgconnection.h
> /usr/include/pgsql/libpq++/pgdatabase.h
> /usr/include/pgsql/libpq++/pgtransdb.h
> /usr/include/pgsql/libpq++/pgcursordb.h
> /usr/include/pgsql/libpq++/pglobject.h
> /usr/include/pgsql/port
> /usr/include/pgsql/port/linux
> /usr/include/pgsql/utils
> /usr/include/pgsql/utils/geo_decls.h
> /usr/include/pgsql/utils/elog.h
> /usr/include/pgsql/utils/palloc.h
> /usr/man/manl/create_trigger.l
> /usr/man/manl/create_type.l
> /usr/man/manl/create_user.l
> /usr/man/manl/create_version.l
> /usr/man/manl/create_view.l
> /usr/man/manl/declare.l
> /usr/man/manl/delete.l
> /usr/man/manl/drop.l
> /usr/man/manl/drop_aggregate.l
> /usr/man/manl/drop_database.l
> /usr/man/manl/drop_function.l
> /usr/man/manl/drop_index.l
> /usr/man/manl/drop_language.l
> /usr/man/manl/drop_operator.l
> /usr/man/manl/drop_rule.l
> /usr/man/manl/drop_sequence.l
> /usr/man/manl/drop_table.l
> /usr/man/manl/drop_trigger.l
> /usr/man/manl/drop_type.l
> /usr/man/manl/drop_user.l
> /usr/man/manl/drop_view.l
> /usr/man/manl/end.l
> /usr/man/manl/explain.l
> /usr/man/manl/fetch.l
> /usr/man/manl/grant.l
> /usr/man/manl/insert.l
> /usr/man/manl/listen.l
> /usr/man/manl/load.l
> /usr/man/manl/lock.l
> /usr/man/manl/move.l
> /usr/man/manl/notify.l
> /usr/man/manl/reset.l
> /usr/man/manl/revoke.l
> /usr/man/manl/rollback.l
> /usr/man/manl/select.l
> /usr/man/manl/set.l
> /usr/man/manl/show.l
> /usr/man/manl/sql.l
> /usr/man/manl/update.l

> /usr/include/pgsql/utils/mcxt.h
> /usr/include/pgsql/fmgr.h
> /usr/include/pgsql/os.h
> /usr/include/pgsql/config.h
> /usr/include/pgsql/c.h
> /usr/include/pgsql/postgres.h
> /usr/include/pgsql/postgres_ext.h
> /usr/include/pgsql/libpq-fe.h
> /usr/include/pgsql/libpq-int.h
> /usr/include/pgsql/ecpgerrno.h
> /usr/include/pgsql/ecpglib.h
> /usr/include/pgsql/ecpgtype.h
> /usr/include/pgsql/sqlca.h
> /usr/include/pgsql/libpq++.H
> /usr/lib/libpq.a
> /usr/lib/libpq.so.2.0
> /usr/lib/libpq.so.2
> /usr/lib/libpq.so
> /usr/lib/libecpg.a
> /usr/lib/libecpg.so.3.0.0
> /usr/lib/libecpg.so.3
> /usr/lib/libecpg.so
> /usr/lib/libpq++.a
> /usr/lib/libpq++.so.3.0
> /usr/lib/libpq++.so.3
> /usr/lib/libpq++.so
> /usr/lib/plpgsql.so
> /usr/lib/pgsql
> /usr/lib/pgsql/global1.bki.source
>/usr/lib/pgsql/local1_template1.bki.source
> /usr/lib/pgsql/pg_geqo.sample
> /usr/man/man1/vacuum.1
> /var/lib/pgsql
> /var/lib/pgsql/base
> /var/lib/pgsql/base/template1
> /var/lib/pgsql/base/template1/pg_proc
> /var/lib/pgsql/base/template1/pg_type
> /var/lib/pgsql/base/template1/pg_attribute
> /var/lib/pgsql/base/template1/pg_class
> /var/lib/pgsql/base/template1/pg_inherits
> /var/lib/pgsql/base/template1/pg_index
> /var/lib/pgsql/base/template1/pg_statistic
> /var/lib/pgsql/base/template1/pg_operator
> /var/lib/pgsql/base/template1/pg_opclass
> /var/lib/pgsql/base/template1/pg_am
> /var/lib/pgsql/base/template1/pg_amop
> /var/lib/pgsql/base/template1/pg_amproc
> /var/lib/pgsql/base/template1/pg_language
> /var/lib/pgsql/base/template1/pg_aggregate
> /var/lib/pgsql/base/template1/pg_ipl
> /var/lib/pgsql/base/template1/pg_inheritproc
> /var/lib/pgsql/base/template1/pg_rewrite
> /var/lib/pgsql/base/template1/pg_listener
> /var/lib/pgsql/base/template1/pg_description
> /var/lib/pgsql/base/template1/
pg_attribute_relid_attnam_index
> /var/lib/pgsql/base/template1/
pg_attribute_relid_attnum_index
> /var/lib/pgsql/base/template1/
pg_attribute_attrelid_index
> /var/lib/pgsql/base/template1/
pg_proc_oid_index
>/var/lib/pgsql/base/template1/
pg_proc_proname_narg_type_index
> /var/lib/pgsql/base/template1/
pg_proc_prosrc_index
>/var/lib/pgsql/base/template1/
pg_type_oid_index
> /var/lib/pgsql/base/template1/
pg_type_typname_index

> /usr/lib/pgsql/pg_hba.conf.sample

> /usr/man/man1/cleardbdir.1

> /usr/man/man1/createdb.1

> /usr/man/man1/createuser.1

> /usr/man/man1/destroydb.1

> /usr/man/man1/destroyuser.1

> /usr/man/man1/ecpg.1

> /usr/man/man1/initdb.1

> /usr/man/man1/initlocation.1

> /usr/man/man1/ipcclean.1

> /usr/man/man1/pg_dump.1

> /usr/man/man1/pg_dumpall.1

> /usr/man/man1/pg_passwd.1

> /usr/man/man1/pg_upgrade.1

> /usr/man/man1/postgres.1

> /usr/man/man1/postmaster.1

> /usr/man/man1/psql.1

> /usr/man/man3/catalogs.3

> /usr/man/man3/libpq.3

> /usr/man/man5/pg_hba.conf.5

> /usr/man/man1

> /usr/man/man1/abort.1

> /usr/man/man1/alter_table.1

> /usr/man/man1/alter_user.1

> /var/lib/pgsql/base/template1/pg_class_oid_index

> /var/lib/pgsql/base/template1/pg_class_relname_index

> /var/lib/pgsql/base/template1/pg_attrdef

> /var/lib/pgsql/base/template1/pg_attrdef_adrelid_index

> /var/lib/pgsql/base/template1/pg_relcheck

> /var/lib/pgsql/base/template1/pg_relcheck_rcrelid_index

> /var/lib/pgsql/base/template1/pg_trigger

> /var/lib/pgsql/base/template1/pg_trigger_tgrelid_index

> /var/lib/pgsql/base/template1/pg_description_objoid_index

> /var/lib/pgsql/base/template1/PG_VERSION

> /var/lib/pgsql/base/template1/pg_user

> /var/lib/pgsql/base/template1/pg_rules

> /var/lib/pgsql/base/template1/pg_views

> /var/lib/pgsql/base/template1/pg_tables

> /var/lib/pgsql/base/template1/pg_indexes

> /var/lib/pgsql/pg_variable

> /var/lib/pgsql/pg_database

> /var/lib/pgsql/pg_shadow

> /var/lib/pgsql/pg_group

> /var/lib/pgsql/pg_log

> /var/lib/pgsql/PG_VERSION

> /var/lib/pgsql/pg_hba.conf

> /var/lib/pgsql/pg_geqo.sample

> /var/lib/pgsql/pg_pwd

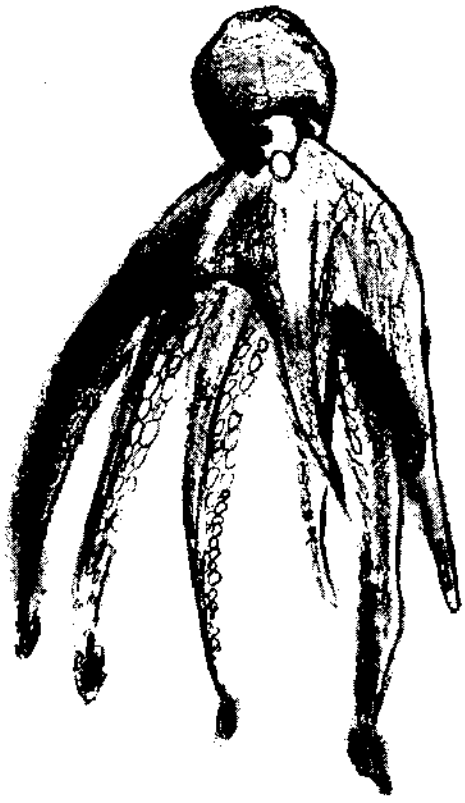
Chương 18:

PHẦN MỀM SERVER

(Dịch vụ Proxy)

Trong chương này chúng ta sẽ đề cập đến các vấn đề sau:

- Linux Squid Proxy Server.
  - Sử dụng thư viện GNU malloc để cải tiến khả năng thực hiện cache của Squid.
  - Sự cấu hình.
  - Bảo mật Squid.
  - Tối ưu Squid.
  - Trình tiện ích cachemgr.cgi của Squid.
  - Cấu hình proxy cho Netscape để dùng với Squid.



Linux Squid Proxy Server

Tổng quan:

Squid là một proxy server, khả năng của squid là tiết kiệm băng thông (bandwidth), cải tiến việc bảo mật, tăng tốc độ truy cập web cho người sử dụng và trở thành một trong những proxy phổ biến được nhiều người biết đến. Tại thời điểm này chỉ có một vài chương trình proxy-server trên thị trường. Với mục đích tiết kiệm chi phí nhưng vẫn đáp ứng được yêu cầu truy cập Internet, các proxy server này có hai mặt hạn chế chính là: chúng là những sản phẩm thương mại tức là chúng ta phải trả tiền khi sử dụng những sản phẩm này. Điểm hạn chế quan trọng hơn là chúng không hỗ trợ ICP (ICP được sử dụng để cập nhật những thay đổi về nội dung của những URL sẵn có trong cache (cache là nơi lưu trữ những trang web mà bạn đã từng đi qua). Squid là sự lựa chọn tốt nhất cho một proxy-cache server, squid đáp ứng hai yêu cầu của chúng ta là sử dụng miễn phí và có thể sử dụng đặc trưng ICP.

Xuất phát từ phần mềm "cached" của dự án nghiên cứu của ARPA-funded được phát triển ở National Laboratory cho việc nghiên cứu network và được tài trợ bởi National Science Foundation, squid đưa ra kỹ thuật lưu giữ ở cấp độ cao của các web client, đồng thời hỗ trợ các dịch vụ thông thường như FTP, Gopher và HTTP. Squid lưu giữ những thông tin mới nhất của các dịch vụ trên trong RAM, quản lý một cơ sở dữ liệu lớn của các thông tin trên đĩa, có một kỹ thuật điều khiển truy cập phức tạp, hỗ trợ giao thức SSL cho các kết nối bảo mật thông qua proxy. Hơn nữa squid có thể liên kết với các cache của các proxy server khác trong việc sắp xếp lưu trữ các trang web một cách hợp lý.

Trong việc biên dịch và cấu hình của squid, chúng ta sẽ định hình squid chạy như một web server tăng tốc (httpd-accelerator) để nhận nhiều khả năng thực thi bên ngoài web server của chúng ta. Trong chế độ tăng tốc (accelerator mode), squid hành động như một reverse proxy cache, nó chấp nhận các yêu cầu từ client, xử lý các yêu cầu này trên cache của squid proxy khác hoặc gởi yêu cầu này đến chính nơi mà client yêu cầu. Cũng trong chương này chúng ta sẽ biết làm thế nào để cấu hình Squid như một proxy-caching server có thể cho phép tất cả các user trong mạng sử dụng Squid để truy cập Internet.

Hướng dẫn cài đặt

- Các lệnh thì tương thích với unix.
- Đường dẫn nguồn là /var/tmp (các đường dẫn khác cũng có thể được).
- Sự cài đặt đã được kiểm tra trên Red Hat Linux 6.1 và 6.2.
- Tất cả các bước cài đặt sẽ được thực hiện với supper-user "root".
- Phiên bản Squid là 2.3.STABLE2.

Các gói (packages)

Bạn có thể nhận những package hay thông tin về Squid theo địa chỉ sau:

Squid Homepage: http://www.squid-cache.org/

Squid FTP Site: 204.144.128.89

Bạn cũng đảm bảo rằng đã download được file: squid-2.3.STABLE2-src.tar.gz

Dạng lưu trữ Tarballs

Trước hết hãy tạo ra danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt Squid, sau đó so sánh chúng dùng lệnh "diff" để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh "find /* > Squid1" trước và "find /* > Squid2" sau khi bạn cài đặt phần mềm, và sử dụng lệnh "diff Squid1 Squid2 > Squid-Installed" để lấy một danh sách các thay đổi nào đó.

Sự biên dịch

Giải nén tập tin dạng tarball (tar.gz)

```
[root@deep /]# cp squid-version.STABLEz-src.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf squid-version.STABLEz-src.tar.gz
```

Cấu hình và tối ưu

Bước 1:

Squid không thể chạy như super-user root và do lý do này chúng ta phải tạo một user đặc biệt không có shell để chạy Squid Proxy Server

```
[root@deep /]# useradd -d /cache/ -r -s /dev/null squid >/dev/null 2>&1
[root@deep /]# mkdir /cache/
[root@deep /]# chown -R squid.squid /cache/
```

Đầu tiên chúng ta tạo user squid trong tập tin /etc/passwd, sau đó chúng ta tạo thư mục "/cache" nếu thư mục này không có sẵn (việc này chỉ làm nếu /cache không có sẵn). Cuối cùng chúng ta thay đổi quyền sở hữu thư mục cache cho user squid.

Chú ý: *Thường thì chúng ta không cần thực hiện lệnh (mkdir /cache) bởi vì chúng ta đã tạo thư mục này khi phân chia đĩa cứng trong quá trình cài đặt Linux (nếu bạn theo chỉ dẫn cài đặt trong cuốn sách này). Nếu sự phân chia của bạn không tạo thư mục /cache bạn phải thực thi lệnh tạo thư mục /cache trên.*

Bước 2:

Chuyển tới thư mục mới của Squid và gõ vào những lệnh sau trên terminal của bạn.

Hiệu chỉnh tập tin Makefile.in (vi +18 icons/Makefile.in) và thay đổi dòng:

```
DEFAULT_ICON_DIR = $(sysconfdir)/icons
```

Trở thành:

DEFAULT_ICON_DIR = $(libexecdir)/icons

Chúng ta đã thay đổi biến (sysconfdir) thành (libexecdir). Với sự thay đổi này thư mục "icons" của Squid sẽ được đặt dưới thư mục "/usr/lib/squid".

Hiệu chỉnh tập tin Makefile.in (vi +34 src/Makefile.in) và thay đổi các dòng:

DEFAULT_CACHE_LOG = $(localstatedir)/logs/cache.log

Trở thành:

DEFAULT_CACHE_LOG = $(localstatedir)/log/squid/cache.log

DEFAULT_ACCESS_LOG = $(localstatedir)/logs/access.log

Trở thành:

DEFAULT_ACCESS_LOG = $(localstatedir)/log/squid/access.log

DEFAULT_STORE_LOG = $(localstatedir)/logs/store.log

Trở thành:

DEFAULT_STORE_LOG = $(localstatedir)/log/squid/store.log

DEFAULT_PID_FILE = $(localstatedir)/logs/squid.pid

Trở thành:

DEFAULT_PID_FILE = $(localstatedir)/run/squid.pid

DEFAULT_SWAP_DIR = $(localstatedir)/cache

Trở thành:

DEFAULT_SWAP_DIR = /cache

DEFAULT_ICON_DIR = $(sysconfdir)/icons

Trở thành:

DEFAULT_ICON_DIR = $(libexecdir)/icons

Chúng ta đã thay đổi vị trí mặc định của "cache.log", "access.log", "store.log" tới vị trí đặt dưới thư mục "/var/log/squid". Sau đó đặt tập tin pid của squid dưới thư mục "/var/run" và cuối cùng tạo thư mục "icons" dưới thư mục "/usr/lib/squid/icons" với biến libexecdir ở trên.

Sử dụng thư viện GNU malloc để cải tiến khả năng thực hiện cache của Squid

Nếu bạn đang khổ sở với sự giới hạn về bộ nhớ trên hệ thống của bạn, khả năng thực thi cache của Squid sẽ bị ảnh hưởng. Để giảm bớt vấn đề này bạn có thể kết nối Squid với một thư viện malloc bên ngoài chẳng hạn như GNU malloc. Để

cho Squid sử dụng GNU malloc như một thư viện bên ngoài theo những bước đơn giản sau:

Gói dữ liệu (packages)

GNU malloc Homepage: http://www.gnu.org/order/ftp.html

Bạn phải bảo đảm rằng đã tải về file: malloc.tar.gz

```
[root@deep /]# cp malloc.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf malloc.tar.gz
```

Bước 1:

Biên dịch và cài đặt GNU malloc trên hệ thống của bạn bằng cách thực thi những lệnh sau:

```
[root@deep tmp]# cd malloc
[root@deep malloc]# export CC=egcs
[root@deep malloc]# make
```

Bước 2:

Copy tập tin "libmalloc.a" vào thư mục thư viện hệ thống và bảo đảm nó với tên "libgnumalloc.a"

```
[root@deep malloc]# cp libmalloc.a /usr/lib/libgnumalloc.a
```

Bước 3

Copy tập tin "malloc.h" tới thư mục include của hệ thống và đảm bảo nó với tên "gnumalloc.h"

```
[root@deep malloc]# cp malloc.h /usr/include/gnumalloc.h
```

Với các tập tin "libgnumalloc.a" và "gnumalloc.h" được cài đặt trên hệ thống, Squid sẽ nhận biết một cách tự động trong thời gian biên dịch và sẽ sử dụng chúng để cải tiến khả năng thực thi cache của nó.

Biên dịch và tối ưu

Bước 1:

Trở lại thư mục mới của Squid và gõ vào những lệnh sau:

```
CC="egcs" \
CFLAGS="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomit-frame-
pointer -fno-exceptions" \
./configure \
--prefix=/usr \
--exec-prefix=/usr \
--bindir=/usr/sbin \
--libexecdir=/usr/lib/squid \
```

```
--localstatedir=/var \
--sysconfdir=/etc/squid \
--enable-delay-pools }_
--enable-cache-digests \
--enable-poll \
--disable-ident-lookups \
--enable-truncate \
--enable-heap-replacement
```

Thiết đặt Squid cho những phần cứng quen thuộc với những tùy chọn như sau:

- Sử dụng đặc trưng làm trễ của Squid để giới hạn và điều khiển khoảng băng thông sử dụng cho các người dùng.
- Sử dụng việc lưu giữ có sắp xếp (Cache digest) để cải tiến thời gian trả lời và sự tận dụng mạng của client.
- Cho phép hàm poll() thay cho select().
- Làm mất tác dụng khả năng tìm kiếm nhận biết (ident-lookups) để xoá bỏ mã chương trình cái mà thực hiện việc ident-lookups (RFC 931) và giảm bớt sự phá hoại có thể có theo kiểu từ chối dịch vụ (denial of service).
- Cho phép bỏ bớt tập tin dư thừa trong một vài sự cải tiến thực hiện khi đang loại bỏ những tập tin cache.
- Sử dụng những đặt trưng thay thế của squid để chọn những thuật toán thay thế khác nhau thay cho thuật toán chuẩn LRU nhằm để tăng thêm khả năng thực hiện. Xem phần dưới cho nhiều sự diễn giải trong chi tiết.

Bước 2:

Bây giờ chúng ta biên dịch và cài đặt Squid trên server

```
[root@deep squid-2.3.STABLE2]# make -f makefile
[root@deep squid-2.3.STABLE2]# make install
[root@deep squid-2.3.STABLE2]# mkdir -p /var/log/squid
[root@deep squid-2.3.STABLE2]# rm -rf /var/logs/
[root@deep squid-2.3.STABLE2]# chown squid.squid /var/log/squid/
[root@deep squid-2.3.STABLE2]# chmod 750 /var/log/squid/
[root@deep squid-2.3.STABLE2]# chmod 750 /cache/
[root@deep squid-2.3.STABLE2]# rm -f /usr/sbin/RunCache
[root@deep squid-2.3.STABLE2]# rm -f /usr/sbin/RunAccel
[root@deep squid-2.3.STABLE2]# strip /usr/sbin/squid
[root@deep squid-2.3.STABLE2]# strip /usr/sbin/client
[root@deep squid-2.3.STABLE2]# strip /usr/lib/squid/dnsserver
[root@deep squid-2.3.STABLE2]# strip /usr/lib/squid/unlinkd
```

```
[root@deep squid-2.3.STABLE2]# strip /usr/lib/squid/cachemgr.cgi
```

Lệnh "make –f" sẽ biên dịch tất cả các tập tin nguồn thành các tập tin nhị phân có thể thi hành được, và "make install" sẽ cài đặt những tập tin nhị phân và những tập tin hỗ trợ cho chúng vào những nơi thích hợp. Lệnh "mkdir" sẽ tạo một thư mục mới với tên là "squid" trong "/var/log". Lệnh "rm –rf" sẽ xoá bỏ thư mục /var/logs vì thư mục này đã tạo để giải quyết những tập tin log liên quan tới Squid mà chúng ta đã chuyển tới thư mục "/var/log/squid". Lệnh "chown" sẽ thay đổi chủ sở hữu của "/var/log/squid" tới user squid và lệnh chmod sẽ thay đổi mode của các thư mục "squid" và "cache" trở thành (0750/drwxr-x---) cho nguyên nhân bảo mật.

Nên nhớ rằng chúng ta loại bỏ những tập tin script nhỏ tên "RunCache" và "RunAccel" cái mà khởi động Squid hoặc trong mode caching hoặc mode accelerater, bởi vì chúng ta sử dụng script tốt hơn tên là squid được đặt trong thư mục "/etc/rc.d/init.d" theo đúng kiến trúc của Linux. Lệnh "strip" sẽ giảm bớt kích thước của các tập tin nhị phân để tăng khả năng thực thi.

Dọn dẹp sau khi cài đặt

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf squid-version/ squid-version.STABLEz-src.tar.gz
[root@deep tmp]# rm -rf malloc/ malloc.tar.gz (if you used the GNU malloc
external library)
```

Lệnh "rm" sẽ xoá tất cả các tập tin nguồn chúng ta sử dụng để biên dịch và cài đặt Squid và GNU malloc. Nó cũng sẽ xoá bỏ tập tin dạng nén Squid và GNU malloc từ thư mục "/var/tmp".

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy Squid server trong mode httpd-accelerator, những tập tin sau đây được yêu cầu và phải được tạo hoặc được copy tới những thư mục thích hợp trên server của bạn.

  Sao chép tập tin squid.conf vào trong thư mục "/etc/squid/".

  Sao chép tập tin script squid vào trong thư mục "/etc/rc.d/init.d/".

  Sao chép tập tin squid vào trong thư mục "/etc/logrotate.d/".

- Để chạy Squid server trong mode proxy-chaching, những tập tin sau đây được yêu cầu và phải được tạo hoặc được copy tới những thư mục thích hợp trên server của bạn.

  Sao chép tập tin squid.conf vào trong thư mục "/etc/squid/".

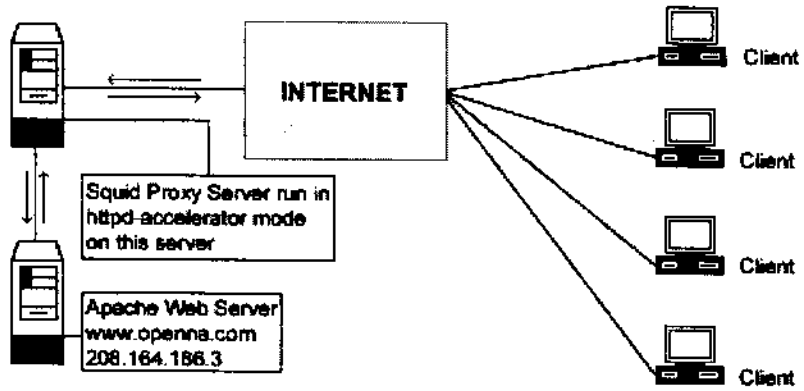
  Sao chép tập tin script squid vào trong thư mục "/etc/rc.d/init.d/".

  Sao chép tập tin squid vào trong thư mục "/etc/logrotate.d/".

Cấu hình tập tin "/etc/squid/squid.conf" làm việc với mode httpd-accelerator

Tập tin "squid.conf" được sử dụng để thiết đặt và cấu hình tất cả các tùy chọn khác nhau cho Squid proxy server của bạn. Trong tập tin cấu hình dưới đây chúng ta sẽ cấu hình tập tin "/etc/squid/squid.conf" trong mode httpd-accelerator. Trong mode accelerator này nếu cả hai web server và Squid chạy trên cùng một máy thì bạn phải thiết đặt chương trình daemon của nó chạy trên port 81. Với Apache Web server bạn có thể làm điều này bằng cách thay đổi trên dòng Port 80 thành Port 81 trong tập tin cấu hình httpd.conf. Nếu Web server chạy trên các server khác trong network của bạn, bạn có thể giữ nguyên port 80 cho Apache vì Squid sẽ chạy trên một máy có IP khác, ở đó port 80 đã không sử dụng.

Xem hình vẽ sau:



Hiệu chỉnh tập tin squid.conf (vi /etc/squid/squid.conf) và thêm hoặc thay đổi những tùy chọn như sau:

```
http_port 8080
icp_port 0
acl QUERY urlpath_regex cgi-bin \?
no_cache deny QUERY
cache_mem 16 MB
cache_dir ufs /cache 200 16 256
emulate_httpd_log on
redirect_rewrites_host_header off
replacement_policy GDSF
acl all src 0.0.0.0/0.0.0.0
http_access allow all
cache_mgr admin@openna.com
cache_effective_user squid
```

```
cache_effective_group squid
httpd_accel_host 208.164.186.3
httpd_accel_port 80
log_icp_queries off
cachemgr_passwd my-secret-pass all
buffered_logs on
```

Việc này sẽ bảo mật tập tin squid.conf thiết đặt chính nó cho cấu hình quen thuộc với những tùy chọn:

http_port 8080

Tùy chọn http_port chỉ định port mà Squid sẽ lắng nghe những yêu cầu HTTP từ client. Nếu bạn thiết đặt tùy chọn này với port 80, client sẽ có ảo tưởng rằng được kết nối tới Apache web server. Bởi vì chúng ta đang chạy Squid trong mode accelerator chúng ta phải listen trên port 80.

icp_port 0

Tùy chọn icp_port chỉ định port ở đó Squid sẽ gởi và nhận các yêu cầu ICP từ những cache láng giềng. Bạn phải đặt giá trị của tùy chọn này tới 0 để làm mất tác dụng chúng. Vì chúng ta đang cấu hình squid trong accelerator mode cho Web server. Đặc trưng ICP chỉ cần thiết trong một môi trường cache ở nhiều cấp với cache cha (parent cache) và cache anh em (sibling cache). Sử dụng ICP trong cấu hình accelerator mode có thể thêm tổng phí không mong muốn cho squid.

acl QUERY urlpath_regex cgi-bin \? and no_cache deny QUERY

Tùy chọn "acl QUERY urlpath_regex cgi-bin \? and no_cache deny QUERY" được dùng để bắt buộc các object không bao giờ được cache, như những tập tin dưới thư mục cgi-bin. Đây là đặc trưng bảo mật.

cache_mem 16 MB

Tùy chọn này chỉ rõ tổng số bộ nhớ RAM được sử dụng cho cache như được gọi là In-Transit objects, Hot Objects, Negative-Cached objects. Đây là một đặc trưng tối ưu hoá. Có một điều quan trọng cần thông báo rằng Squid có thể sử dụng nhiều hơn giá trị bạn chỉ rõ trong tham số này và vì lý do này, nếu bạn có 48MB dành cho Squid bạn phải đặt 48/3=16MB ở đây.

cache_dir ufs /cache 200 16 256

Tùy chọn "cache_dir" chỉ rõ trong thứ tự: loại gì của hệ thống lưu trữ sử dụng (ufs), tên của thư mục /cache cho Squid, khoảng trống đĩa tính trên megabyte sử dụng dưới thư mục này (200MB), số các thư mục con mức thứ nhất được tạo dưới thư mục cache (16 Level 1), và số thư mục con mức 2 được tạo dưới mỗi thư mục cache mức thứ nhất (256 Level 2). Trong mode accelerator, tùy chọn này liên hệ trực tiếp đến kích thước và số tập tin mà bạn muốn đáp ứng với Apache web server.

emulate_httpd_log on

Tùy chọn "emulate_httpd_log" nếu đặt là ON, sẽ chỉ rõ rằng Squid sẽ mô phỏng định dạng logfilecủa Apache Web server. Việc này rất có ích nếu bạn muốn dùng một chương trình của hãng thứ ba giống như Webalizer để phân tích logfileWeb server.

replacement_policy GDSF

Tùy chọn replacement_policy chỉ rõ chính sách cache mà Squid sẽ dùng để chỉ định object nào trong cache phải được thay thế khi proxy cần làm nhiều khoảng trống đĩa. Chính sách Squid LRU được sử dụng bởi mặc định nếu bạn không chỉ rõ tùy chọn "--enable-heap-replacement" trong thời gian biên dịch. Trong tập tin cấu hình chúng ta chọn GDSF (Greedy-Dual Size Frequency) như chính sách mặc định.

acl all src 0.0.0.0/0.0.0.0 and http_access allow all

Tùy chọn "acl" và "http_access" chỉ rõ và định nghĩa một danh sách điều khiển truy cập được áp dụng trên proxy server Squid. Các tùy chọn "acl" và "http_access" của chúng ta không bị hạn chế và cho phép mọi kết nối trên proxy server từ khi chúng ta sử dụng proxy này để làm nhanh thêm Web server Apache. Xem sự chỉ dẫn Squid cho nhiều thông tin khi dùng Squid trong mode non-accelerate.

cache_mgr admin

Tùy chọn "cache_mgr" chỉ rõ địa chỉ mail của người quản trị chịu trách nhiệm cho Squid proxy server. Người này là một trong những người nhận mail nếu Squid xuất hiện vấn đề. Bạn có thể chỉ rõ tên hoặc hoặc địa chỉ e-mail trong tùy chọn này.

cache_effective_user squid and cache_effective_group squid

Tùy chọn "cache_effective_user" và "cache_effective_group" chỉ rõ UID/GID mà cache sẽ chạy trên đó. Đừng bao giờ quên chạy Squid với quyền root. Trong tập tin cấu hình này chúng ta sử dụng squid cho cả hai UID và GID.

httpd_accel_host 208.164.186.3 and httpd_accel_port 80

Tùy chọn httpd_accel_host và httpd_accel_port chỉ rõ cho squid địa chỉ IP và port ở đâu là server HTTP thật. Trong cấu hình của chúng ta HTTP Web server là máy có địa chỉ 208.164.186.3 (www.openna.com) và chạy trên port 80. Trong khi Squid proxy server là một máy khác trong network và bởi vì Squid proxy server không chạy trên cùng máy của Web server Apache nên chúng ta có thể dùng port 80 cho Squid proxy server của chúng ta và port 80 cho Web server Apache mà không bị xung đột.

log_icp_queries off

Tùy chọn log_icp_queries chỉ rõ nếu bạn muốn ICP (ICP được dùng để thay đổi dấu vết về sự tồn tại của các URL trong các cache láng giềng) truy vấn tập tin access.log hoặc không. Trước đây chúng ta không dùng đặc trưng ICP trong mode accelerator, chúng ta có thể an toàn đặt tùy chọn này tới "OFF".

cachemgr_passwd my-secret-pass all

Tùy chọn "cachemgr_passwd" chỉ rõ mật khẩu sẽ được yêu cầu cho việc truy cập sự hoạt động của trình tiện ích "cachemgr.cgi". Trình tiện ích CGI này được thiết kế để chạy thông qua giao diện web và thống kê khả năng thực thi và cấu hình Squid. Giá trị của <my-secret-pass> là password bạn chọn và từ khoá <all> chỉ rõ là đặt password này cho tất cả các hành động bạn có thể thực hiện với chương trình này. Xem trình tiện ích cachemgr.cgi của squid phía dưới trong chương này, nó sẽ cho nhiều thông tin.

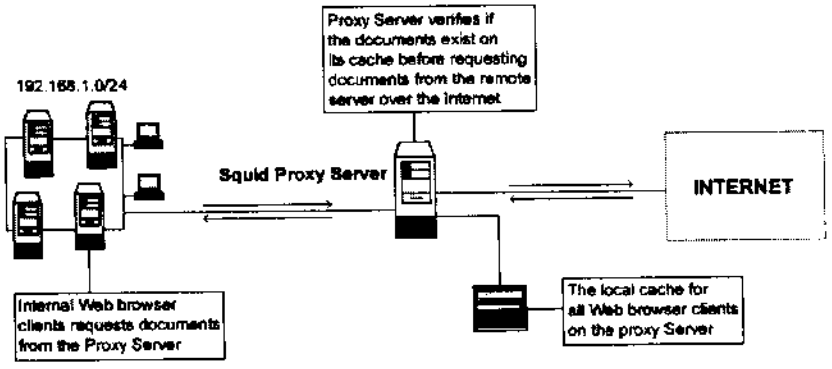
buffered_logs on

Tùy chọn buffered_logs, nếu được mở có thể tăng tốc việc ghi của một số log file.

Cấu hình của tập tin "/etc/squid/squid.conf" như một proxy-caching mode

Với một vài sự thay đổi trong tập tin squid.conf chúng ta đã định nghĩa ở trên để chạy trong httpd-accelerator mode, chúng ta có thể chạy Squid như một proxy-caching server. Với một proxy cache server thì tất cả các người sử dụng bên trong mạng sử dụng Squid để truy cập Internet. Với cấu hình này bạn có thể điều khiển, thêm một vài chính sách đặc biệt trên những gì bạn có thể xem, truy cập và download. Bạn cũng có thể điều khiển băng thông sử dụng, thời gian kết nối và một vài thứ khác nữa. Một proxy cache server có thể được cấu hình chạy như một stand-alone server trong network của bạn hoặc sử dụng và chia sẻ cache với những proxy server khác trên thế giới.

Với ví dụ thứ nhất phía dưới sẽ trình bày cho bạn làm thế nào cấu hình Squid như một stand-alone server và sau đó nói một ít về cấu hình cache có thứ bậc. Ở đây có hai hoặc nhiều proxy proxy-cache server hợp lại cung cấp thông tin cho mỗi proxy khác.



Hiệu chỉnh tập tin squid.conf (vi /etc/squid/squid.conf) thêm hoặc thay đổi những tùy chọn như sau cho proxy cache để chạy như một stand-alone server:

```
http_port 8080
icp_port 0
acl QUERY urlpath_regex cgi-bin \?
no_cache deny QUERY
cache_mem 16 MB
cache_dir ufs /cache 200 16 256
redirect_rewrites_host_header off
replacement_policy GDSF
acl localnet src 192.168.1.0/255.255.255.0
acl localhost src 127.0.0.1/255.255.255.255
acl Safe_ports port 80 443 210 119 70 21 1025-65535
acl CONNECT method CONNECT
acl all src 0.0.0.0/0.0.0.0
http_access allow localnet
http_access allow localhost
http_access deny !Safe_ports
http_access deny CONNECT
http_access deny all
cache_mgr admin@openna.com
cache_effective_user squid
cache_effective_group squid
log_icp_queries off
cachemgr_passwd my-secret-pass all
buffered_logs on
```

Điểm khác lớn so với sự cấu hình httpd-accellerator là sử dụng access control lists (ACL). Đặc trưng này cho phép bạn giới hạn việc truy cập trên cơ sở địa chỉ IP nguồn (source IP address) thường ký hiệu src, địa chỉ IP đích (destination IP) (dst), domain đích, thời gian ..... Nhiều kiểu có sẵn với đặc trưng này và bạn nên tham khảo tập tin squid.conf cho một danh sách đầy đủ. Bốn kiểu được dùng phổ biến nhất là:

```
acl   name     type      data
 |      |        |         |
acl some-name src a.b.c.d/e.f.g.h   # ACL restrict access based on source IP address
acl some-name dst a.b.c.d/e.f.g.h   # ACL restrict access based on destination IP address
acl some-name srcdomain foo.com # ACL restrict access based on source domain
acl some-name dstdomain foo.com # ACL restrict access based on destination domain
```

Ví dụ như trường hợp giới hạn truy cập tới Squid proxy server chỉ với các client bên trong mạng và với một số port được cho phép. Công việc được làm như sau:

```
acl localnet src 192.168.1.0/255.255.255.0
acl localhost src 127.0.0.1/255.255.255.255
acl Safe_ports port 80 443 210 119 70 21 1025-65535
acl CONNECT method CONNECT
acl all src 0.0.0.0/0.0.0.0
http_access allow localnet
http_access allow localhost
```

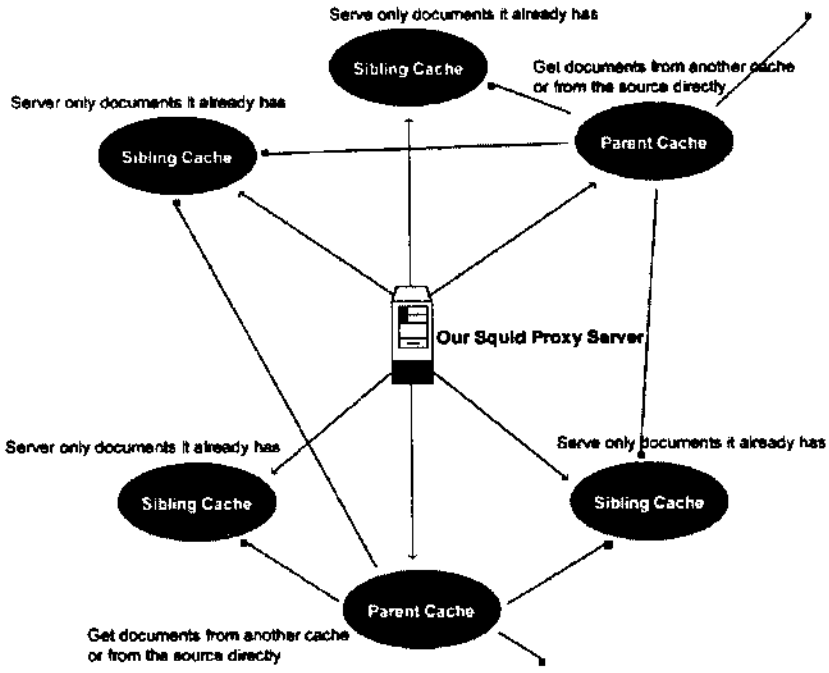
```
http_access deny !Safe_ports
http_access deny CONNECT
http_access deny all
```

Cấu hình acl này sẽ cho phép tất cả các client bên trong mạng với khoảng địa chỉ thuộc class C 192.168.1.0 được truy cập tới proxy server. Nó cũng thông báo rằng bạn cho phép localhost IP (một địa chỉ IP đặt biệt được dùng bởi chính server của bạn) truy cập tới proxy. Sau khi chọn một số port (80=http, 443=https, 210=wais, 119=nntp, 70=gopher, và 21=ftp) cho phép các client bên trong có thể dùng để truy cập Internet, chúng ta từ chối các phương pháp CONNECT để ngăn chặn những người bên ngoài cố gắng kết nối tới proxy server và cuối cùng từ chối tất cả địa chỉ IP nguồn và port trên proxy server.

Lưu giữ Web nhiều cấp (Multi-level Web Caching)

Phương pháp thứ hai của proxy cache được gọi là Multi-level Web Caching ở đây bạn chia sẻ và hợp tác với nhiều proxy-cache server khác trên Internet. Với phương pháp này công ty của bạn sử dụng cache của nhiều proxy cache server khác và ngược lại những proxy cache server khác sử dụng lại cache của bạn. Điều quan trọng là trong trường hợp này proxy cache có thể làm việc trên hai qui tắc khác nhau trong hệ thống cấp bậc của cache. Nó có thể được cấu hình như một sibling cache và chỉ có thể cung cấp những thông tin mà nó sẵn có hoặc nó có thể được cấu hình như một parent cache và có thể gởi thông tin từ những cache khác hoặc trực tiếp từ nguồn.

Xem hình vẽ bên dưới:



Có một điều chúng ta cần lưu ý rằng một kế hoạch tốt để tránh sự phát sinh nhiều lưu thông network hơn khi không có web caching là chọn có vài sibling cache và chỉ có một số nhỏ của các parent cache.

Cấu hình tập tin script/etc/squid/squid.conf cho tất cả các loại cấu hình

Tập tin "/etc/rc.d/init.d/squid" của chúng ta để start hoặc stop chương trình Squid. Tập tin script này đã thay đổi thư mục cache là /cache thay cho "/var/spool/squid".

Tạo tập tin script squid (touch /etc/rc.d/init.d/squid) và thêm vào những thông tin như sau:

```
#!/bin/bash
# squid This shell script takes care of starting and stopping
# Squid Internet Object Cache
#
# chkconfig: - 90 25
# description: Squid - Internet Object Cache. Internet object caching is \
# a way to store requested Internet objects (i.e., data available \
# via the HTTP, FTP, and gopher protocols) on a system closer to the \
# requesting site than to the source. Web browsers can then use the \
# local Squid cache as a proxy HTTP server, reducing access time as \
# well as bandwidth consumption.
# pidfile: /var/run/squid.pid
# config: /etc/squid/squid.conf
PATH=/usr/bin:/sbin:/bin:/usr/sbin
export PATH

# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0

# check if the squid conf taäp tin is present
[ -f /etc/squid/squid.conf ] || exit 0

# determine the name of the squid binary
```

```
[ -f /usr/sbin/squid ] && SQUID=squid
[ -z "$SQUID" ] && exit 0

# determine which one is the cache_swap directory
CACHE_SWAP=`sed -e 's/#.*//g' /etc/squid/squid.conf | \
      grep cache_dir | sed -e 's/cache_dir//' | \
      cut -d ' ' -f 2`
[ -z "$CACHE_SWAP" ] && CACHE_SWAP=/cache

# default squid options
# -D disables initial dns checks. If you most likely will not to have an
# internet connection when you start squid, uncomment this
#SQUID_OPTS="-D"

RETVAL=0
case "$1" in
start)
    echo -n "Starting $SQUID: "
    for adir in $CACHE_SWAP; do
        if [ ! -d $adir/00 ]; then
            echo -n "init_cache_dir $adir... "
            $SQUID -z -F 2>/dev/null
        fi
    done
    $SQUID $SQUID_OPTS &
    RETVAL=$?
    echo $SQUID
    [ $RETVAL -eq 0 ] && touch /var/lock/subsys/$SQUID
    ;;

stop)
    echo -n "Stopping $SQUID: "
    $SQUID -k shutdown &
    RETVAL=$?
    if [ $RETVAL -eq 0 ] ; then
        rm -f /var/lock/subsys/$SQUID
        while : ; do
            [ -f /var/run/squid.pid ] || break
            sleep 2 && echo -n "."
        done
        echo "done"
```

```
    else
        echo
    fi
    ;;
reload)
    $SQUID $SQUID_OPTS -k reconfigure
    exit $?
    ;;

restart)
    $0 stop
    $0 start
    ;;

status)
    status $SQUID
    $SQUID -k check
    exit $?
    ;;

probe)
    exit 0;
    ;;

*)
    echo "Usage: $0 {start|stop|status|reload|restart}"
    exit 1
esac

exit $RETVAL
```

Bây giờ chúng ta làm cho tập tin này có thể thực thi được và thay đổi quyền mặc định của chúng:

```
[root@deep /]# chmod 700 /etc/rc.d/init.d/squid
```

Tạo một biểu tượng liên kết rc.d cho Squid với lệnh:

```
[root@deep /]# chkconfig --add squid
```

Bởi mặc định script squid sẽ không tự động khởi động proxy server trên Red Hat Linux khi bạn khởi động lại server. Bạn có thể thay đổi mặc định của nó bằng cách thực thi những lệnh sau:

```
[root@deep /]# chkconfig --level 345 squid on
```

Khởi động Squid proxy server mới bằng tay với lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/squid start
Starting squid: init_cache_dir ufs... squid
```

Cấu hình của tập tin "/etc/logrotate.d/squid"

Cấu hình tập tin "/etc/logrotate.d/squid" của bạn để nó tự động quay vòng sau mỗi tuần như sau:

Tạo tập tin squid (touch "/etc/logrotate.d/squid") và thêm vào:

```
/var/log/squid/access.log {
    weekly
    rotate 5
    copytruncate
    compress
    notifempty
    missingok
}
/var/log/squid/cache.log {
    weekly
    rotate 5
    copytruncate
    compress
    notifempty
    missingok
}
/var/log/squid/store.log {
    weekly
    rotate 5
    copytruncate
    compress
    notifempty
    missingok
# This script asks squid to rotate its logs on its own.
# Restarting squid is a long process and it is not worth
# doing it just to rotate logs
    postrotate
      /usr/sbin/squid -k rotate
    endscript
}
```

Bảo mật Squid

Nhiều điều khiển trong việc gắn vào (mounting) thư mục cache của Squid

Nếu bạn đã tạo thư mục cache của Squid trên một phân vùng (partition) riêng lẻ của hệ thống Linux của bạn (ví dụ /cache) giống như chúng ta đã làm trong quá trình cài đặt, bạn có thể sử dụng những đặc trưng noexec, nodev và nosuid để cải tiến và củng cố việc bảo mật cache. Những đặc trưng này có thể cài đặt trong tập tin "/etc/fstab" và báo cho hệ thống không cho phép sự thực thi của bất kỳ tập tin nhị phân (noexec), không phiên dịch những đặc tính hoặc ngăn chặn những thiết bị đặt biệt, không cho phép các bit set-user-identifier hoặc set-group-identifier tác động lên tập tin hệ thống được mount (/cache trong ví dụ của chúng ta). Áp dụng thủ tục này trên partition mà ở đó Squid Cache đang tồn tại sẽ giúp đỡ loại trừ khả năng của các DEV, SUID/SGID và sự thực thi của bất kỳ tập tin nhị phân nào.

Như ví dụ, giả sử rằng "/dev/sda8" là partition trên hệ thống ở đó thư mục /cache của Squid đang tồn tại, bạn phải hiệu chỉnh tập tin fstab (vi /etc/fstab) và thay đổi những dòng liên quan tới "/dev/sda8":

```
/dev/sda8       /cache  ext2    defaults                1 2
```

Trở thành:

```
/dev/sda8       /cache  ext2    noexec,nodev,nosuid     1 2
```

Chú ý: *Bạn phải reboot lại máy để những thay đổi có hiệu lực.*

Làm mất tác hại những tập tin cấu hình quan trọng như "squid.conf".

Như chúng ta đã biết bit không thay đổi (immutable bit) có thể được dùng để ngăn chặn sự xoá đi, ghi đè hoặc tạo một biểu tượng liên kết tới một file. Một khi tập tin squid.conf của bạn được cấu hình, tốt nhất là bạn làm cho nó không thể thay đổi với lệnh sau:

```
[root@deep /]# chattr +i /etc/squid/squid.conf
```

Tối ưu Squid

Các thuộc tính atime và noatime

Các thuộc tính atime và noatime có thể được dùng để thử tăng năng thực thi trong thư mục cache của Squid. Xem chương "Tổng quan về tối ưu hệ thống" cho nhiều thông tin về vấn đề này.

Bộ nhớ vật lý

Tài nguyên quan trọng nhất của Squid là bộ nhớ vật lý (physical memory). Bộ vi xử lý của bạn không cần quá nhanh. Đĩa cứng phải nhanh, khi đĩa cứng nhanh thì quan trọng cho việc lưu trữ số nhiều với tốc độ cao.

Trình tiện ích cachemgr.cgi của Squid

Mặc định trình tiện ích cachemgr.cgi đã sẵn có khi bạn biên dịch và cài đặt Squid vào trong hệ thống của bạn. Nó được thiết kế để chạy thông qua giao diện Web và đưa ra các thống kê khác nhau về cấu hình và khả năng thực hiện của Squid.

Chương trình này được đặt dưới thư mục "/usr/lib/squid" và bạn phải đặt nó trong thư mục cgi-bin (ví dụ /home/httpd/cgi-bin) để có thể sử dụng nó. Theo những bước đơn giản dưới đây để sử dụng chương trình này.

Bước 1:

Chuyển chương trình cachemgr.cgi vào thư mục cgi-bin.

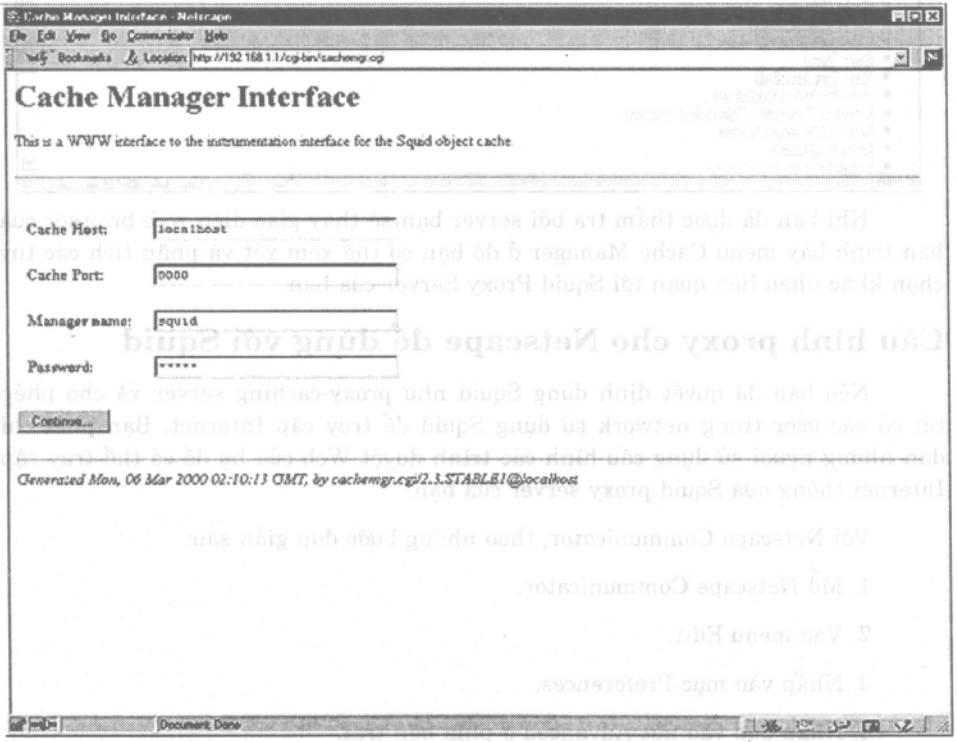
```
[root@deep /]# mv /usr/lib/squid/cachemgr.cgi /home/httpd/cgi-bin
```

Chú ý: *Tôi giả sử rằng thư mục cgi-bin được đặt dưới thư mục "/home/httpd/cgi-bin", những đường dẫn khác cũng có thể được. Thư mục cgi-bin sẽ sẵn có nếu bạn đã cài đặt Web server Apache trên hệ thống của bạn.*

Bước 2:

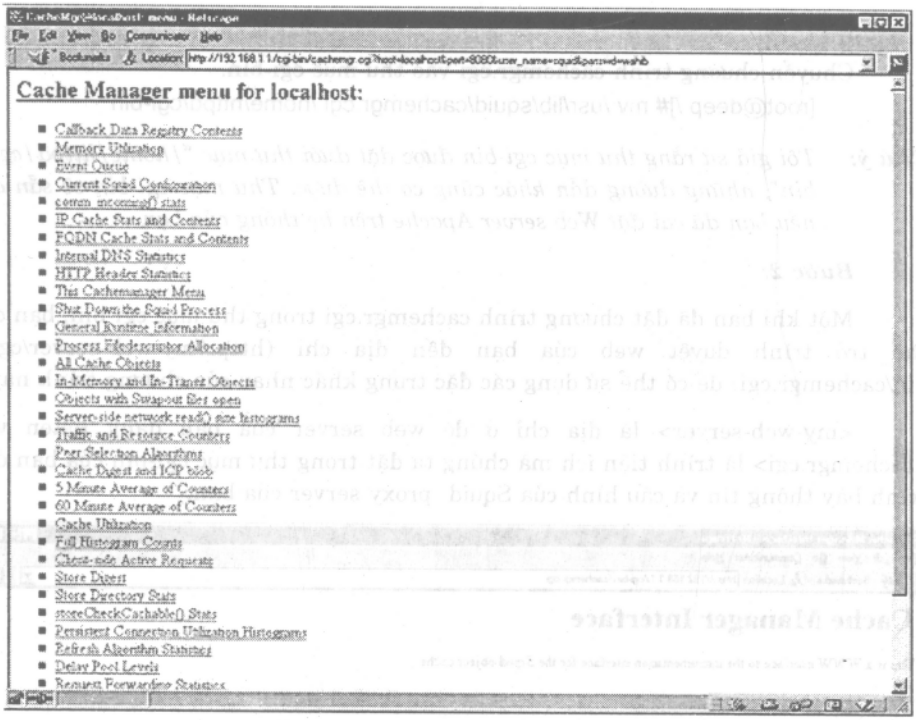
Một khi bạn đã đặt chương trình cachemgr.cgi trong thư mục cgi-bin, bạn có thể trỏ trình duyệt web của bạn đến địa chỉ (http://my-web-server/cgi-bin/cachemgr.cgi) để có thể sử dụng các đặc trưng khác nhau của chương trình này.

<my-web-server> là địa chỉ ở đó web server của bạn đang listen và <cachemgr.cgi> là trình tiện ích mà chúng ta đặt trong thư mục cgi-bin của bạn để trình bày thông tin và cấu hình của Squid proxy server của bạn.



Nếu bạn cấu hình tập tin squid sử dụng thẩm tra mật khẩu cho "cachemgr.cgi" bạn sẽ được đề nghị gõ vào thông tin về Cache Host, Cache Port,

Manager name và Password trước khi bạn có thể truy cập chương trình "cachemgr.cgi". Xem tập tin cấu hình "/etc/squid/squid.conf" ở trên cho nhiều thông tin.



Khi bạn đã được thẩm tra bởi server bạn sẽ thấy giao diện web browser của bạn trình bày menu Cache Manager ở đó bạn có thể xem xét và phân tích các tùy chọn khác nhau liên quan tới Squid Proxy Server của bạn.

Cấu hình proxy cho Netscape để dùng với Squid

Nếu bạn đã quyết định dùng Squid như proxy-caching server và cho phép tất cả các user trong network sử dụng Squid để truy cập Internet. Bạn phải chỉ dẫn những người sử dụng cấu hình các trình duyệt Web của họ để có thể truy cập Internet thông qua Squid proxy server của bạn.

Với Netscape Communicator, theo những bước đơn giản sau:

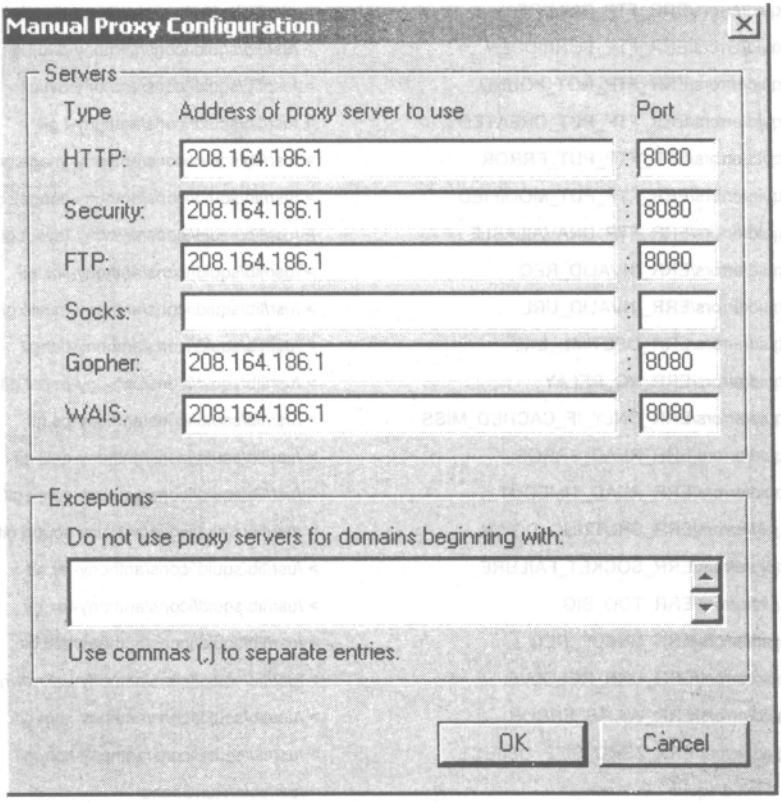
1. Mở Netscape Communicator.
2. Vào menu Edit.
3. Nhấp vào mục Preferences.
4. Nhấp đúp vào nút Advanced ở phía bên trái.
5. Nhấp vào tùy chọn Proxies.
6. Chọn trên nút Manual proxy configuration.

7. Nhấp vào nút View.

8. Điền thông tin cấu hình server của bạn vào các mục liên quan.

Ví dụ:

- HTTP: 208.164.186.1 Port: 8080
- Security: 208.164.186.1 Port: 8080
- FTP: 208.164.186.1 Port: 8080
- Gopher: 208.164.186.1 Port: 8080
- WAIS: 208.164.186.1 Port: 8080



Các tập tin được cài đặt

> /etc/squid
> /etc/squid/mib.txt
> /etc/squid/squid.conf.default
> /etc/squid/squid.conf
> /etc/squid/mime.conf.default
> /etc/rc.d/rc4.d/S90squid
> /etc/rc.d/rc5.d/S90squid
> /etc/rc.d/rc6.d/K25squid
> /etc/logrotate.d/squid
> /usr/lib/squid

> /etc/squid/mime.conf
> /etc/squid/errors
> /etc/squid/errors/ERR_ACCESS_DENIED
> /etc/squid/errors/ERR_CACHE_ACCESS_DENIED
> /etc/squid/errors/ERR_CACHE_MGR_ACCESS_DENIED
> /etc/squid/errors/ERR_CANNOT_FORWARD
> /etc/squid/errors/ERR_CONNECT_FAIL
> /etc/squid/errors/ERR_DNS_FAIL
> /etc/squid/errors/ERR_FORWARDING_DENIED
> /etc/squid/errors/ERR_FTP_DISABLED
> /etc/squid/errors/ERR_FTP_FAILURE
> /etc/squid/errors/ERR_FTP_FORBIDDEN
> /etc/squid/errors/ERR_FTP_NOT_FOUND
> /etc/squid/errors/ERR_FTP_PUT_CREATED
> /etc/squid/errors/ERR_FTP_PUT_ERROR
> /etc/squid/errors/ERR_FTP_PUT_MODIFIED
> /etc/squid/errors/ERR_FTP_UNAVAILABLE
> /etc/squid/errors/ERR_INVALID_REQ
> /etc/squid/errors/ERR_INVALID_URL
> /etc/squid/errors/ERR_LIFETIME_EXP
> /etc/squid/errors/ERR_NO_RELAY
> /etc/squid/errors/ERR_ONLY_IF_CACHED_MISS
> /etc/squid/errors/ERR_READ_ERROR
> /etc/squid/errors/ERR_READ_TIMEOUT
> /etc/squid/errors/ERR_SHUTTING_DOWN
> /etc/squid/errors/ERR_SOCKET_FAILURE
> /etc/squid/errors/ERR_TOO_BIG
> /etc/squid/errors/ERR_UNSUP_REQ
> /etc/squid/errors/ERR_URN_RESOLVE
> /etc/squid/errors/ERR_WRITE_ERROR
> /etc/squid/errors/ERR_ZERO_SIZE_OBJECT
> /etc/rc.d/init.d/squid
> /etc/rc.d/rc0.d/K25squid
> /etc/rc.d/rc1.d/K25squid
> /etc/rc.d/rc2.d/K25squid
> /etc/rc.d/rc3.d/S90squid
> /usr/lib/squid/dnsserver
> /usr/lib/squid/unlinkd
> /usr/lib/squid/cachemgr.cgi
> /usr/lib/squid/icons
> /usr/lib/squid/icons/anthony-binhex.gif
> /usr/lib/squid/icons/anthony-bomb.gif
> /usr/lib/squid/icons/anthony-box.gif
> /usr/lib/squid/icons/anthony-box2.gif
> /usr/lib/squid/icons/anthony-c.gif
> /usr/lib/squid/icons/anthony-compressed.gif
> /usr/lib/squid/icons/anthony-dir.gif
> /usr/lib/squid/icons/anthony-dirup.gif
> /usr/lib/squid/icons/anthony-dvi.gif
> /usr/lib/squid/icons/anthony-f.gif
> /usr/lib/squid/icons/anthony-image.gif
> /usr/lib/squid/icons/anthony-image2.gif
> /usr/lib/squid/icons/anthony-layout.gif
> /usr/lib/squid/icons/anthony-link.gif
> /usr/lib/squid/icons/anthony-movie.gif
> /usr/lib/squid/icons/anthony-pdf.gif
> /usr/lib/squid/icons/anthony-portal.gif
> /usr/lib/squid/icons/anthony-ps.gif
> /usr/lib/squid/icons/anthony-quill.gif
> /usr/lib/squid/icons/anthony-script.gif
> /usr/lib/squid/icons/anthony-sound.gif
> /usr/lib/squid/icons/anthony-tar.gif
> /usr/lib/squid/icons/anthony-tex.gif
> /usr/lib/squid/icons/anthony-text.gif
> /usr/lib/squid/icons/anthony-unknown.gif
> /usr/lib/squid/icons/anthony-xbm.gif
> /usr/lib/squid/icons/anthony-xpm.gif
> /usr/sbin/RunCache
> /usr/sbin/RunAccel
> /usr/sbin/squid
> /usr/sbin/client
> /var/log/squid

Chương 19:

PHẦN MỀM SERVER

(Dịch vụ Web)

Trong chương này chúng ta sẽ đề cập đến các vấn đề sau:

- Linux MM – Shared Memory Library for Apache.
- Linux Apache Web Server.
  - Biên dịch và cài đặt Apache.
  - Cấu hình.
  - Ngôn ngữ script PHP4 server-side.
  - Modun Perl Devel::Symdump.
  - Thư viện Perl CGI.pm.
  - An toàn cho Apache.
  - Chạy Apache trong một chroot jail.
  - Tối ưu Apache.



Linux MM – Shared Memory Library for Apache

Tổng Quan

Phần này chỉ giới thiệu cho bạn về cách biên dịch và cài đặt một chương trình nhỏ này nếu bạn dự định cài đặt và sử dụng máy chủ web Apache với những module của hãng thứ ba như là mod_ssl cho việc mã hoá dữ liệu, mod_perl cho ngôn ngữ lập trình Perl, hoặc mode_php cho ngôn ngữ PHP server-side. Chương trình này cung cấp khả năng thực thi đáng kể đến những module của Apache. Ví dụ nếu bạn cần cài đặt Apache với sự hỗ trợ của SSL dùng cho thương mại điện tử trên Internet, chương trình này sẽ cho phép giao thức SSL thực hiện trên vùng nhớ cao thay vì thực hiện trên đĩa cứng.

Như sự giải thích trên trang web MM Shared Memory Library.

MM Library là thư viện trừu tượng hai lớp, điều này sẽ làm đơn giản cách sử dụng chia sẻ bộ nhớ giữa các tiến trình dưới môi trường UNIX. Ở lớp đầu tiên nó sẽ ẩn tất cả sự thực hiện dựa trên những chi tiết khi làm việc với những phân đoạn chia sẻ bộ nhớ, và trên lớp hai nó hỗ trợ ở chế độ cao malloc(3) – style API cho một tiện lợi và là phương pháp tốt để làm việc với những cấu trúc dữ liệu bên trong những phân đoạn chia sẻ bộ nhớ.

Thư viện này được đưa ra dưới dạng mã nguồn mở open-source (BSD-style), bởi vì nguyên thủy nó được viết như một sự đề xuất cho việc sử dụng bên trong phiên bản tiếp theo của web server Apache và như một thư viện nền cho việc chia sẻ bộ nhớ dùng chung (bởi vì hiện nay những module Apache có thể chỉ sử dụng bộ nhớ heap-allocated, nó không bị chia sẻ ngang qua những tiến trình server pre-forked). Yêu cầu thực tế đến từ toàn diện những module giống như mod_ssl, mod_perl và mod_php, chúng rất thuận lợi để chia sẻ bộ nhớ dùng chung.

Những hướng dẫn cài đặt :

Những lệnh tương thích với UNIX.

Đường dẫn nguồn là "/var/tmp" (Có thể là một đường dẫn khác).

Sự cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Những bước cài đặt sẽ thực hiện với quyền của tài khoản supper-user (root).

Phiên bản MM là 1.1.2.

Packages

Vào trang web chủ của MM : http://www.engelschall.com/sw/mm

Bạn phải chắc chắn tải đúng tập tin : mm-1.1.2.tar.gz

Dạng lưu trữ Tarballs

Trước hết hãy tạo ra danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt MM, và sau đó so sánh chúng bằng cách dùng lệnh "diff" phát hiện những tập tin nào và được đặt ở đâu. Đơn giản chạy lệnh "find /* >MM1" trước và

tiếp theo thực hiện lệnh "find /* >MM2" sau cùng dùng lệnh "diff MM1 MM2 >Mminstalled" để tìm ra danh sách các tập tin bị thay đổi.

Sự biên dịch

Những bước giải nén tập tin đuôi .tar.gz.

```
[root@deep /]# cp mm-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf mm-version.tar.gz
```

Biên dịch :

Bước 1 :

Di chuyển đến thư mục mm mới và gõ lệnh sau :

```
./configure \
--disable-shared \
--prefix=/usr
```

Điều này nhắc MM tự thiết đặt chính nó cho việc cài đặt phần cứng quen thuộc

- Làm mất tác dụng chia sẻ những thư viện.

Bước 2 :

Bây giờ, chúng ta phải biên dịch và cài đặt thư viện chia sẻ bộ nhớ MM trong máy chủ.

```
[root@deep mm-1.1.2]# make
[root@deep mm-1.1.2]# make test
[root@deep mm-1.1.2]# make install
```

Chú ý : *Lệnh " make test " sẽ tạo nên vài kiểm tra quan trọng trên chương trình để chắc chắn nó hoạt động, và chắc rằng có sự phản hồi trước khi cài đặt.*

Dọn dẹp sau quá trình cài đặt

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf mm-version/ mm-version.tar.gz
```

Lệnh "rm" sẽ xóa tất cả những tập tin nguồn chúng ta đã dùng biên dịch và cài đặt MM. Nó cũng sẽ xóa tập tin nơi lưu trữ nén MM từ thư mục "/var/tmp".

Tài liệu tham khảo

Để biết thêm nhiều thông tin, bạn cần đọc thêm những tài liệu sau:

MM(3) – Thư viện chia sẻ bộ nhớ.

Mm – config(1) – MM thư viện chức năng cấu hình hoặc cài đặt.

Những tập tin đã cài đặt

/usr/bin/mm-config

/usr/include/mm.h

/usr/lib/libmm.la

/usr/lib/libmm.a

/usr/man/man1/mm-config.1

/usr/man/man3/mm.3

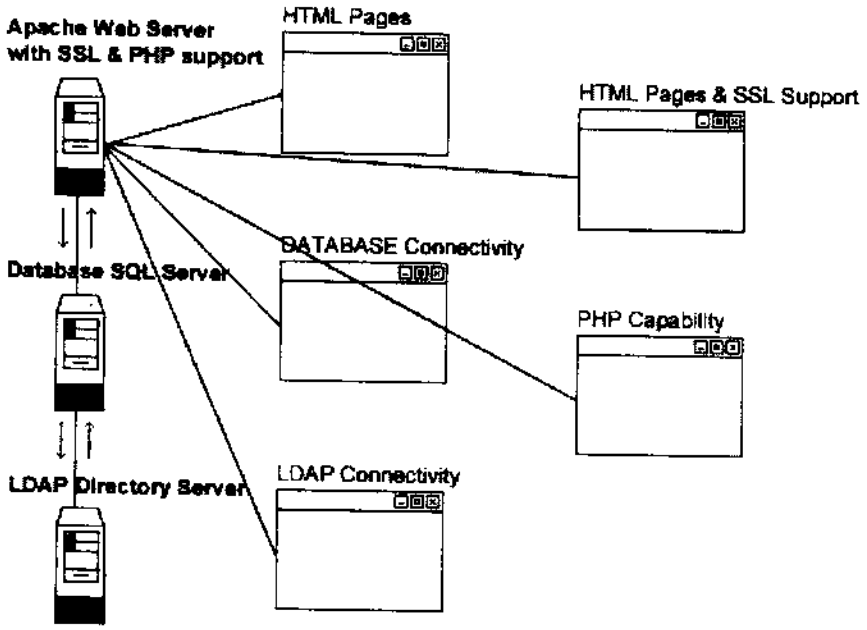
Linux Apache Web Server

Tổng Quan

Ngày nay trên thế giới HTTP- server của Apache được sử dụng rộng rãi. Nó vượt trội hơn các đối thủ cạnh tranh trên thương trường nhờ vào tất cả đều miễn phí, và có nhiều đặc trưng hơn các đối thủ thuộc họ UNIX. Apache là web server được sử dụng rộng rãi nhất làm web server cho hệ thống Linux. Với web server Apache, chức năng đơn giản nhất của nó là: đây là phần mềm dùng để hiển thị và hỗ trợ đưa những trang HTML trên máy chủ đến trình duyệt của người sử dụng để nó hiểu mã của HTML đó. Cài đặt nó với những module và chương trình của hãng thứ ba, làm cho nó trở nên mạnh hơn, những điều này sẽ hỗ trợ mạnh và những dịch vụ hữu ích cho trình duyệt của người dùng.

Tôi mong muốn rằng tất cả những người đọc quyển sách này đều sẽ cảm thấy thích thú vì sẽ biết cách như thế nào để cài đặt web server Apache một cách chắc chắn và an toàn. Trong vấn đề cài đặt, Apache không khó khăn để cài đặt hơn so với các phần mềm khác chúng ta đã cài đặt trên máy chủ Linux. Những thủ tục này đòi hỏi chúng ta phải khéo léo khi chúng ta muốn thêm vào một số module hoặc những chương trình của hãng thứ ba. Có nhiều tính năng, sự khác nhau và sự chọn lựa cho việc cài đặt Apache. Do đó trong phần này chúng tôi hỗ trợ một số ví dụ từng bước để bạn có thể biết được cài đặt Apache với những module và chương trình của hãng thứ ba giống như mod_ssl, mod_perl, PHP4, LDAP, ...v...v. Dĩ nhiên, xây dựng những chương trình này là tùy chọn, và bạn đã sẵn sàng chỉ để biên dịch những gì bạn muốn. (có nghĩa là, bạn có thể muốn biên dịch Apache với sự hỗ trợ cho PHP4, nhưng không có hỗ trợ giao thức bảo mật SSL hoặc khả năng kết nối cơ sở dữ liệu PostgreSQL). Để đơn giản chúng ta có một số yêu cầu trước hết cho mỗi ví dụ. Nếu những điều này không phù hợp hoàn cảnh của các bạn, để đơn giản thì bạn hãy điều chỉnh lại những bước này.

Trong chương này, chúng ta giải thích và thực hiện vài cách đơn giản để làm thế nào bạn có thể điều chỉnh cấu hình và cải thiện tốc độ của máy chủ. Cũng vậy, cho người quan tâm, chúng tôi sẽ hỗ trợ một thủ tục có thể chạy được Apache không phải là root-user và trong môi trường chrooted cho sự tối ưu bảo mật.



Biên dịch và cài đặt Apache

Những hướng dẫn cài đặt.

Những lệnh tương thích UNIX.

Nơi lưu trữ nguồn là "/var/tmp" (có thể là những đường dẫn khác).

Những cài đặt được kiểm tra trên Rad Hat Linux 6.1 và 6.2.

Những bước cài đặc này chỉ thực hiện bởi super-user "root".

Apache với phiên bản 1.3.12.

Mod_SSL với phiên bản 2.6.4 – 1.3.12.

Mod_Perl với phiên bản là 1.24.

Mod_PHP với phiên bản là 4.0.0.

Những gói tin

Trang chủ Apache : http://www.apache.org/.

Trang Apache FTP : 63.211.145.10.

Bạn phải chắc chắn tải về : apache_1.3.12.tar.gz.

Trang chủ Mod_SSL : http://www.modssl.org/.

Trang Mod_SSL FTP : 129.132.7.171.

Bạn phải chắc chắn tải về : mod_ssl-2.6.4-1.3.12.tar.gz.

Trang chủ Mod_Perl : http://www.perl.apache.org/.

Trang Mod_Perl FTP : 63.211.145.10.

Bạn phải chắc chắn tải về : mod_perl-1.24.tar.gz.

Trang chủ Mod_PHP: http://www.php.net/.

Bạn phải chắc chắn tải về : php-4.0.0.tar.gz.

Các yêu cầu trước hết :

- OpenSSL nên được cài đặt trên hệ thống của bạn nếu bạn muốn hỗ trợ mã hoá Apache và SSL.
- PostgreSQL nên được cài đặt trên hệ thống của bạn nếu bạn muốn hỗ trợ liên kết dữ liệu Apache và PostgreSQL.
- MM nên được cài đặt trên hệ thống của bạn nếu bạn muốn hỗ trợ tốc độ cao dựa trên bộ nhớ.
- OpenLDAP nên được cài đặt trên hệ thống của bạn nếu bạn muốn hỗ trợ liên kết thư mục Apache và LDAP.
- IMAP và POP nên được cài đặt trên hệ thống của bạn nếu bạn muốn hỗ trợ khả năng Apache và IMAP & POP.

Chú ý : *Để tìm hiểu nhiều thông tin với yêu cầu phần mềm, bạn hãy tham khảo trong quyển sách này.*

Dạng lưu trữ Tarballs

Trước hết hãy tạo danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt Apache, và sau đó so sánh chúng dùng lệnh "**diff**" để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh "**find /* >** Apache1" trước và "**find /* >** Apache2" sau khi bạn cài đặt phần mềm, và sử dụng lệnh "**diff** Apache1 Apache2 **>** Apache-**Installed**" để lấy một danh sách các thay đổi nào đó.

Biên dịch

Giải nén taballs (tar.gz)

```
[root@deep /]# cp apache_version.tar.gz /var/tmp
[root@deep /]# cp mod_ssl-version-version.tar.gz /var/tmp
[root@deep /]# cp mod_perl-version.tar.gz /var/tmp
[root@deep /]# cp php-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp/
[root@deep tmp]# tar xzpf apache_version.tar.gz
[root@deep tmp]# tar xzpf mod_ssl-version-version.tar.gz
[root@deep tmp]# tar xzpf mod_perl-version.tar.gz
[root@deep tmp]# tar xzpf php-version.tar.gz
```

Biên dịch và tối ưu

Bước 1

Web server Apache giống như nhiều ứng dụng chúng ta đã cài đặt, không thể chạy như super-user root. Vì lý do này chúng ta phải tạo một tài khoản người dùng đặc biệt mà nó có ít quyền truy cập hệ thống, và vẫn đủ quyền để chạy web server Apache. Đó là cách tốt nhất để chọn và tạo một tài khoản người dùng mới mục đích chỉ để chạy daemon web server.

```
[root@deep /]# useradd -c "Apache Server" -u 80 -s /bin/false -r -d
/home/httpd www 2>/dev/null || :
```

Bước 2

Áp dụng mod_ssl để Apache source tree

Nếu bạn muốn sử dụng, kể cả hỗ trợ mã hoá dữ liệu SSL trong web server Apache, bạn di chuyển vào thư mục nguồn mới mod_ssl (cd mod_ssl-version-version/) và nhập những câu lệnh sau trên cửa sổ lệnh:

```
CC="egcs" \
CFLAGS="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomitframe-pointer -fno-exceptions" \
./configure \
--with-apache=../apache_1.3.12 \
--with-crt=/etc/ssl/certs/server.crt \
--with-key=/etc/ssl/private/server.key
```

Tùy chọn "—with-apache" chỉ rõ nơi lưu trữ của thư mục nguồn Apache (Điều quan trọng để chú ý là chúng ta cho rằng phiên bản Apache trong ví dụ này là 1.3.12), với tùy chọn "—with-crt" chỉ rõ nơi lưu trữ của khóa công cộng sẵn có cho sự mã hóa SSL, và tùy chọn "—with-key" " chỉ rõ nơi lưu trữ của khóa riêng đang tồn tại cho sự mã hóa SSL.

Chú ý : *Phần mềm OpenSSL nên được cài đặt sẵn trên máy chủ của bạn và những khóa chung và riêng phải có sẵn hoặc đã được tạo trên máy chủ, hoặc bạn sẽ nhận một thông báo lỗi trong suốt thời gian cấu hình mod_ssl. Tham khảo chương "Servers Software (Encrypting Network Services)" trong sách này để biết thêm nhiều thông tin về "Phần mềm máy chủ (Những dịch vụ mã hóa mạng)".*

Bước 3

Cải thiện tham số MaxClients của Apache

Bởi mặc định số lớn nhất trong tập tin cấu hình Apache (httpd.conf) bạn có thể đặt cho tham số MaxClients là 256. Cho một trang nhiều truy cập, và để thực hiện tốt hơn, Nó yêu cầu bạn tăng giới hạn của tham số này. Bạn có thể sửa tập tin "src/include/httpd.h" trong thư mục nguồn của Apache và thay đổi giá trị mặc định.

Di chuyển đến thư mục nguồn mới của Apache (cd ../apache_1.3.12/) và sửa tập tin **httpd.h** (vi +333 src/include/httpd.h), thay đổi dòng :

```
#define HARD_SERVER_LIMIT 256
```

Trở thành:

```
#define HARD_SERVER_LIMIT 1024
```

Chú ý : *Nếu bạn muốn cấu hình Apache không có hỗ trợ mod_ssl, thì bạn thay đổi giá trị mặc định sẽ là 316 thay vì 333.*

Bước 4

Cấu hình lại Apache cho bước cấu hình PHP4

Nếu bạn muốn dùng và kể cả hỗ trợ ngôn ngữ phiên dịch server-side PHP4 trên web server Apache của bạn, sau đó di chuyển vào thư mục nguồn mới Apache (cd apache_1.3.12/) nếu bạn chưa có hãy nhập những câu lệnh sau trên cửa sổ lệnh:

```
CC="egcs" \
OPTIM="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomit-framepointer -fno-exceptions" \
CFLAGS="-DDYNAMIC_MODULE_LIMIT=0" \
./configure \
--prefix=/home/httpd \
--bindir=/usr/bin \
--sbindir=/usr/sbin \
--libexecdir=/usr/lib/apache \
--includedir=/usr/include/apache \
--sysconfdir=/etc/httpd/conf \
--localstatedir=/var \
--runtimedir=/var/run \
--logfiledir=/var/log/httpd \
--datadir=/home/httpd \
--proxycachedir=/var/cache/httpd \
--mandir=/usr/man
```

Chú ý : *Bước này chỉ cần thiết nếu bạn muốn hỗ trợ cả PHP4 trong mã nguồn Apache của bạn khi chúng ta cấu hình lại Apache cho các bước cấu hình của PHP4. Hãy chú ý là tùy chọn "-DDYNAMIC_MODULE_LIMIT=0" sẽ không cho phép sử dụng các modules được nạp một cách tự động trong sự biên dịch của Apache, và sẽ cải thiện tốc độ.*

Cấu hình PHP4 và ứng dụng nó cho Apache

Bây giờ, di chuyển đến thư mục mới PHP4 (cd ../php-4.0) và nhập những câu lệnh sau trên cửa sổ lệnh:

1/ Hiệu chỉnh tập tin **php_pgsql.h** (vi +46 ext/pgsql/php_pgsql.h) và thay đổi những dòng sau:

```
#include <libpq-fe.h>
#include <libpq/libpq-fs.h>
```

Đổi thành :

```
#include </usr/include/pgsql/libpq-fe.h>
#include </usr/include/pgsql/libpq/libpq-fs.h>
```

Những thay đổi trong tập tin "php_pgsql.h" là cần thiết để cho biết vị trí những tập tin "libpqfe.h", and "libpq-fs.h" của cơ sở dữ liệu PostgreSQL trong lúc cấu hình PHP4. Trong Red Hat Linux, những thư viện của PostgreSQL được lưu trữ trong thư mục (/usr/include/pgsql).

2/ Bây giờ chúng ta phải cấu hình và cài đặt PHP4 trong server Linux:

```
CC="egcs" \
CFLAGS="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomitframe- pointer -fno-exceptions -I/usr/include/openssl"
\
./configure \
--prefix=/usr \
--with-exec-dir=/usr/bin \
--with-apache=../apache_1.3.12 \
--with-config-file-path=/etc/httpd \
--disable-debug \
--enable-safe-mode \
--with-imap \ (if you want IMAP & POP support).
--with-ldap \ (if you want LDAP database light directory support).
--with-pgsql \ (if you want PostgreSQL database support).
--with-mm \
--enable-inline-optimization \
--with-gnu-ld \
--enable-memory-limit
```

Việc này bảo PHP4 tự thiết lập chính nó cho phần cứng quen thuộc với:

- Biên dịch không có những biểu tượng gỡ rối.
- Mặc nhiên ở chế độ an toàn.
- Hỗ trợ cả IMAP & POP.
- Hỗ trợ cả LDAP Directory.

- Hỗ trợ hệ quản trị cơ sở dữ liệu PostgressSQL.
- Hỗ trợ mm để cải tiến hiệu suất của thư viện bộ nhớ.
- Cho phép inline-optimization cho xử lý tốt hơn.
- Biên dịch với hỗ trợ giới hạn bộ nhớ.
- Giả sử trình biên dịch C dùng GNU Id.

```
[root@deep php-4.0]# make
[root@deep php-4.0]# make install
```

Bước 5 :

Thêm mod_perl vào cây nguồn Apache và xây dựng/cài đặt Perl-side của mod_perl. Nếu bạn muốn dùng và hỗ trợ cả ngôn ngữ lập trình Perl trong web server Apache, thì bạn chuyển vào thư mục nguồn mod_perl mới (cd ../mod_perl-1.24/) và gõ những câu lệnh sau trên cửa sổ lệnh của bạn :

```
perl Makefile.PL \
EVERYTHING=1 \
APACHE_SRC=../apache_1.3.12/src \
USE_APACI=1 \
PREP_HTTPD=1 \
DO_HTTPD=1

[root@deep mod_perl-1.24]# make
[root@deep mod_perl-1.24]# make install
```

Bước 6 :

Xây dựng/cài đặt Apache có/không có mod_ssl +- PHP4 và/hoặc mod_perl

Một khi bạn đã đưa vào trong mã nguồn của Apache những module của hãng thứ ba mà bạn muốn hỗ trợ và sử dụng, đó là thời gian biên dịch và cài đặt nó vào trong hệ thống Linux. Bước cuối cùng là di chuyển nó đến thư mục nguồn Apache mới (cd ../apache_1.3.12/) và gõ những câu lệnh sau trên cửa sổ lệnh của bạn dựa trên những gì bạn muốn cài đặt với Apache:

```
SSL_BASE=SYSTEM \ ( Yêu cầu chỉ hỗ trợ cho mod_ssl).
EAPI_MM=SYSTEM \ (Yêu cầu chỉ hỗ trợ nếu bạn dùng thư viện chia sẻ
bộ nhớ mm cho Apache).
CC="egcs" \
OPTIM="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomit-framepointer
-fno-exceptions" \

CFLAGS="-DDYNAMIC_MODULE_LIMIT=0" \
```

```
./configure \
--prefix=/home/httpd \
--bindir=/usr/bin \
--sbindir=/usr/sbin \
--libexecdir=/usr/lib/apache \
--includedir=/usr/include/apache \
--sysconfdir=/etc/httpd/conf \
--localstatedir=/var \
--runtimedir=/var/run \
--logfiledir=/var/log/httpd \
--datadir=/home/httpd \
--proxycachedir=/var/cache/httpd \
--mandir=/usr/man \
--add-module=src/modules/experimental/mod_mmap_static.c \
--add-module=src/modules/standard/mod_auth_db.c \
--enable-module=ssl \
--enable-rule=SSL_SDBM \
--disable-rule=SSL_COMPAT \
--activate-module=src/modules/php4/libphp4.a \
--enable-module=php4 \
--activate-module=src/modules/perl/libperl.a \
--enable-module=perl \
--disable-module=status \
--disable-module=userdir \
--disable-module=negotiation \
--disable-module=autoindex \
--disable-module=asis \
--disable-module=imap \
--disable-module=env \
--disable-module=actions
```

Apache tự nhận dạng những phần cứng đặc biệt

- Module mod_mmap để cải thiện tốc độ.
- Module mod_auth_db bảo vệ xác thực mật khẩu cho người sử dụng.
- Module mod_ssl dùng mã hóa dữ liệu và an toàn truyền thông.
- Module mod_php4 dùng ngôn ngữ phiên dịch php server-side và cải thiện tốc độ tải những trang web được xây dựng bởi PHP.
- Module mod_perl dùng bảo vệ và tốc độ tốt hơn những đoạn cgi mặc định.
- Không cho phép module status.

- Không cho phép module userdir.
- Không cho phép module negotiation.
- Không cho phép module autoindex.
- Không cho phép module asis.
- Không cho phép module imap.
- Không cho phép module env.
- Không cho phép module actions.

Chú ý : *Điều quan trọng là xóa tất cả những module không cần thiết trong suốt thời gian cấu hình sẽ cải tiến được sự thực hiện của web server Apache. Trong cấu hình ở trên, chúng ta đã xóa hầu hết những module bất lợi làm cho tải hệ điều hành chậm hơn và giới hạn những rủi ro an toàn trong web server Apache. Tham khảo tài liệu về Apache cho thông tin trên mỗi phần*

Bước 7 :

Bây giờ, chúng ta phải cài đặt Apache trong máy chủ Linux

```
[root@deep apache_1.3.12]# make
[root@deep apache_1.3.12]# make install
[root@deep apache_1.3.12]# rm -f /usr/sbin/apachectl
[root@deep apache_1.3.12]# rm -f /usr/man/man8/apachectl.8
[root@deep apache_1.3.12]# rm -rf /home/httpd/icons/
[root@deep apache_1.3.12]# rm -rf /home/httpd/htdocs/
[root@deep apache_1.3.12]# cd /var/tmp/php-4.0
[root@deep php-4.0.0]# install -m 644 php.ini.dist /usr/lib/php.ini
[root@deep php-4.0.0]# rm -rf /etc/httpd/conf/ssl.crl/
[root@deep php-4.0.0]# rm -rf /etc/httpd/conf/ssl.crt/
[root@deep php-4.0.0]# rm -rf /etc/httpd/conf/ssl.csr/
[root@deep php-4.0.0]# rm -rf /etc/httpd/conf/ssl.key/
[root@deep php-4.0.0]# rm -rf /etc/httpd/conf/ssl.prm/
[root@deep php-4.0.0]# rm -f /etc/httpd/conf/srm.conf srm.conf.default
access.conf access.conf.default
```

Lệnh "make" sẽ biên dịch tất cả những tập tin nguồn thành những tập tin nhị phân binaries có thể thực thi được, và lệnh "make install" sẽ cài đặt các binaries và bất kỳ những tập tin hỗ trợ khác vào những nơi thích hợp. Lệnh "rm -f" sẽ xóa tập tin script nhỏ "apachectl" chịu trách nhiệm để khởi động và dừng daemon Apache bởi vì chúng ta dùng một tập tin script tốt hơn tên là "httpd" được lưu trữ dưới thư mục "/etc/rc.d/init.d/" đó là lợi điểm của Linux system V. Chúng ta cũng xóa thư mục "/home/httpd/icons" dưới Apache khi bạn dùng đặc trưng chỉ mục tự động của nó. Đây là đặc trưng có thể mang lại một rủi ro về an toàn, với lý do này chúng ta đã không cho phép nó trong tập tin cấu hình. Vì vậy, chúng ta có thể

an tâm xóa thư mục để tạo khoảng trống trên máy chủ Linux. Thư mục "/home/httpd/docs" lưu trữ tất cả những tập tin tài liệu liên quan đến Apache, do đó sau khi chúng ta đã hoàn thành việc tham khảo tài liệu chúng ta có thể xóa nó để tạo khoảng trống. Lệnh "install -m" sẽ cài đặt tập tin "php.ini.dist" trong thư mục "/etc/httpd", và sẽ đổi tên của tập tin này thành "php.ini", Đây là tập tin kiểm soát nhiều khía cạnh tác động của PHP. Những thư mục "ssl.crl", "ssl.crt", "ssl.csr", "ssl.key", và "ssl.prm" trong thư mục "/etc/httpd/conf" là tất cả những thư mục có liên quan đến SSL, và kiểm soát những khóa riêng và chung. Bởi vì chúng ta dùng nơi lưu trữ khác "/etc/ssl", chúng ta có thể xóa chúng một cách an toàn. Cuối cùng, chúng ta xoá những tập tin không dùng đến "srm.conf", "srm.conf.default", "access.conf", và "access.conf.default", Bây giờ điều này đã được kiểm soát bởi tập tin Apache "httpd.conf".

Dọn dẹp sau quá trình cài đặt

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf apache-version/ apache-version.tar.gz mod_ssl-version-version/ mod_ssl-versionversion.tar.gz php-version/ php-version.tar.gz mod_perl-version/ mod_perl-version.tar.gz
```

Lệnh "rm" sẽ xóa tất cả những tập tin nguồn chúng ta đã dùng để biên dịch và cài đặt Apache, mod_ssl, mod_perl, và php. Nó cũng sẽ xóa nơi lưu trữ những tập tin nén của Apache, mod_ssl, mod_perl, và php từ thư mục "/var/tmp".

Cấu hình

Những tập tin cấu hình cho dịch vụ khác nhau phụ thuộc rất cụ thể trên nhu cầu và kiến trúc mạng của bạn. Một số người có thể cài đặt máy chủ Apache chỉ dùng cho việc đưa những trang web lên; Người khác có thể cài đặt nó với liên kết cơ sở dữ liệu và thương mại điện tử với sự hỗ trợ của SSL, ...v...v. Trong cuốn sách này, chúng tôi cung cấp cho bạn một tập tin cấu hình "httpd.conf", với PHP, Perl, SSL, LDAP, và để xác thực mật khẩu, để bạn thấy được những khả năng khác nhau.

Chúng tôi sẽ quan tâm đến sự tối ưu và bảo mật của những tập tin này, và loại bỏ tất cả những hiệu chỉnh cụ thể để phù hợp với sở thích của bạn. Bạn sẽ cần tham khảo những tài liệu về những chương trình này, và hi vọng rằng bạn sẽ hiểu chúng.

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy Apache Server, yêu cầu những tập tin sau, chúng phải được tạo hoặc được copy tới những thư mục thích hợp trên server của bạn.

  Copy tập tin **httpd.conf** đến thư mục "/etc/httpd/conf/".

  Copy tập tin **Apache** đến thư mục "/etc/logrotate.d/".

  Copy đoạn tập tin **httpd** đến thư mục "/etc/rc.d/init.d/".

Tập tin cấu hình của "/etc/httpd/conf/httpd.conf"

Tập tin "httpd.conf" là tập tin cấu hình chính của web server Apache. Hiện có nhiều sự chọn lựa, và điều quan trọng là tham khảo tài liệu về Apache để được nhiều thông tin về những tham số và tiến trình cài đặt khác nhau. Ví dụ cấu hình sau là một tập tin cấu hình nhỏ đang chạy trên Apache. với sự hỗ trợ SSL. Cũng vậy, điều quan trọng để chú ý rằng chúng tôi chỉ giải thích những tham số có tác động tới bảo mật và tối ưu, còn lại những điều khác chính bạn tự nghiên cứu.

Hiệu chỉnh tập tin httpd.conf (vi /etc/httpd/conf/httpd.conf) và thêm/thay đổi :

```
### Đoạn 1 : Môi trường toàn cục.
    #
    ServerType standalone
    ServerRoot "/etc/httpd"
    PidFile /var/run/httpd.pid
    ResourceConfig /dev/null
    AccessConfig /dev/null
    Timeout 300
    KeepAlive On
    MaxKeepAliveRequests 0
    KeepAliveTimeout 15
    MinSpareServers 16
    MaxSpareServers 64
    StartServers 16
    MaxClients 512
    MaxRequestsPerChild 100000

### Đoạn 2: Cấu hình chính của máy chủ.
    #
    Port 80

    <IfDefine SSL>
    Listen 80
    Listen 443
    </IfDefine>

    User www
    Group www
    ServerAdmin admin@openna.com
    ServerName www.openna.com
    DocumentRoot "/home/httpd/ona"
```

```
<Directory />
  Options None
  AllowOverride None
  Order deny,allow
  Deny from all
</Directory>

<Directory "/home/httpd/ona">
  Options None
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>

<Files .pl>
  Options None
  AllowOverride None
  Order deny,allow
  Deny from all
</Files>

<IfModule mod_dir.c>
DirectoryIndex index.htm index.html index.php index.php3 default.html
index.cgi
</IfModule>

#<IfModule mod_include.c>
#Include conf/mmap.conf
#</IfModule>

UseCanonicalName On

<IfModule mod_mime.c>
TypesConfig /etc/httpd/conf/mime.types
</IfModule>

DefaultType text/plain
HostnameLookups Off

ErrorLog /var/log/httpd/error_log
LogLevel warn
```

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\""
combined
SetEnvIf Request_URI \.gif$ gif-image
CustomLog /var/log/httpd/access_log combined env=!gif-image
ServerSignature Off

<IfModule mod_alias.c>
ScriptAlias /cgi-bin/ "/home/httpd/cgi-bin/"
<Directory "/home/httpd/cgi-bin">
  AllowOverride None
  Options None
  Order allow,deny
  Allow from all
</Directory>

</IfModule>

<IfModule mod_mime.c>
AddEncoding x-compress Z
AddEncoding x-gzip gz tgz
AddType application/x-tar .tgz
</IfModule>

ErrorDocument 500 "The server made a boo boo.
ErrorDocument 404 http://192.168.1.1/error.htm
ErrorDocument 403 "Access Forbidden -- Go away.

<IfModule mod_setenvif.c>
BrowserMatch "Mozilla/2" nokeepalive
BrowserMatch "MSIE 4\.0b2;" nokeepalive downgrade-1.0 force-response-1.0
BrowserMatch "RealPlayer 4\.0" force-response-1.0
BrowserMatch "Java/1\.0" force-response-1.0
BrowserMatch "JDK/1\.0" force-response-1.0
</IfModule>
```

Đoạn 3: Virtual Hosts.

```
#
<IfDefine SSL>
AddType application/x-x509-ca-cert .crt
AddType application/x-pkcs7-crl .crl
</IfDefine>
```

```
<IfModule mod_ssl.c>
SSLPassPhraseDialog builtin
SSLSessionCache dbm:/var/run/ssl_scache
SSLSessionCacheTimeout 300

SSLMutex file:/var/run/ssl_mutex

SSLRandomSeed startup builtin
SSLRandomSeed connect builtin

SSLLog /var/log/httpd/ssl_engine_log
SSLLogLevel warn
</IfModule>

<IfDefine SSL>
<VirtualHost _default_:443>

DocumentRoot "/home/httpd/ona"
ServerName www.openna.com
ServerAdmin admin@openna.com
ErrorLog /var/log/httpd/error_log

SSLEngine on
SSLCipherSuite
ALL:!ADH:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL

SSLCertificateFile /etc/ssl/certs/server.crt
SSLCertificateKeyFile /etc/ssl/private/server.key
SSLCACertificatePath /etc/ssl/certs
SSLCACertificateFile /etc/ssl/certs/ca.crt
SSLCARevocationPath /etc/ssl/crl
SSLVerifyClient none
SSLVerifyDepth 10

SSLOptions +ExportCertData +StrictRequire

SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-shutdown
SetEnvIf Request_URI \.gif$ gif-image
CustomLog /var/log/httpd/ssl_request_log \
   "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b" env=!gif-
image
```

```
</VirtualHost>
</IfDefine>
```

Tập tin httpd.conf tự thiết đặt chính nó cho việc cài đặt cấu hình quen thuộc với:

ServerType standalone

Tùy chọn "ServerType" chỉ rõ Apache phải chạy trên hệ thống như thế nào. Bạn có thể chạy nó từ super-server inetd, hoặc như standalone daemon. Nó được đề nghị chạy Apache theo kiểu standalone để thi hành và có tốc độ tốt hơn.

ServerRoot "/etc/httpd"

Tùy chọn "ServerRoot" chỉ rõ thư mục ở đó lưu những tập tin cấu hình đang sử dụng của máy chủ Apache. Nó chỉ cho Apache biết nơi nào có thể tìm thấy những tập tin khi nó khởi động.

PidFile /var/run/httpd.pid

Tùy chọn "PidFile" chỉ rõ nơi mà ở đó máy chủ sẽ ghi lại id của tiến trình của daemon khi nó khởi động. Tùy chọn này chỉ được yêu cầu khi bạn cấu hình Apache ở kiểu standalone.

ResourceConfig /dev/null

Tùy chọn "ResourceConfig" chỉ rõ nơi lưu trữ tập tin cũ "srm.conf" nơi này Apache đọc sau khi nó đã đọc xong tập tin "httpd.conf". Khi bạn đặt đường dẫn đến "/dev/null", Apache cho phép bạn đưa vào nội dung của tập tin này trong tập tin "httpd.conf", và theo cách này, bạn vừa có một tập tin kiểm soát tất cả những tham số cấu hình cho đơn giản.

AccessConfig /dev/null

Tùy chọn "AccessConfig" chỉ rõ nơi lưu trữ tập tin cũ "access.conf" nơi này Apache đọc sau khi nó đã đọc xong tập tin "srm.conf". Khi bạn đặt đường dẫn đến "/dev/null", Apache cho phép bạn đưa vào nội dung của tập tin này trong tập tin "httpd.conf", và theo cách này, bạn vừa có một tập tin kiểm soát tất cả những tham số cấu hình cho đơn giản.

Timeout 300

Tùy chọn "Timeout" chỉ rõ khoảng thời gian Apache sẽ chờ cho một yêu cầu GET, POST, PUT và phát tín hiệu ACKs. Bạn có thể an toàn bỏ đi tùy chọn trên giá trị mặc định của nó.

KeepAlive On

Tùy chọn "KeepAlive", nếu mở "On", cho phép duy trì kết nối liên tục trên web server. Để thực hiện tốt hơn, nên để chọn là "On", và cho phép nhiều hơn một yêu cầu trên kết nối.

MaxKeepAliveRequests 0

Tùy chọn "MaxKeepAliveRequests" chỉ rõ số yêu cầu cho phép trên kết nối khi tùy chọn "KeepAlive" ở trên được để "On". Khi giá trị bằng "0" thì không có giới hạn yêu cầu được phép trên máy chủ. Tăng khả năng thực hiện cho máy chủ, nên cho phép không giới hạn những yêu cầu.

KeepAliveTimeout 15

Tùy chọn "KeepAliveTimeout" chỉ rõ khoảng thời gian được tính bằng giây, Apache sẽ chờ yêu cầu đến sau trước khi ngắt kết nối. "15" giây là một giới hạn tốt cho máy chủ thực hiện.

MinSpareServers 16

Tùy chọn "MinSpareServers" chỉ rõ số nhỏ nhất của idle child server processes cho Apache, điều này không giải quyết một yêu cầu. Đây là tham số điều chỉnh quan trọng đối với sự thực hiện của web server Apache. Sự hoạt động với trọng tải cao, "16" là một giá trị đề nghị bởi những điểm chuẩn khác nhau trên Internet.

MaxSpareServers 64

Tùy chọn "MaxSpareServers" chỉ rõ số lớn nhất của idle child server processes cho Apache, điều này không giải quyết một yêu cầu. Đây là tham số điều chỉnh quan trọng đối với sự thực hiện của web server server Apache. Sự hoạt động với trọng tải cao, "64" là một giá trị đề nghị bởi những điểm chuẩn khác nhau trên Internet.

StartServers 16

Tùy chọn "StartServers" chỉ rõ số của child server processes nó sẽ được tạo bởi Apache lúc khởi động. Điều này, một lần nữa là tham số điều chỉnh quan trọng đối với sự thực hiện của web server Apache. Sự hoạt động tải trọng cao, "16" là một giá trị đề nghị bởi những điểm chuẩn khác nhau trên Internet.

MaxClients 512

"MaxClients" chỉ rõ số những yêu cầu xảy ra cùng một thời điểm nó có thể được hỗ trợ bởi Apache. đây cũng là tham số điều chỉnh quan trọng đối với sự thực hiện của web server server Apache. Đối với web server thực hiện với tải trọng cao, "512" là một giá trị đề nghị bởi những chuẩn khác nhau trên Internet.

MaxRequestsPerChild 100000

Tùy chọn "MaxRequestsPerChild" chỉ rõ số những yêu cầu của một child server process riêng lẻ sẽ xử lý. Đây cũng là tham số điều chỉnh quan trọng đối với sự thực hiện của web server server Apache.

User www

Tùy chọn "User" chỉ rõ UID mà máy chủ Apache sẽ chạy. Điều quan trọng để tạo mới một người dùng có quyền tối thiểu truy cập hệ thống, và những chức năng chạy được web server daemon.

Group www

Tùy chọn "Group" chỉ rõ GID mà máy chủ Apache sẽ chạy. Điều quan trọng để tạo mới một nhóm người dùng có quyền tối thiểu truy cập hệ thống, và những chức năng chạy được web server daemon.

DirectoryIndex index.htm index.html index.php index.php3 default.html index.cgi

Tùy chọn "DirectoryIndex" chỉ rõ những tập tin dùng bởi Apache như là một thư mục index HTML viết sẵn. Nói cách khác. nếu Apache không thấy trang index mặc định để hiển thị, nó sẽ cố tìm danh sách kế tiếp trong tham số này, nếu sẵn có. Để tăng tốc độ của web server nên có danh sách chứa hầu hết những trang index mặc định để hiển thị của những trang web đầu tiên.

Include conf/mmap.conf

Tùy chọn "Include" chỉ rõ nơi lưu trữ những tập tin khác mà bạn có thể đưa vào từ bên trong những tập tin cấu hình (httpd.conf). Trong trường hợp của chúng ta, chúng ta đưa vào tập tin "mmap.conf" được lưu trữ trong thư mục "/etc/httpd/conf". Tập tin ("mmap.conf") sắp xếp các tập tin vào trong bộ nhớ để truy xuất nhanh hơn. Tham khảo phần "Optimizing Apache" để có nhiều thông tin hơn.

HostnameLookups Off

Tùy chọn "HostnameLookups", nếu chọn "Off" định rõ không cho DNS tra cứu. Nên chọn "Off" để mà có giảm thời gian lưu trông trên mạng, và cải thiện tốc độ của web server server Apache.

Cấu hình của tập tin "/etc/logrotate.d/apache"

Cấu hình tập tin "/etc/logrotate.d/apache" luân phiên mỗi tuần ghi lại quá trình hoạt động của Apache một cách tự động.

Tạo tập tin Apache (touch /etc/logrotate.d/apache) và thêm vào :

```
/var/log/httpd/access_log {
    missingok
    postrotate
    /usr/bin/killall -HUP httpd
    endscript
}
/var/log/httpd/error_log {
    missingok
```

```
        postrotate
            /usr/bin/killall -HUP httpd
        endscript
}

/var/log/httpd/ssl_request_log {
        missingok
        postrotate
            /usr/bin/killall -HUP httpd
        endscript
}

/var/log/httpd/ssl_engine_log {
        missingok
        postrotate
            /usr/bin/killall -HUP httpd
        endscript
}
```

Chú ý: *Những dòng để luân phiên tự động ghi những tập tin log SSL có tên là "ssl_request_log" và "ssl_engine_log" được đưa vào trong tập tin này. Nếu bạn có ý định chạy Apache không có hỗ trợ của SSL, bạn phải xóa những dòng có liên quan đến SSL.*

Cấu hình của tập tin script"/etc/rc.d/init.d/httpd"

Cấu hình của đoạn tập tin "/etc/rc.d/init.d/httpd" để khởi động và dừng web server Apache.

Tạo tập tin script httpd (touch /etc/rc.d/init.d/httpd) và thêm vào :

```
#!/bin/sh
#
# Startup script for the Apache Web Server
#
# chkconfig: 345 85 15
# description: Apache là một máy chủ World Wide Web. Nó dùng để # phục vụ các tập tin HTML and CGI.
# processname: httpd
# pidfile: /var/run/httpd.pid
# config: /etc/httpd/conf/httpd.conf

# Source function library.
```

```
. /etc/rc.d/init.d/functions

# Xem chúng tôi gọi nó như thế nào
  case "$1" in
  start)
 echo -n "Starting httpd: "
 daemon httpd -DSSL
 echo
 touch /var/lock/subsys/httpd
 ;;
 stop)
 echo -n "Shutting down http: "
 killproc httpd
 echo
 rm -f /var/lock/subsys/httpd
 rm -f /var/run/httpd.pid
 ;;
 status)
 status httpd
 ;;
 restart)
 $0 stop
 $0 start
 ;;
 reload)
 echo -n "Reloading httpd: "
 killproc httpd -HUP
 echo
 ;;
 *)
 echo "Usage: $0 {start|stop|restart|reload|status}"
 exit 1
 esac
exit 0
```

Bây giờ, làm cho tập tin này có thể thực thi được và thay đổi những quyền mặc định :

```
[root@deep /]# chmod 700 /etc/rc.d/init.d/httpd
```

Tạo biểu tượng liên kết rc.d cho Apache với câu lệnh sau :

```
[root@deep /]# chkconfig --add httpd
```

Khởi động máy chủ Apache mới bằng tay với câu lệnh sau :

```
[root@deep /]# /etc/rc.d/init.d/httpd start
Starting httpd: [ OK ]
```

Chú ý : *Tùy chọn "-DSSL" sẽ khởi động Apache trong chế độ SSL. Nếu bạn muốn khởi động Apache trong chế độ bình thường, xóa tùy chọn "-DSSL" gần dòng bạn đã thấy "daemon httpd".*

Ngôn ngữ script PHP4 server-side

Nếu bạn có ý dùng hỗ trợ ngôn ngữ phiên dịch PHP4 server-side với web server Apache đừng quên đưa vào trong tập tin "/etc/httpd/conf/httpd.conf" của bạn những dòng sau cho phép thực hiện điều này:

Bước 1

Thay đổi nội dung tập tin httpd.conf (vi /etc/httpd/conf/httpd.conf), và thêm những dòng sau giữa những đoạn <IfModule mod_mime.c> và </IfModule>:

```
AddType application/x-httpd-php .php
AddType application/x-httpd-php .php3
AddType application/x-httpd-php-source .phps
```

Bước 2

Bạn phải khởi động lại web server Apache để các thay đổi có hiệu lực:

- Để khởi động lại Apache, dùng những lệnh:

```
[root@deep /]# /etc/rc.d/init.d/httpd restart
Shutting down http: [ OK ]
Starting httpd: [ OK ]
```

Bước 3

Một khi những dòng ở trên được đưa vào trong tập tin "httpd.conf", chúng ta phải kiểm tra đặc điểm mới PHP4 để chắc rằng nó đang làm việc tốt. Chúng ta sẽ tạo một tập tin nhỏ tên "php.php" trong DocumentRoot, và kế đến trỏ trình duyệt web của chúng ta đến tài liệu PHP này để biết PHP4 có hoạt động trên server hay không.

Tạo tập tin php.php trong DocumentRoot của bạn:

(touch /home/httpd/ona/php.php) và thêm vào những dòng sau trong tập tin PHP :

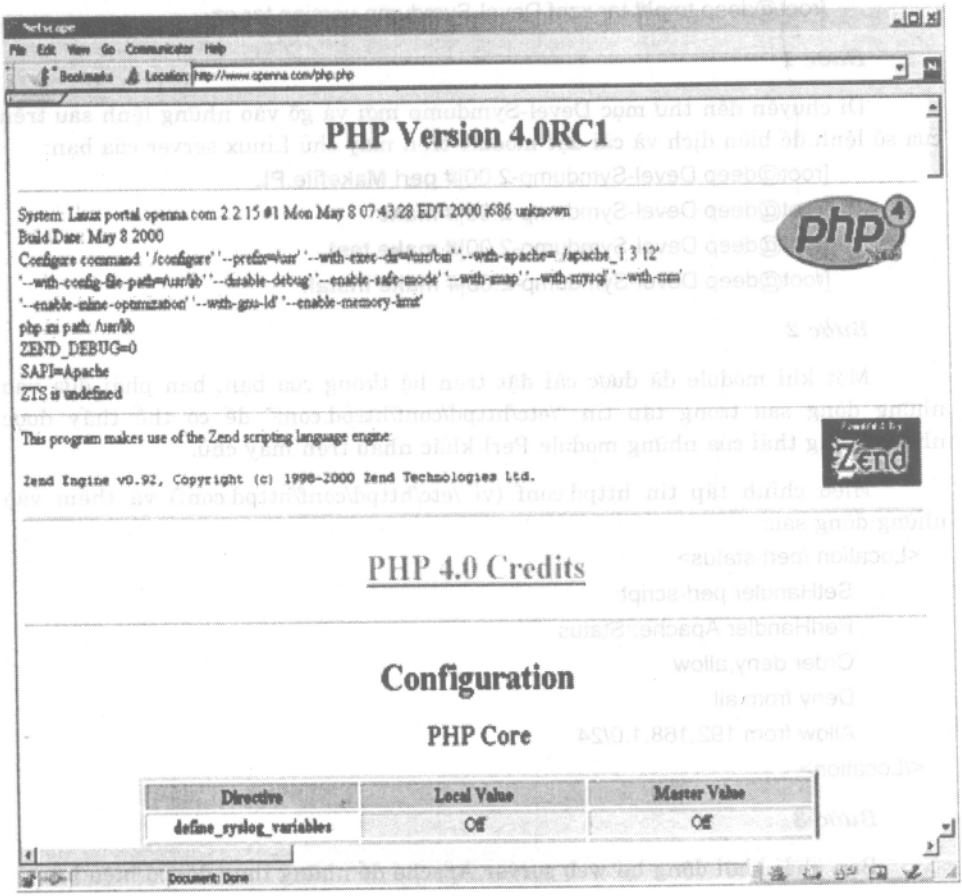
```
<body bgcolor="#FFFFFF">
<?php phpinfo()?>
</body>
```

Chú ý : *Những dòng này sẽ làm cho chương trình PHP4 biết để hiển thị những thông tin khác về cấu hình của máy chủ Linux.*

Bước 4

Bây giờ, trỏ trình duyệt web của bạn đến địa chỉ web sau: http://my-web-server/php.php

<my-web-server> là địa chỉ mà ở đó web server Apache đang hoạt động, và <php.php> là tập tin script PHP chúng ta đã tạo ở trên để hiển thị thông tin cấu hình của máy chủ Linux server của bạn.



Nếu bạn thấy mọi thứ hiển thị giống như trang ở trên trong trình duyệt web của bạn. Chúc mừng! Module PHP của bạn đã hoạt động.

Môđun Perl Devel::Symdump

Nếu bạn định dùng hỗ trợ ngôn ngữ lập trình mod_perl với web server Apache, nó có thể cài đặt module chương trình nhỏ perl tên là "Devel::Symdump". Đây là module của hãng thứ ba sẽ cho phép bạn kiểm tra bảng ký hiệu của perl và hệ thống các lớp (class) trong chương trình đang chạy. Để xây dựng và cài đặt nó, theo những bước sau :

Packages

Trang chủ Devel-Symdump : http://www.perl.com/CPAN/modules/by-module/Devel/
Bạn phải chắc tải về : Devel-Symdump-2_00_tar.gz
Phiên bản Devel-Symdump là 2.00

```
[root@deep /]# cp Devel-Symdump-version.tar.gz /var/tmp/
[root@deep /]# cd /var/tmp/
[root@deep tmp]# tar xzpf Devel-Symdump-version.tar.gz
```

Bước 1

Di chuyển đến thư mục Devel-Symdump mới và gõ vào những lệnh sau trên cửa sổ lệnh để biên dịch và cài đặt module trên máy chủ Linux server của bạn:

```
[root@deep Devel-Symdump-2.00]# perl Makefile.PL
[root@deep Devel-Symdump-2.00]# make
[root@deep Devel-Symdump-2.00]# make test
[root@deep Devel-Symdump-2.00]# make install
```

Bước 2

Một khi module đã được cài đặt trên hệ thống của bạn, bạn phải đưa vào những dòng sau trong tập tin "/etc/httpd/conf/httpd.conf" để có thể thấy được những trạng thái của những module Perl khác nhau trên máy chủ.

Hiệu chỉnh tập tin httpd.conf (vi /etc/httpd/conf/httpd.conf) và thêm vào những dòng sau:

```
<Location /perl-status>
    SetHandler perl-script
    PerlHandler Apache::Status
    Order deny,allow
    Deny from all
    Allow from 192.168.1.0/24
</Location>
```

Bước 3

Bạn phải khởi động lại web server Apache để những thay đổi có hiệu lực:

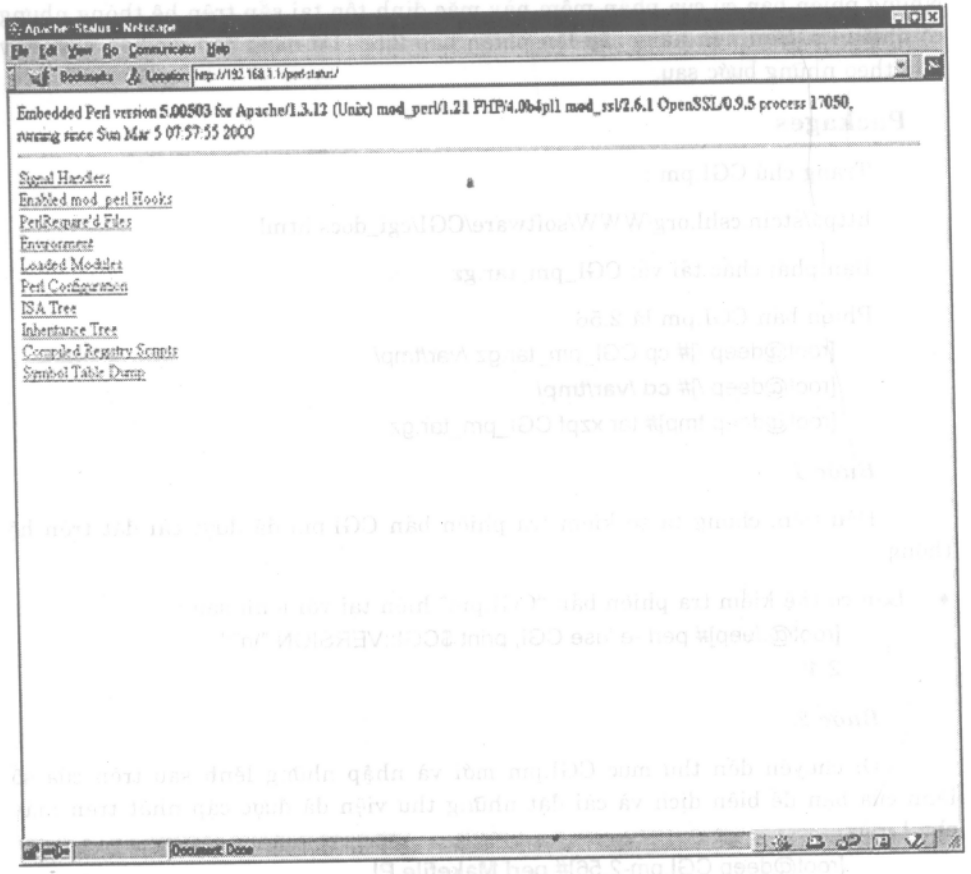
- Để khởi động lại Apache, dùng những lệnh sau :

```
[root@deep /]# /etc/rc.d/init.d/httpd restart
Shutting down http: [ OK ]
Starting httpd: [ OK ]
```

Bước 4

Cuối cùng, phải kiểm tra module mới Devel-Symdump để chắc rằng chúng ta có thể thấy được các trạng thái của những module Perl khác nhau trên máy chủ.

Để thẩm tra là các module này làm việc, trỏ web browser của bạn đến địa chỉ web: http://my-web-server/perl-status/. Ở đây <my-web-server> là địa chỉ ở đó web server Apache hoạt động.



Dọn dẹp lại sau quá trình cài đặt :

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf Devel-Symdump.version/Devel-Symdump-version.tar.gz
```

Những tập tin được cài đặt

```
> /usr/lib/perl5/man/man3/Devel::Symdump.3
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Devel
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Devel/Symdump
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Devel/Symdump/.packlist
> /usr/lib/perl5/site_perl/5.005/Devel
> /usr/lib/perl5/site_perl/5.005/Devel/Symdump
> /usr/lib/perl5/site_perl/5.005/Devel/Symdump/Export.pm
> /usr/lib/perl5/site_perl/5.005/Devel/Symdump.pm
```

Thư viện Perl CGI.pm

CGI.pm là thư viện Perl5 cho việc tạo những script CGI World Wide Web. Những phiên bản cũ của phần mềm này mặc định tồn tại sẵn trên hệ thống nhưng có nhiều lỗi. Bạn nên nâng cấp lên phiên bản 2.56. Để nâng cấp module này, hãy làm theo những bước sau.

Packages

Trang chủ CGI.pm :

http://stein.cshl.org/WWW/software/CGI/cgi_docs.html

Bạn phải chắc tải về: CGI_pm_tar.gz

Phiên bản CGI.pm là 2.56

```
[root@deep /]# cp CGI_pm_tar.gz /var/tmp/
[root@deep /]# cd /var/tmp/
[root@deep tmp]# tar xzpf CGI_pm_tar.gz
```

Bước 1

Đầu tiên, chúng ta sẽ kiểm tra phiên bản CGI.pm đã được cài đặt trên hệ thống.

- Bạn có thể kiểm tra phiên bản "CGI.pm" hiện tại với lệnh sau :

```
[root@deep]# perl -e 'use CGI; print $CGI::VERSION."\n";'
2.46
```

Bước 2

Di chuyển đến thư mục CGI.pm mới và nhập những lệnh sau trên cửa sổ lệnh của bạn để biên dịch và cài đặt những thư viện đã được cập nhật trên máy chủ Linux.

```
[root@deep CGI.pm-2.56]# perl Makefile.PL
[root@deep CGI.pm-2.56]# make
[root@deep CGI.pm-2.56]# make test
[root@deep CGI.pm-2.56]# make install
```

Dọn dẹp lại sau quá trình cài đặt

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf CGI.pm-version/ CGI_pm_tar.gz
```

Những tập tin đã được cài đặt

```
> /usr/lib/perl5/5.00503/CGI/Pretty.pm
> /usr/lib/perl5/5.00503/i386-linux/auto/CGI
> /usr/lib/perl5/5.00503/i386-linux/auto/CGI/.packlist
> /usr/lib/perl5/man/man3/CGI::Pretty.3
```

An toàn cho Apache

Thay đổi vài quyền quan trọng trên các tập tin và thư mục cho web server server của bạn.

Khi bạn cài đặt Apache trên server của bạn, có một số tập tin và thư mục có quá nhiều quyền được xác lập quyền mặc nhiên. Chương trình nhị phân "httpd" chỉ có thể đọc bởi super-user "root", và có thể thi hành được bởi quyền owner, quyền nhóm, và những quyền khác cho việc bảo mật tốt hơn. Những thư mục "/etc/httpd/conf" và "/var/log/httpd" không cần có quyền đọc, ghi hoặc thực thi bởi người khác.

```
[root@deep /]# chmod 511 /usr/sbin/httpd
[root@deep /]# chmod 750 /etc/httpd/conf/
[root@deep /]# chmod 750 /var/log/httpd/
```

Chỉ mục tự động

Nếu bạn cho phép chỉ mục tự động của những thư mục trong tập tin cấu hình Apache, (IndexOptions trong httpd.conf), bạn sẽ đưa ra một chính sách bảo mật, bởi vì bất kỳ những yêu cầu nào cho một thư mục mà không tìm một tập tin index sẽ xây dựng một chỉ mục của những gì trong thư mục này. Trong nhiều trường hợp bạn chỉ có thể muốn mọi người thấy những tập tin mà bạn liên kết cụ thể. Để khoá việc này bạn cần loại bỏ quyền read từ thư mục DocumentRoot (nhưng không bao gồm những tập tin bên trong nó).

```
[root@deep /]# cd /home/httpd/
[root@deep httpd]# chmod 311 ona
[root@deep httpd]# ls -la

d-wx--x--x 13 webadmin webadmin 1024 Jul 28 08:12 ona
```

Bây giờ, với thay đổi này, bất kỳ yêu cầu nào dành cho thư mục đã được bảo vệ sẽ trả về một thông báo lỗi:

```
Forbidden
You don't have permission to access "/ona/" on this server.
```

***Chú ý** : "ona" là DocumentRoot trong ví dụ của chúng ta.*

Tạo tập tin mật khẩu .dbmpasswd cho việc kiểm tra tính xác thực những người dùng

Bước này chỉ cần thiết nếu bạn nghĩ sẽ dùng một tập tin kiểm tra tính xác thực truy cập hệ thống cho trang web của bạn. Tập tin xác thực truy cập được dùng khi bạn cần bảo vệ một số phần trong trang web của bạn bằng mật khẩu. Đối với Apache, hiện có nhiều chọn lựa để bảo vệ trang web với các username và password.

Bước 1

Chương trình tiện ích "dbmmanage" của Apache có thể được dùng để tạo và cập nhật những tên người sử dụng và mật khẩu của những người sử dụng HTTP. Phương thức này sử dụng một tập tin dạng DBM, đây là kỹ thuật nhanh nhất khi bạn có hàng ngàn người dùng để quản lý tập tin mật khẩu của bạn. Điều quan trọng trước tiên thay đổi quyền của chương trình này trở thành (0750/-rwxr-x---), quyền ghi chỉ được sử dụng bởi super-user "root", quyền đọc và quyền thực thi được thực hiện bởi nhóm và không có quyền nào cho những người khác.

- Để thay đổi quyền trên chương trình "dbmmanage", dùng lệnh sau :

```
[root@deep /]# chmod 750 /usr/bin/dbmmanage
```

- Để tạo tên người dùng và mật khẩu, dùng lệnh sau :

```
[root@deep /]# /usr/bin/dbmmanage /etc/httpd/.dbmpasswd adduser \
username
New password:
Re-type new password:
User username added with password encrypted to l4jrdAL9MH0K.
```

Ở đây </etc/httpd> là nơi lưu trữ tập tin mật khẩu, <.dbmpasswd> là tên của tập tin mật khẩu, và <username> là tên của người dùng mà bạn muốn thêm vào trong tập tin ".dbmpasswd".

Bước 2

Nếu bạn sử dụng tiện ích "dbmmanage" với web server Apache để tạo các mật khẩu và tên người dùng, đừng quên đưa vào trong tập tin cấu hình "/etc/httpd/conf/httpd.conf" của bạn phần của trang web mà bạn cần bảo vệ bằng xác thực mật khẩu người dùng.

Soạn thảo tập tin httpd.conf (vi /etc/httpd/conf/httpd.conf) và thêm vào những dòng sau để bảo vệ thư mục "private" của trang web "ona" của bạn bằng xác thực mật khẩu người dùng :

```
<Directory "/home/httpd/ona/private">
        Options None
        AllowOverride AuthConfig
        AuthName "restricted stuff"
        AuthType Basic
        AuthDBUserFile /etc/httpd/.dbmpasswd
        require valid-user
</Directory>
```

Đường dẫn </home/httpd/ona/private> chỉ rõ thư mục bạn muốn bảo vệ bằng mật khẩu và tên người dùng, </etc/httpd/.dbmpasswd> chỉ rõ nơi lưu trữ tập tin mật khẩu.

Chú ý : *Để thêm vào module xác thực mật khẩu Db đến web server Apache, bạn phải chắc chắn rằng đã đưa vào nó trong suốt thời gian cấu hình Apache với tham số "--add-module = src/modules/standard/mod_auth_db.c". Đọc nhiều thông tin hơn trong tài liệu Apache.*

Bước 3

Khởi động lại web server Apache để những thay đổi có hiệu lực :

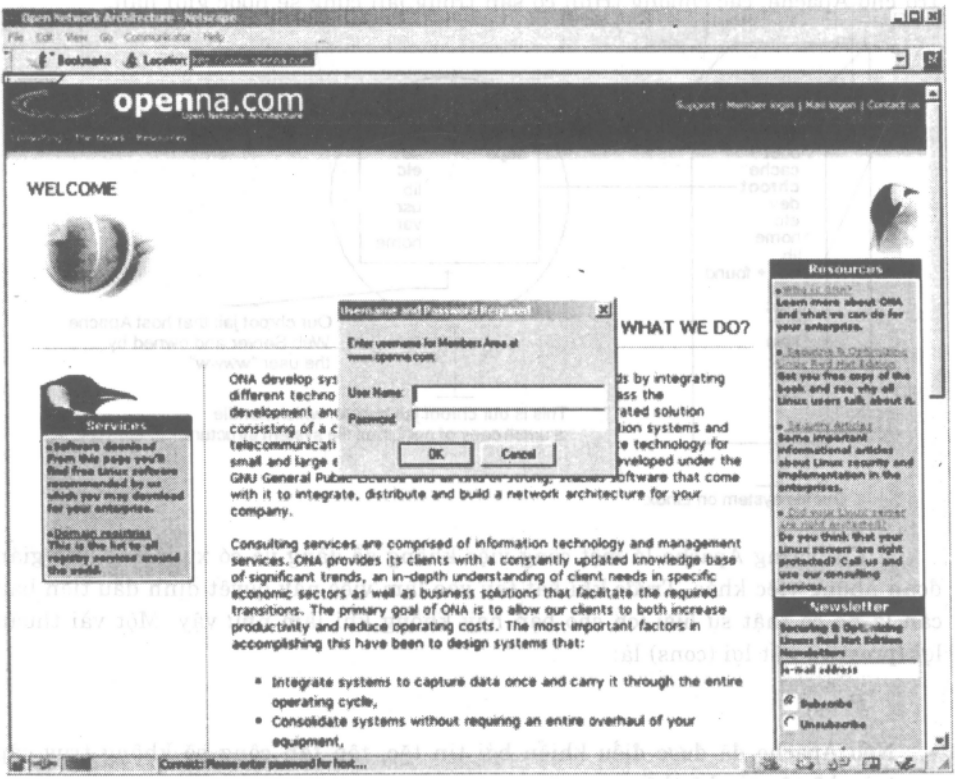
- Để khởi động lại Apache, dùng những lệnh sau :

```
[root@deep /]# /etc/rc.d/init.d/httpd restart
Shutting down http: [ OK ]
Starting httpd: [ OK ]
```

Bước 4

Cuối cùng chúng ta phải kiểm tra thử thư mục mới đã được bảo vệ tên là (private).

Để chắc rằng nó hoạt động, truy cập đến địa chỉ sau : http://my-web-server/private/. <my-web-server> là địa chỉ mà ở đó web server Apache của bạn tồn tại. </private> là thư mục chúng ta muốn bảo vệ bằng xác thực mật khẩu người dùng.



Làm mất tác dụng việc thay đổi tập tin cấu hình quan trọng như "httpd.conf"

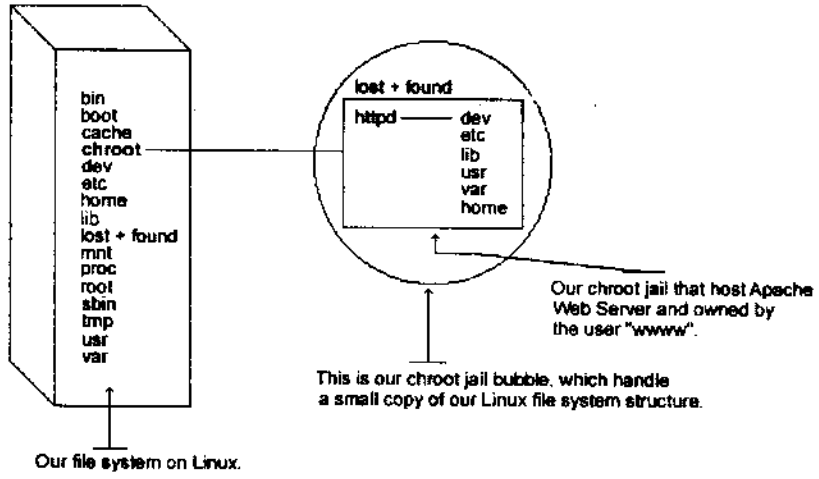
Như chúng ta đã biết, bit không thay đổi dùng để ngăn không cho xóa, ghi đè hoặc tạo biểu tượng liên kết đến tập tin. Một khi tập tin "httpd.conf" đã được cấu hình, đây là ý tốt để làm mất tác dụng nó bằng dòng lệnh sau:

```
[root@deep /]# chattr +i /etc/httpd/conf/httpd.conf
```

Chạy Apache trong một vùng riêng biệt chroot (jail chroot)

Phần này tập trung vào việc ngăn chặn Apache như một điểm để các tin tặc đi vào hệ thống của bạn. Mặc định Apache được chạy với account non-root, điều này sẽ giới hạn bất kỳ sự thiệt hại có thể gây ra bởi người sử dụng thông thường with a local shell. Dĩ nhiên, chúng ta cũng yêu cầu vấn đề bảo mật cho web server Apache dù nó đang chạy với account bất kỳ guest, nobody. Với nhu cầu này, một bước chúng ta cần làm thêm là chạy Apache trong một vùng riêng biệt chroot **(chroot jail)**.

Tiện ích chính của một chroot jail là nơi riêng biệt jail sẽ giới hạn phần của tập tin hệ thống nhìn thấy thư mục root của jail. Thêm nữa bởi vì jail chỉ cần hỗ trợ cho Apache, các chương trình có sẵn trong jail cũng sẽ được giới hạn.



Chrooting Apache là một công việc không dễ dàng và có xu hướng làm gián đoạn những việc khác. Trước khi bắt tay vào làm việc này quyết định đầu tiên bạn cần là nó có thật sự hữu ích cho bạn hay không khi làm như vậy. Một vài thuận lợi (pros) và bất lợi (cons) là:

Pros:

- Nếu Apache đã được điều khiển bởi tin tặc, tên tấn công sẽ không truy cập toàn bộ tập tin hệ thống.

- Những đoạn CGI được viết kém cỏi có thể cho phép một số người truy cập đến server sẽ không làm việc.

Cons:

- Có thêm một số thư viện bạn sẽ cần trong chroot jail để Apache hoạt động.
- Nếu bạn dùng bất kỳ những đặc trưng Perl/CGI với Apache, bạn sẽ cần sao chép lại những tập tin nhị phân, những thư viện Perl và những tập tin đến nơi thích hợp trong giới hạn khoảng trống chroot. Tương tự chúng ta áp dụng cho SSL, PHP, LDAP, PostgressSQL và những chương trình của hãng thứ ba khác.

Cấu hình chrooted được liệt kê dưới giả sử rằng bạn đã biên dịch máy chủ Apache với chương trình bên ngoài mod_ssl. Những điều khác nhau mà bạn biên dịch mod_ssl với web server Apache được lưu ở nơi nào đó, sau đó bạn cần copy những thư viện và những tập tin nhị phân liên quan vào trong thư mục chrooted.

Nên nhớ rằng nếu bạn đã biên dịch Apache có sử dụng mod_perl, bạn phải copy tất cả những tập tin nhị phân và những thư viện Perl có liên quan đến thư mục chrooted. Perl được lưu trữ trong "/usr/lib/perl5" và trong trường hợp bạn sử dụng những đặc trưng của Perl, copy những thư mục Perl đến "/chroot/httpd/usr/lib/perl5". Đừng quên tạo thư mục "/chroot/httpd/usr/lib/perl5" trong cấu trúc chrooted trước khi copy.

Theo sau là những bước cần thiết để chạy Web server Apache trong một chroot jail :

Bước 1 :

Chúng ta phải tìm thư viện phụ thuộc được chia sẻ của httpd. Những tập tin này sẽ cần được copy vào trong chroot jail sau đó.

- Để tìm thư viện phụ thuộc chia sẻ của httpd, thi hành lệnh sau :

```
[root@deep /]# ldd /usr/sbin/httpd
libpam.so.0 => /lib/libpam.so.0 (0x40016000)
libm.so.6 => /lib/libm.so.6 (0x4001f000)
libdl.so.2 => /lib/libdl.so.2 (0x4003b000)
libcrypt.so.1 => /lib/libcrypt.so.1 (0x4003e000)
libnsl.so.1 => /lib/libnsl.so.1 (0x4006b000)
libresolv.so.2 => /lib/libresolv.so.2 (0x40081000)
libdb.so.3 => /lib/libdb.so.3 (0x40090000)
libc.so.6 => /lib/libc.so.6 (0x400cb000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Chú ý : *Các tập tin được liệt kê trên bạn sẽ cần trong những bước tiếp theo.*

Bước 2

Thêm một UID và một GID mới, nếu điều này chưa làm sẵn để chạy httpd Apache. Đây là điều quan trọng bởi vì chạy Apache như root sẽ không đúng mục đích của jail, và dùng một UID khác đã có trên hệ thống (ví dụ: nobody) có thể cho phép những dịch vụ của bạn cùng truy cập đến những tài nguyên khác. Hãy suy nghĩ đến viễn cảnh mà ở đó web server đang chạy như nobody, hoặc bất kỳ UID/GID hợp lệ khác đã có sẵn trên hệ thống. Kẻ phá hoại có thể can thiệp bất kỳ tiến trình nào chạy như nobody từ bên trong chroot.

Kiểm tra những tập tin "/etc/passwd" và "/etc/group" cho một số tự chọn UID/GID. Trong cấu hình này chúng ta sẽ dùng giá trị số "80" và UID/GID là "WWW".

```
[root@deep /]# useradd -c "Apache Server" -u 80 -s /bin/false -r –d \
/home/httpd www 2>/dev/null || :
```

Những lệnh trên sẽ tạo nhóm "WWW" với trị số GID là 80, và người dùng "WWW" với trị số UID là 80.

Bước 3

Cài đặt môi trường chroot. Trước tiên cần tạo cấu trúc chrooted Apache. Chúng ta dùng "/chroot/httpd" cho chrooted Apache. "/chroot/httpd" là một thư mục lưu trữ trên một partition khác, nơi chúng ta đã quyết định đặt Apache cho an toàn hơn.

```
[root@deep /]# /etc/rc.d/init.d/httpd stop (Chỉ nếu Apache đã được cài đặt
và chạy trên hệ thống của bạn).
Shutting down http: [ OK ]
[root@deep /]# mkdir /chroot/httpd
```

Kế đến, tạo những thư mục còn lại như sau:

```
[root@deep /]# mkdir /chroot/httpd/dev
[root@deep /]# mkdir /chroot/httpd/lib
[root@deep /]# mkdir /chroot/httpd/etc
[root@deep /]# mkdir -p /chroot/httpd/usr/sbin
[root@deep /]# mkdir -p /chroot/httpd/var/run
[root@deep /]# mkdir -p /chroot/httpd/var/log/httpd
[root@deep /]# chmod 750 /chroot/httpd/var/log/httpd/
[root@deep /]# mkdir -p /chroot/httpd/home/httpd
```

Copy thư mục cấu hình chính, các tập tin cấu hình, thư mục cgi-bin, thư mục root và chương trình httpd đến chroot jail:

```
[root@deep /]# cp -r /etc/httpd /chroot/httpd/etc/
[root@deep /]# cp -r /home/httpd/cgi-bin /chroot/httpd/home/httpd/
[root@deep /]# cp -r /home/httpd/your-DocumentRoot /chroot/httpd/home/httpd/
```

```
[root@deep /]# mknod /chroot/httpd/dev/null c 1 3
[root@deep /]# chmod 666 /chroot/httpd/dev/null
[root@deep /]# cp /usr/sbin/httpd /chroot/httpd/usr/sbin/
```

Chúng ta cần những thư mục

"/chroot/httpd/etc","/chroot/httpd/dev","/chroot/httpd/lib",

"/chroot/httpd/usr/sbin","/chroot/httpd/var/run",

"/chroot/httpd/home/httpd" và "/chroot/httpd/var/log/httpd" bởi vì, từ thư mục chroot, xem như chúng ta đang ở tại thư mục gốc '/'.

Bước 4

Nếu đã biên dịch web server Apache với sự hỗ trợ SSL, bạn phải copy toàn bộ thư mục "/etc/ssl" cùng với tất cả các tiến trình xử lý những khoá chung (public key) và khoá riêng (private key) đến chroot jail.

```
[root@deep /]# cp -r /etc/ssl /chroot/httpd/etc/
[root@deep /]# chmod 600 /chroot/httpd/etc/ssl/certs/ca.crt
[root@deep /]# chmod 600 /chroot/httpd//etc/ssl/certs/server.crt
[root@deep /]# chmod 600 /chroot/httpd/etc/ssl/private/ca.key
[root@deep /]# chmod 600 /chroot/httpd/etc/ssl/private/server.key
```

Bước 5

Bởi vì biên dịch Apache để dùng những thư viện dùng chung, chúng ta cần cài đặt chúng vào trong cấu trúc thư mục chroot. Dùng lệnh **ldd /chroot/httpd/usr/sbin/httpd** tìm ra những thư viện cần thiết. Đưa ra (dựa trên những gì bạn đã biên dịch với Apache) màn hình kết quả tương tự như sau:

```
libpam.so.0 => /lib/libpam.so.0 (0x40016000)
libm.so.6 => /lib/libm.so.6 (0x4001f000)
libdl.so.2 => /lib/libdl.so.2 (0x4003b000)
libcrypt.so.1 => /lib/libcrypt.so.1 (0x4003e000)
libnsl.so.1 => /lib/libnsl.so.1 (0x4006b000)
libresolv.so.2 => /lib/libresolv.so.2 (0x40081000)
libdb.so.3 => /lib/libdb.so.3 (0x40090000)
libc.so.6 => /lib/libc.so.6 (0x400cb000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Sao lại những thư viện dùng chung đã nhận biết ở trên :

```
[root@deep /]# cp /lib/libpam.so.0 /chroot/httpd/lib/
[root@deep /]# cp /lib/libm.so.6 /chroot/httpd/lib/
[root@deep /]# cp /lib/libdl.so.2 /chroot/httpd/lib/
[root@deep /]# cp /lib/libcrypt.so.1 /chroot/httpd/lib/
[root@deep /]# cp /lib/libnsl* /chroot/httpd/lib/
[root@deep /]# cp /lib/libresolv* /chroot/httpd/lib/
```

```
[root@deep /]# cp /lib/libdb.so.3 /chroot/httpd/lib/
[root@deep /]# cp /lib/libc.so.6 /chroot/httpd/lib/
[root@deep /]# cp /lib/ld-linux.so.2 /chroot/httpd/lib/
```

Bạn sẽ cần những thư viện đặc biệt cho một vài chức năng mạng, giống như sau :

```
[root@deep /]# cp /lib/libnss_compat* /chroot/httpd/lib/
[root@deep /]# cp /lib/libnss_dns* /chroot/httpd/lib/
[root@deep /]# cp /lib/libnss_files* /chroot/httpd/lib/
```

Bước 6

Bây giờ chúng ta cần copy các tập tin passwd và group vào trong thư mục "/chroot/httpd/etc" của chrooted. Khái niệm ở đây giống như ftpd sử dụng những tập tin passwd và group. Kế đến, xóa tất cả những gì Apache không cần đến.

```
[root@deep /]# cp /etc/passwd /chroot/httpd/etc/
[root@deep /]# cp /etc/group /chroot/httpd/etc/
```

Thay đổi tập tin passwd (vi /chroot/httpd/etc/passwd) và xóa tất cả những mục ngoại trừ entry cho người dùng chạy Apache như trong cấu hình của chúng ta, nó là "WWW":

```
WWW:x:80:80::/home/WWW:/bin/bash
```

Thay đổi tập tin group (vi /chroot/httpd/etc/group) và xóa tất cả những mục ngoại trừ group Apache chạy như (trong cấu hình chúng ta, nó là "WWW"):

```
www:x:80
```

Bước 7

Bạn cũng sẽ cần những tập tin "/etc/resolv.conf", "/etc/nsswith.conf" và "/etc/hosts" trong chroot jail.

```
[root@deep /]# cp /etc/resolv.conf /chroot/httpd/etc/
[root@deep /]# cp /etc/hosts /chroot/httpd/etc/
[root@deep /]# cp /etc/nsswitch.conf /chroot/httpd/etc/
```

Bước 8

Bây giờ chúng ta phải thiết lập đặc quyền vài tập tin trong thư mục chroot jail trở thành không thể thay đổi cho việc bảo mật tốt hơn.

- Thiết lập bit không thể thay đổi trong tập tin "passwd" :

```
[root@deep /]# cd /chroot/httpd/etc/
[root@deep /]# chattr +i passwd
```

- Thiết lập bit không thể thay đổi trong tập tin "group":

```
[root@deep /]# cd /chroot/httpd/etc/
[root@deep /]# chattr +i group
```

- Thiết lập bit không thể thay đổi trong tập tin "httpd.conf" :

```
[root@deep /]# cd /chroot/httpd/etc/httpd/conf/
```

```
[root@deep /]# chattr +i httpd.conf
```

- Thiết lập bit không thể thay đổi trong tập tin "resolv.conf" :

```
[root@deep /]# cd /chroot/httpd/etc/
[root@deep /]# chattr +i resolv.conf
```

- Thiết lập bit không thể thay đổi trong tập tin "hosts" :

```
[root@deep /]# cd /chroot/httpd/etc/
[root@deep /]# chattr +i hosts
```

- Để bit an toàn trong tập tin "nsswich.conf":

```
[root@deep /]# cd /chroot/httpd/etc/
[root@deep /]# chattr +i nsswitch.conf
```

Bước 9

Copy tập tin "localtime" đến jail như vậy toàn bộ logfile được điều chỉnh theo chính xác timezone nội bộ của chúng ta:

```
[root@deep /]# cp /etc/localtime /chroot/httpd/etc/
```

Bước 10

Xóa những tập tin và thư mục Apache không cần thiết:

```
[root@deep /]# rm -rf /var/log/httpd/
[root@deep /]# rm -rf /etc/httpd/
[root@deep /]# rm -rf /home/httpd/
[root@deep /]# rm -f /usr/sbin/httpd
```

Chúng ta có thể xóa tất cả những tập tin và thư mục ở trên an toàn bởi vì chúng được lưu trữ tập tin và thư mục này trong chroot jail.

Bước 11

Cho syslogd biết về dịch vụ chrooted mới.

Thông thường, những tiến trình cho syslog biết thông qua "/dev/log". Như kết quả của chroot jail, điều này sẽ không thực hiện được, do đó syslog cần được lắng nghe trên "/chroot/httpd/dev/log". Để làm điều này, hiệu chỉnh script khởi động syslog để chỉ ra thêm những nơi để syslog lắng nghe.

Thay đổi script syslog (vi /etc/rc.d/init.d/syslog) thay đổi dòng :

```
daemon syslogd -m 0
```

Trở thành:

```
daemon syslogd -m 0 -a /chroot/httpd/dev/log
```

Bước 12

Mặc định tập tin script Apache khởi động "httpd" daemon bên ngoài chroot jail. Chúng ta phải thay đổi nó để khởi động httpd từ chroot jail. Thay đổi script httpd (vi /etc/rc.d/init.d/httpd) và thay đổi dòng sau :

daemon httpd

Trở thành:

/usr/sbin/chroot /chroot/httpd/ /usr/sbin/httpd -DSSL

rm -f /var/run/httpd.pid

Trở thành:

rm -f /chroot/httpd/var/run/httpd.pid

Bước 13

Cuối cùng, chúng ta phải kiểm tra cấu hình chrooted jail mới của Web server Apache.

- Đầu tiên là khởi động lại syslog daemon với lệnh sau :

```
[root@deep /]# /etc/rc.d/init.d/syslog restart
Shutting down kernel logger: [ OK ]
Shutting down system logger: [ OK ]
Starting system logger: [ OK ]
Starting kernel logger: [ OK ]
```

- Bây giờ, khởi động Apache chroot jail mới với lệnh:

```
[root@deep /]# /etc/rc.d/init.d/httpd start
Starting httpd: [ OK ]
```

- Nếu bạn không có lỗi nào, làm một lệnh **ps ax| grep httpd** và sẽ thấy như dưới đây nếu chúng đang chạy :

```
[root@deep /]# ps ax | grep httpd
14373 ? S 0:00 httpd -DSSL
14376 ? S 0:00 httpd -DSSL
14377 ? S 0:00 httpd -DSSL
14378 ? S 0:00 httpd -DSSL
14379 ? S 0:00 httpd -DSSL
14380 ? S 0:00 httpd -DSSL
14381 ? S 0:00 httpd -DSSL
14382 ? S 0:00 httpd -DSSL
14383 ? S 0:00 httpd -DSSL
14384 ? S 0:00 httpd -DSSL
14385 ? S 0:00 httpd -DSSL
14386 ? S 0:00 httpd -DSSL
14387 ? S 0:00 httpd -DSSL
14388 ? S 0:00 httpd -DSSL
14389 ? S 0:00 httpd -DSSL
14390 ? S 0:00 httpd -DSSL
14391 ? S 0:00 httpd -DSSL
```

```
14397 ? S 0:00 httpd -DSSL
14476 ? S 0:00 httpd -DSSL
14477 ? S 0:00 httpd -DSSL
14478 ? S 0:00 httpd -DSSL
```

Nếu kết quả như trên và để kiểm tra chắc chắn đúng là chrooted, bằng cách chọn ra một trong những số của tiến trình của chúng và chạy lệnh **ls –la /proc/that_process_number/root/.**

```
[root@deep /]# ls -la /proc/14373/root/
```

Nếu bạn thấy :

```
dev
etc
home
lib
usr
var
```

Chúc mừng ! bạn đã cài đặt đúng

Như đề cập ở trên, nếu bạn dùng Perl, bạn sẽ cần sao lại hoặc làm liên kết (link) bất kỳ các thư viện hệ thống, thư viện perl "/usr/lib/perl5", và những tập tin nhị phân vào trong vùng chroot. Làm công việc tương tự này cho những ứng dụng cho SSL, PHP, LDAP, PostgreSQL và những chương trình khác.

Cấu hình của tập tin "/etc/logrotate.d/apache" mới

Bây giờ những logfile của Apache được lưu trong thư mục "/chroot/var/log/httpd" thay vì "/var/log/httpd", với lý do này chúng ta cần phải sửa đổi tập tin "/etc/logrotate.d/httpd" trỏ tới thư mục mới chrooted. Cũng thế, chúng ta biên dịch Apache với mod_ssl, do đó chúng ta sẽ thêm một dòng để cho phép chương trình logrotate quay vòng tự động mỗi tuần những tập tin "ssl_request_log" và "ssl_engine_log". Cấu hình tập tin "/etc/logrotate.d/apache" để quay vòng những logfile của bạn tự động mỗi tuần như sau:

Tạo tập tin Apache (touch /etc/logrotate.d/apache) và thêm vào :

```
/chroot/httpd/var/log/httpd/access_log {
missingok
postrotate
/usr/bin/killall -HUP /chroot/httpd/usr/sbin/httpd
endscript
}

/chroot/httpd/var/log/httpd/error_log {
missingok
postrotate
```

```
/usr/bin/killall -HUP /chroot/httpd/usr/sbin/httpd
endscript
}

/chroot/httpd/var/log/httpd/ssl_request_log {
missingok
postrotate
/usr/bin/killall -HUP /chroot/httpd/usr/sbin/httpd
endscript
}

/chroot/httpd/var/log/httpd/ssl_engine_log {
missingok
postrotate
/usr/bin/killall -HUP /chroot/httpd/usr/sbin/httpd
endscript
}
```

Tối ưu Apache

Module mod_mmap_static của Apache

Có một module đặc biệt được phân bố với Apache tên là "mod_mmap_static" mà bạn có thể dùng để cải thiện khả năng thực hiện của web server của bạn. Module này hoạt động bằng cách cung cấp sự sắp xếp của danh sách cấu hình tĩnh của yêu cầu thường xuyên, nhưng không thay đổi, những tập tin trong RootDicrectory. Do vậy, nếu những tập tin được hiển thị bởi Apache sẽ không thường thay đổi, bạn có thể dùng module này để định trong bộ nhớ những tài liệu tĩnh và tăng tốc độ xử lý của web server Apache.

Điều quan trọng để biết rằng module mod_mmap_static của Apache phải được cho phép trong suốt thời gian cấu hình và biên dịch của Apache trước khi bạn có thể dùng nó. Nếu bạn tuân theo những gì đã diễn giải trong cấu hình và biên dịch như trên, thì module này đã có trong Apache (--add-mode-../mod_mmap_static.c).

Bước 1

Đặt những tài liệu tĩnh trong bản đồ vùng nhớ, dùng lệnh sau :

```
[root@deep /]# find /home/httpd/ona -type f -print | sed -e 's/.*/mmapfile &/' >
/etc/httpd/conf/mmap.conf
```

</home/httpd/ona> là RootDirectory, hoặc chính xác hơn, thư mục mà bạn sẽ cung cấp tài liệu của bạn và </etc/httpd/conf/mmap.conf> là nơi mà chúng ta muốn tạo tập tin này, tập tin "mmap.conf" bao gồm một bản đồ bộ nhớ tĩnh của tất cả những tài liệu lưu trữ dưới RootDirectory của chúng ta.

Bước 2

Một khi tập tin "mmap.conf" được tạo ở nơi lưu trữ mà ở đó chúng ta chọn để quản lý tập tin này, chúng ta phải đưa vào trong tập tin "httpd.conf" của Apache để có thể sử dụng những đặc trưng của nó trên web server của chúng ta.

Soạn thảo tập tin httpd.conf (vi /etc/httpd/conf/httpd.conf) và thêm vào dòng sau :

```
<IfModule mod_include.c>
Include conf/mmap.conf
</IfModule>
```

***Chú ý** : Xem tài liệu Apache sẽ cung cấp nhiều thông tin về cách sử dụng mod_mmap_static. Hãy nhớ rằng đặc trưng này chỉ được sử dụng khi bạn cung cấp những tài liệu không thay đổi thường xuyên trên trang web của bạn.*

Bước 3

Bạn phải khởi động lại web server cho thay đổi có tác dụng :

- Để khởi động lại Apache, dùng những câu lệnh sau :

```
[root@deep /]# /etc/rc.d/init.d/httpd restart
Shutting down http: [ OK ]
Starting httpd: [ OK ]
```

Các thuộc tính atime và noatime

Các thuộc tính atime và noatime của Linux có thể được dùng để đo lường khả năng thực hiện trong Apache. Hãy tìm hiểu chương "Tổng quan về tối ưu hệ thống" để có thêm nhiều thông tin.

Những tập tin được cài đặt cho web server Apache

> /etc/rc.d/init.d/httpd
> /etc/rc.d/rc0.d/K15httpd
> /etc/rc.d/rc1.d/K15httpd
> /etc/rc.d/rc2.d/K15httpd
> /etc/rc.d/rc3.d/S85httpd
> /etc/rc.d/rc4.d/S85httpd
> /etc/rc.d/rc5.d/S85httpd
> /etc/rc.d/rc6.d/K15httpd
> /etc/logrotate.d/apache
> /etc/httpd
> /etc/httpd/conf
> /etc/httpd/conf/httpd.conf.default
> /etc/httpd/conf/httpd.conf

> /usr/include/apache/ap_md5.h
> /usr/include/apache/ap_mm.h
> /usr/include/apache/ap_mmn.h
> /usr/include/apache/ap_sha1.h
> /usr/include/apache/buff.h
> /usr/include/apache/compat.h
> /usr/include/apache/conf.h
> /usr/include/apache/explain.h
> /usr/include/apache/fnmatch.h
> /usr/include/apache/hsregex.h
> /usr/include/apache/http_conf_globals.h
> /usr/include/apache/http_config.h
> /usr/include/apache/http_core.h

> /etc/httpd/conf/mime.types.default
> /etc/httpd/conf/mime.types
> /etc/httpd/conf/magic.default
> /etc/httpd/conf/magic
> /etc/httpd/php.ini
> /home/httpd
> /home/httpd/cgi-bin
> /home/httpd/cgi-bin/printenv
> /home/httpd/cgi-bin/test-cgi
> /usr/bin/htpasswd
> /usr/bin/htdigest
> /usr/bin/dbmmanage
> /usr/include/apache
> /usr/include/apache/xml
> /usr/include/apache/xml/asciitab.h
> /usr/include/apache/xml/hashtable.h
> /usr/include/apache/xml/iasciitab.h
> /usr/include/apache/xml/latin1tab.h
> /usr/include/apache/xml/nametab.h
> /usr/include/apache/xml/utf8tab.h
> /usr/include/apache/xml/xmldef.h
> /usr/include/apache/xml/xmlparse.h
> /usr/include/apache/xml/xmlrole.h
> /usr/include/apache/xml/xmltok.h
> /usr/include/apache/xml/xmltok_impl.h
> /usr/include/apache/alloc.h
> /usr/include/apache/ap.h
> /usr/include/apache/ap_compat.h
> /usr/include/apache/ap_config.h
> /usr/include/apache/ap_config_auto.h
> /usr/include/apache/ap_ctx.h
> /usr/include/apache/ap_ctype.h
> /usr/include/apache/ap_hook.h
> /usr/include/apache/http_log.h
> /usr/include/apache/http_main.h
> /usr/include/apache/http_protocol.h
> /usr/include/apache/http_request.h
> /usr/include/apache/http_vhost.h
> /usr/include/apache/httpd.h
> /usr/include/apache/multithread.h
> /usr/include/apache/rfc1413.h
> /usr/include/apache/scoreboard.h
> /usr/include/apache/util_date.h
> /usr/include/apache/util_md5.h
> /usr/include/apache/util_script.h
> /usr/include/apache/util_uri.h
> /usr/include/apache/os.h
> /usr/include/apache/os-inline.c
> /usr/lib/apache
> /usr/man/man1/htpasswd.1
> /usr/man/man1/htdigest.1
> /usr/man/man1/dbmmanage.1
> /usr/man/man8/ab.8
> /usr/man/man8/httpd.8
> /usr/man/man8/logresolve.8
> /usr/man/man8/rotatelogs.8
> /usr/man/man8/apxs.8
> /usr/sbin/httpd
> /usr/sbin/ab
> /usr/sbin/logresolve
> /usr/sbin/rotatelogs
> /usr/sbin/apxs
> /var/log/httpd
> /var/cache
> /var/cache/httpd

Những tập tin được cài đặt cho scripting language PHP4 với web server Apache

> /usr/bin/phpize
> /usr/bin/php-config
>/usr/include/php/ext/standard/php_browscap.h
> /usr/include/php/ext/standard/php_crypt.h

> /usr/include/php
> /usr/include/php/Zend
> /usr/include/php/Zend/FlexLexer.h
> /usr/include/php/Zend/acconfig.h
> /usr/include/php/Zend/modules.h
> /usr/include/php/Zend/zend-parser.h
> /usr/include/php/Zend/zend-scanner.h
> /usr/include/php/Zend/zend.h
> /usr/include/php/Zend/zend_API.h
> /usr/include/php/Zend/zend_alloc.h
> /usr/include/php/Zend/zend_builtin_functions.h
> /usr/include/php/Zend/zend_compile.h
> /usr/include/php/Zend/zend_config.h
> /usr/include/php/Zend/zend_config.w32.h
> /usr/include/php/Zend/zend_constants.h
> /usr/include/php/Zend/zend_dynamic_array.h
> /usr/include/php/Zend/zend_errors.h
> /usr/include/php/Zend/zend_execute.h
> /usr/include/php/Zend/zend_execute_locks.h
> /usr/include/php/Zend/zend_extensions.h
> /usr/include/php/Zend/zend_fast_cache.h
> /usr/include/php/Zend/zend_globals.h
> /usr/include/php/Zend/zend_globals_macros.h
> /usr/include/php/Zend/zend_hash.h
> /usr/include/php/Zend/zend_highlight.h
> /usr/include/php/Zend/zend_indent.h
> /usr/include/php/Zend/zend_list.h
> /usr/include/php/Zend/zend_llist.h
> /usr/include/php/Zend/zend_operators.h
> /usr/include/php/Zend/zend_ptr_stack.h
> /usr/include/php/Zend/zend_stack.h
> /usr/include/php/Zend/zend_variables.h
> /usr/include/php/TSRM
> /usr/include/php/TSRM/TSRM.h
> /usr/include/php/ext
> /usr/include/php/ext/standard
> /usr/include/php/ext/standard/base64.h
> /usr/include/php/ext/standard/basic_functions.h
> /usr/include/php/ext/standard/cyr_convert.h
> /usr/include/php/ext/standard/php_dir.h
> /usr/include/php/ext/standard/php_filestat.h
> /usr/include/php/ext/standard/php_image.h
> /usr/include/php/ext/standard/php_iptc.h
> /usr/include/php/ext/standard/php_lcg.h
> /usr/include/php/ext/standard/php_link.h
> /usr/include/php/ext/standard/php_mail.h
> /usr/include/php/ext/standard/php_metaphone.h
> /usr/include/php/ext/standard/php_output.h
> /usr/include/php/ext/standard/php_rand.h
> /usr/include/php/ext/standard/php_standard.h
> /usr/include/php/ext/standard/php_string.h
> /usr/include/php/ext/standard/php_syslog.h
> /usr/include/php/ext/standard/php_var.h
> /usr/include/php/ext/standard/phpdir.h
> /usr/include/php/ext/standard/phpmath.h
> /usr/include/php/ext/standard/quot_print.h
> /usr/include/php/ext/standard/reg.h
> /usr/include/php/ext/standard/type.h
> /usr/include/php/ext/standard/uniqid.h
> /usr/include/php/ext/standard/url.h
> /usr/include/php/ext/standard/url_scanner.h
> /usr/include/php/regex
> /usr/include/php/regex/regex.h
> /usr/include/php/regex/regex_extra.h
> /usr/include/php/php.h
> /usr/include/php/php_regex.h
> /usr/include/php/php3_compat.h
> /usr/include/php/safe_mode.h
> /usr/include/php/fopen-wrappers.h
> /usr/include/php/php_version.h
> /usr/include/php/php_globals.h
> /usr/include/php/php_reentrancy.h
> /usr/include/php/php_ini.h
> /usr/include/php/SAPI.h
> /usr/include/php/php_config.h
> /usr/include/php/zend_config.h
> /usr/include/php/build-defs.h
> /usr/lib/php

> /usr/include/php/ext/standard/datetime.h
> /usr/include/php/ext/standard/dl.h
> /usr/include/php/ext/standard/dns.h
> /usr/include/php/ext/standard/exec.h
> /usr/include/php/ext/standard/file.h
> /usr/include/php/ext/standard/flock_compat.h
> /usr/include/php/ext/standard/fsock.h
> /usr/include/php/ext/standard/global.h
> /usr/include/php/ext/standard/head.h
> /usr/include/php/ext/standard/html.h
> /usr/include/php/ext/standard/info.h
> /usr/include/php/ext/standard/md5.h
> /usr/include/php/ext/standard/microtime.h
> /usr/include/php/ext/standard/pack.h
> /usr/include/php/ext/standard/pageinfo.h
> /usr/include/php/ext/standard/php_array.h
> /usr/include/php/ext/standard/php_assert.h
> /usr/lib/php/DB
> /usr/lib/php/DB/common.php
> /usr/lib/php/DB/odbc.php
> /usr/lib/php/DB/mysql.php
> /usr/lib/php/DB/pgsql.php
> /usr/lib/php/DB/storage.php
> /usr/lib/php/build
> /usr/lib/php/build/pear.m4
> /usr/lib/php/build/fastgen.sh
> /usr/lib/php/build/library.mk
> /usr/lib/php/build/ltlib.mk
> /usr/lib/php/build/program.mk
> /usr/lib/php/build/rules.mk
> /usr/lib/php/build/rules_pear.mk
> /usr/lib/php/build/shtool
> /usr/lib/php/build/acinclude.m4
> /usr/lib/php/DB.php

Những tập tin được cài đặt cho Perl language với web server Apache

> /usr/lib/perl5/5.00503/i386-linux/perllocal.pod
> /usr/lib/perl5/man/man3/Apache.3
> /usr/lib/perl5/man/man3/Apache::Constants.3
> /usr/lib/perl5/man/man3/Apache::Leak.3
> /usr/lib/perl5/man/man3/Apache::Log.3
> /usr/lib/perl5/man/man3/Apache::PerlRunXS.3
> /usr/lib/perl5/man/man3/Apache::Symbol.3
> /usr/lib/perl5/man/man3/Apache::Table.3
> /usr/lib/perl5/man/man3/Apache::URI.3
> /usr/lib/perl5/man/man3/Apache::Util.3
> /usr/lib/perl5/man/man3/Apache::FakeRequest.3
> /usr/lib/perl5/man/man3/mod_perl.3
> /usr/lib/perl5/man/man3/Apache::ExtUtils.3
> /usr/lib/perl5/man/man3/Apache::SIG.3
> /usr/lib/perl5/man/man3/Apache::Status.3
> /usr/lib/perl5/man/man3/Apache::Include.3
> /usr/lib/perl5/man/man3/Apache::Debug.3

> /usr/lib/perl5/man/man3/Apache::Resource.3

> /usr/lib/perl5/man/man3/Apache::src.3

> /usr/lib/perl5/man/man3/Apache::PerlRun.3

> /usr/lib/perl5/man/man3/Apache::httpd_conf.3

> /usr/lib/perl5/man/man3/mod_perl_traps.3

> /usr/lib/perl5/man/man3/Apache::Options.3

> /usr/lib/perl5/man/man3/mod_perl_cvs.3

> /usr/lib/perl5/man/man3/Apache::Symdump.3

> /usr/lib/perl5/man/man3/Apache::RegistryLoader.3

> /usr/lib/perl5/man/man3/mod_perl_method_handlers.3

> /usr/lib/perl5/man/man3/mod_perl_tuning.3

> /usr/lib/perl5/man/man3/cgi_to_mod_perl.3

> /usr/lib/perl5/man/man3/Apache::StatINC.3

> /usr/lib/perl5/man/man3/Apache::Registry.3

> /usr/lib/perl5/man/man3/Bundle::Apache.3

> /usr/lib/perl5/man/man3/Apache::SizeLimit.3

> /usr/lib/perl5/man/man3/Apache::PerlSections.3

> /usr/lib/perl5/man/man3/Apache::RedirectLogFix.3

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/buff.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/multithread.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/httpd.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_config.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/alloc.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_md5.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_ctx.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/util_md5.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/rfc1413.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/conf.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/util_uri.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/explain.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_compat.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/http_config.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_sha1.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/scoreboard.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/compat.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/http_request.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/http_core.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_mm.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/http_protocol.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/util_date.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_hook.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/http_main.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/http_conf_globals.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/util_script.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/http_vhost.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_ctype.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/hsregex.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_mmn.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/ap_config_auto.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/http_log.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/include/fnmatch.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/netware
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/netware/os.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/netware/getopt.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/netware/test_char.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/netware/uri_delims.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/netware/precomp.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/bs2000
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/bs2000/os-inline.c
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/bs2000/ebcdic.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/bs2000/os.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/tpf

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/tpf/ebcdic.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/tpf/os.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/tpf/os-inline.c
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/service.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/getopt.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/registry.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/resource.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/installer
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/installer/installdll
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/installer/installdll/test
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/installer/installdll/test/test.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/installer/installdll/test/resource.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/os.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/passwd.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/win32/readdir.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/unix
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/unix/os.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/unix/os-inline.c
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/os390
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/os390/os-inline.c
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/os390/ebcdic.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/os390/os.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/mpeix
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/mpeix/os-inline.c
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/mpeix/os.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/os2
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/os2/os.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/os/os2/os-inline.c
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/ssl
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/ssl/ssl_expr.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/ssl/ssl_util_table.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/ssl/ssl_util_ssl.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/ssl/ssl_expr_parse.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/ssl/mod_ssl.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/ssl/ssl_util_sdbm.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/perl
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/perl/mod_perl.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/perl/mod_perl_version.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/perl/perl_PL.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/perl/mod_perl_xs.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/php4
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/php4/mod_php4.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/proxy
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/proxy/mod_proxy.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/standard
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/modules/standard/mod_rewrite.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/support
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/support/suexec.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/iasciitab.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/latin1tab.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/xmldef.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/xmlparse.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/xmltok.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/xmlrole.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/hashtable.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/nametab.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/xmltok_impl.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/utf8tab.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/lib/expat-lite/asciitab.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/regex
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/regex/utils.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/regex/regex2.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/regex/cclass.h
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/include/regex/cname.h

> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/typemap
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/Leak
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/Leak/Leak.so
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/Leak/Leak.bs
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/Symbol
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/Symbol/Symbol.so
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/Apache/Symbol/Symbol.bs
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/mod_perl
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/mod_perl/.packlist
> /usr/lib/perl5/site_perl/5.005/i386-linux/mod_perl.pod
> /usr/lib/perl5/site_perl/5.005/i386-linux/Bundle
> /usr/lib/perl5/site_perl/5.005/i386-linux/Bundle/Apache.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/test.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Debug.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Resource.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/src.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/httpd_conf.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Symdump.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/RegistryLoader.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Registry.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/SizeLimit.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/RedirectLogFix.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/MyConfig.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Constants
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Constants/Exports.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/SIG.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/StatINC.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Opcode.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/PerlSections.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/FakeRequest.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/ExtUtils.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Include.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Status.pm

> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/PerlRun.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Options.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/RegistryNG.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/RegistryBB.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Connection.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Constants.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/File.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Leak.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Log.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/ModuleConfig.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/PerlRunXS.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Server.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Symbol.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Table.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/URI.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache/Util.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/mod_perl_hooks.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/mod_perl_hooks.pm.PL
> /usr/lib/perl5/site_perl/5.005/i386-linux/mod_perl_tuning.pod
> /usr/lib/perl5/site_perl/5.005/i386-linux/mod_perl_cvs.pod
> /usr/lib/perl5/site_perl/5.005/i386-linux/mod_perl_method_handlers.pod
> /usr/lib/perl5/site_perl/5.005/i386-linux/mod_perl.pm
> /usr/lib/perl5/site_perl/5.005/i386-linux/mod_perl_traps.pod
> /usr/lib/perl5/site_perl/5.005/i386-linux/cgi_to_mod_perl.pod
> /usr/lib/perl5/site_perl/5.005/i386-linux/Apache.pm

Chương 20:

CÀI ĐẶT APACHE VỚI CÁC THÀNH PHẦN TÙY CHỌN

Trong chương này chúng ta sẽ đề cập đến các phần mềm sau:

- Linux Webalizer

  Cấu hình.

  Báo cho Apache biết về thư mục output của Webalizer.

  Chạy Webalizer bằng tay trong lần đầu tiên.

  Chạy Webalizer một cách tự động với một cron job.

- Linux FAQ-O-Matic

  Báo cho Apache biết vị trí của các tập tin Faq-O-Matic.

  Cấu hình phần mềm FAQ-O-Matic của bạn.

- Webmail IMP

  Cài đặt PHPLib, thành phần được yêu cầu bởi chương trình Horde của Webmail IMP.

  Cấu hình và tạo cơ sở dữ liệu SQL cho Webmail IMP.

  Cấu hình tập tin "php.ini" cho PHP4 của bạn.

  Cấu hình Apache để nhận biết Webmail IMP.



Trong chương này, chúng tôi sẽ nói về ba chương trình tiện ích mà bạn có thể cài đặt trên máy Linux server của bạn. Các chương trình này đã được tạo để dùng với Apache Web Server, chúng chỉ ích lợi nếu bạn có ý định dùng Apache trong cơ quan, tổ chức của bạn. Còn trong trường hợp ngược lại, bạn có thể bỏ qua chương này, để tiếp tục với những chương khác trong quyển sách. Đầu tiên, chúng tôi sẽ nói về Webalizer, đó là một chương trình phân tích logfile cho web server. Sau đó chúng ta sẽ thảo luận về FAQ-O-Matic, nó có thể được dùng khi bạn cần một danh sách FAQ (Frequently Asked Question), cơ sở dữ liệu theo dõi lỗi (bug-tracing database), tài liệu và nhiều thứ khác nữa trên máy server của bạn. Cuối cùng, chúng tôi sẽ nói về một chương trình rất thú vị và tinh vi được gọi là webmail, nó cho phép bạn sẵn sàng cung cấp các tài khoản mail miễn phí cho các khách hàng của bạn, hoặc truy cập và đọc mail của bạn, trong một tình trạng được bảo mật.

Linux Webalizer

Giới thiệu tổng quan

Một web server giống như Apache ghi lại tất cả các thông tin đầu vào đã được định nghĩa trước vào trong một tập tin văn bản mà nó có thể được xem và được phân tích bởi người quản trị web. Tập tin này được đánh giá bởi một chương trình đặc biệt mà có thể sản sinh ra thông tin bằng một sự trình bày đồ hoạ, làm cho việc thông dịch trở nên dễ dàng hơn cho người quản trị.

Cũng như được giải thích trong tập tin README của Webalizer:

Webalizer là một chương trình phân tích logfile web server, nó đưa ra thống kê việc sử dụng HTML cho một trình duyệt web. Các kết quả được trình bày ở cả hai dạng bao gồm hình cột và ở dạng đồ hoạ, mà nó dễ dàng thông dịch. Thống kê việc sử dụng hàng giờ, hàng ngày, hàng tháng, hàng năm được trình bày cùng với khả năng hiển thị việc sử dụng (usage) bởi URL site, người tham khảo (referer), tác nhân người dùng (user agent) trên trình duyệt browser.

Hướng dẫn cài đặt

Các dòng lệnh là tương thích với Unix.

Đường dẫn nguồn sẽ là "/var/tmp" (cũng có thể chọn những đường dẫn khác).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ xảy ra với người sử dụng có quyền cao nhất trong hệ thống (root).

Phiên bản Webalizer là 1_30-04

Các gói cần thiết (Packages)

Webalizer Homepage Site: http://www.mrunix.net/webalizer/

Webalizer FTP Site: 207.153.121.6

Bạn phải bảo đảm là tải về gói : webalizer –1_30-04-src.tgz

Sự biên dịch

Giải nén dạng lưu trữ tarball (tar.gz)

```
[root@deep /]# cp webalizer-version-src.tgz /var/tmp/
[root@deep /]# cd /var/tmp/
[root@deep tmp]# tar xzpf webalizer-version-src.tgz
```

Yêu cầu ban đầu

Webalizer yêu cầu phải có thư viện đồ hoạ GD Graphics Library, mà nó được yêu cầu cho việc phát sinh các đồ hoạ nội tuyến (inline) cho chương trình webalizer, được cài đặt sẵn trên hệ thống của bạn. Nếu trong trường hợp chưa cài đặt, bạn phải cài đặt nó từ Red Hat Linux CD-ROM.

- Để kiểm tra xem package GD đã được cài đặt trên Linux server của bạn chưa, dùng lệnh sau:

```
[root@deep /]# rpm -qi gd
package gd is not installed
```

- Để cài đặt các package GD trên hệ thống Linux của bạn, dùng lệnh sau:

```
[root@deep /]# mount /dev/cdrom /mnt/cdrom/
[root@deep /]# cd /mnt/cdrom/Red Hat/RPMS/
[root@deep RPMS]# rpm -Uvh gd-version.i386.rpm
gd
##################################################
[root@deep RPMS]# rpm -Uvh gd-devel-version.i386.rpm
gd-devel
##################################################
[root@deep RPMS]# cd /; umount /mnt/cdrom/
```

- Apache web server nên được cài đặt sẵn trên hệ thống của bạn và hiển nhiên là có thể sử dụng phần mềm Webalizer.

***Ghi chú**: Để biết thêm thông tin về Apache web server, xem các chương liên quan trong quyển sách này.*

Biên dịch

Chuyển vào trong thư mục Webalizer mới và gõ các lệnh sau trên terminal của bạn:

```
CC="egcs" \
CFLAGS="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomit-frame-
pointer -fno-exceptions" \
./configure \
--prefix=/usr

[root@deep webalizer-1.30-04]# make
[root@deep webalizer-1.30-04]# make install
[root@deep webalizer-1.30-04]# mkdir /home/httpd/usage
```

Lệnh "**make**" biên dịch tất cả các tập tin nguồn sang dạng nhị phân có thể thi hành, và lệnh "**make install**" cài đặt các tập tin nhị phân này và bất kỳ các tập tin hỗ trợ vào các nơi thích hợp. Lệnh "**mkdir**" sẽ tạo một thư mục mới có tên là "usage" trong thư mục "/home/httpd" nơi mà chúng ta sẽ xử lý tất cả các tập tin Webalizer liên quan.

Dọn dẹp sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf webalizer-version/ webalizer-version-src.tgz
```

Lệnh "**rm**" sẽ xoá tất cả các tập tin nguồn mà chúng ta dùng để biên dịch và cài đặt Webalizer. Nó cũng sẽ xoá dạng lưu trữ nén Webalizer từ thư mục "/var/tmp".

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy Webalizer, các tập tin sau được yêu cầu và phải được tạo hay được sao chép vào các thư mục thích hợp trên máy server của bạn:

  Sao chép tập tin Webalizer.config vào trong thư mục "/etc/".

Cấu hình cho tập tin "/etc/webalizer.config"

Tập tin "/etc/webalizer.config" là tập tin cấu hình mặc định cho Webalizer. Với nó, bạn có thể chỉ rõ những thư mục hay trang nào trong web site của bạn để phân tích, các URL nào để ẩn, và v.v...Mặc nhiên chương trình Webalizer đã cài đặt một tập tin cấu hình mẫu có tên là "webalizer.conf.sample" trong thư mục "/etc/" của Linux. Bạn có thể dùng tập tin này để cấu hình các chọn lựa của bạn và sau đó đổi tên nó là "webalizer.conf", và chương trình Webalizer sẽ có thể tìm và dùng nó. Có rất nhiều tùy chọn có sẵn và như vậy điều quan trọng là phải đọc kỹ tài liệu để đến với Webalizer, biết nhiều thông tin hơn về tất cả các thông số cài đặt. Một việc cũng quan trọng đáng lưu ý là bạn phải ghi chú trong tập tin cấu hình Webalizer này hầu hết các tham số được dùng và chung nhất.

Soạn thảo tập tin cấu hình Webalizer.conf.sample (vi /etc/webalizer.conf.sample) hay tạo tập tin webalizer.conf (touch /etc/webalizer.conf) và thêm/thay đổi trong tập tin này:

```
LogFile /var/log/httpd/access_log
OutputDir /home/httpd/usage
Incremental yes
PageType htm*
PageType cgi
PageType php
HideURL *.gif
HideURL *.GIF
HideURL *.jpg
HideURL *.JPG
```

```
HideURL *.ra
IgnoreURL /taskbar*
```

Việc này bảo Webalizer.conf tự thiết lập chính nó cho việc cài đặt cấu hình riêng biệt:

LogFile /var/log/httpd/access_log

Tùy chọn "*LogFile*" chỉ ra logfile để dùng với webalizer. Logfile mặc định được giả định là "access_log" của Apache Web Server, nhưng bạn có thể chỉ ra một logfile khác, tựa như một Squid Proxy Server được đặt tên "access.log" nếu bạn dùng nó trong mode httpd-accelerator. Xem chương "Các dịch vụ Proxy" để biết thêm chi tiết.

OutputDir /home/httpd/usage

Tùy chọn "*OutputDir*" chỉ rõ vị trí của thư mục output để dùng cho các báo cáo (report) của Webalizer. Tất cả các tập tin báo cáo hiện tại và tương lai được sinh ra bởi chương trình Webalizer sẽ được lưu giữ trong thư mục này. Có một lời khuyên là bạn nên tạo thư mục này ở nơi mà web site Apache của bạn đang ở đó.

Incremental yes

Tùy chọn "*Incremental*" nếu được đặt là "yes" thì nó chỉ bảo chương trình xử lý logfile từng phần, và cho phép bạn quay vòng các logfile của bạn nhiều hơn khi bạn cần mà không mất thông tin truy cập. Một điều khuyên bạn là nên đặt tùy chọn này là "yes".

PageType htm* cgi php

Tùy chọn "*PageType*" xác định rõ các phần mở rộng mà bạn muốn webalizer xem nó như là một trang để đếm. Mỗi phần mở rộng của tập tin được thêm vào phải được chỉ rõ trên một dòng riêng như được trình bày trong tập tin cấu hình ở trên.

HideURL *.gif *.GIF *.jpg *.JPG *.ra

Tùy chọn "*HideURL*" xác định rõ các dạng nào của các mục gồm các tập tin đồ hoạ, tập tin âm thanh (audio) hay các tập tin non-html khác để ẩn từ các trang báo cáo. Mỗi hạng mục được thêm vào phải được chỉ rõ trên một dòng riêng như được trình bày trong tập tin cấu hình ở trên.

IgnoreURL /taskbar*

Tùy chọn "*IgnoreURL*" chỉ rõ các URL được lờ qua một cách hoàn toàn từ các báo cáo thống kê được phát sinh. Tùy chọn này có thể được dùng để bỏ qua các thư mục nào không quan trọng trong các báo cáo thống kê của chúng ta. Đó cũng là các tiện ích khi bạn muốn quản lý và phân loại các URL nào nên được theo dõi và các URL nào nên được bỏ qua.

Ghi chú: *Nếu bạn quyết định dùng tập tin đã có là "/etc/webalizer.conf.sample" để cấu hình các thông số của bạn cho webalizer, đừng quên đổi tên nó là "webalizer.conf" nếu không chương trình sẽ không thể sử dụng nó.*

Báo cho Apache biết về thư mục output của Webalizer

Một khi Webalizer đã được cài đặt trên hệ thống, chúng ta phải thêm các dòng sau vào tập tin "httpd.conf" của Apache để có thể định vị và dùng các đặc trưng của nó.

Bước 1

Soạn thảo tập tin cấu hình httpd.conf (vi /etc/httpd.conf) và thêm các dòng sau giữa phần tag <lfModule mod_alias.c> và </lfModule>:

```
Alias /usage/ "/home/httpd/usage/"
<Directory "/home/httpd/usage">
    Options None
    AllowOverride None
    Order deny,allow
    Deny from all
    Allow from 192.168.1.0/24
</Directory>
```

Bước 2

Đừng quên khởi động lại Apache web server của bạn một khi bạn đã thêm vào các dòng trên vào tập tin "httpd.conf":

- Để khởi động lại Apache web server, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/httpd restart
Shutting down http:     [ OK ]
Starting httpd:         [ OK ]
```

Chạy Webalizer bằng tay trong lần đầu tiên

Bây giờ đã đến lúc chạy chương trình để phát sinh các báo cáo (dạng html) và hình ảnh đồ họa trong thư mục webalizer mặc định, vì vậy có thể nhìn thấy chúng trong giao diện trình duyệt web của chúng ta. Bước này được yêu cầu chỉ cho lần chạy đầu tiên mà bạn cài đặt và sử dụng Webalizer, và khi nó thật sự trở nên cần thiết thì dùng một cron job để tự động hoá công việc này trong tương lai.

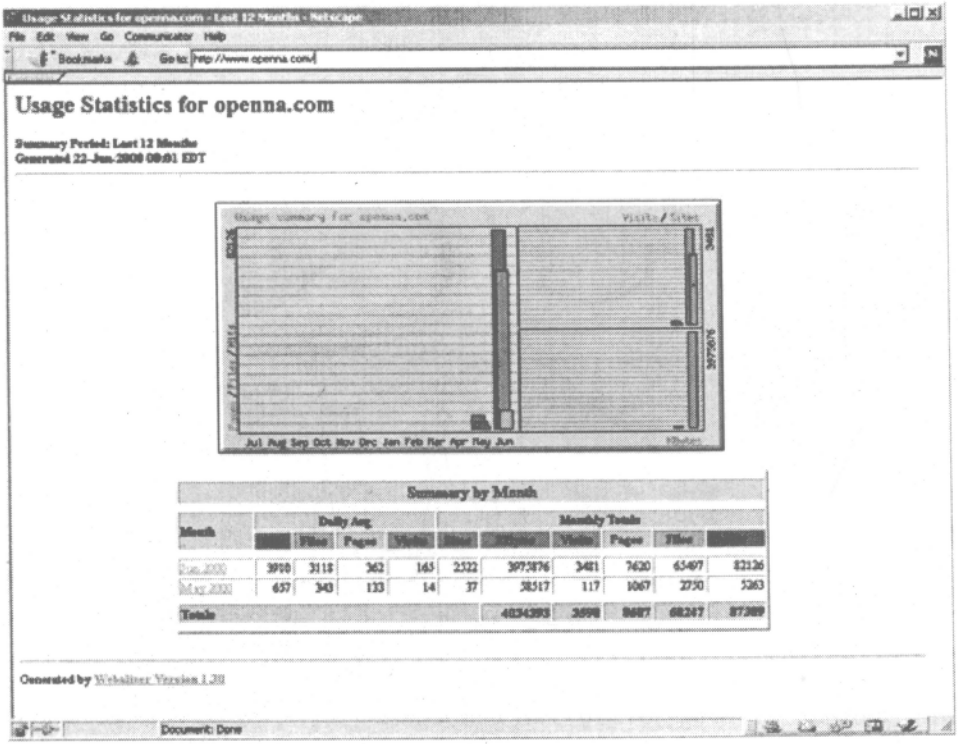
- Chạy Webalizer bằng tay, để sinh ra các báo cáo, dùng lệnh sau:

```
[root@deep /]# /usr/bin/webalizer
Webalizer V1.30-04 (Linux 2.2.14) English
Using logfile /var/log/httpd/access_log
Creating output in /home/httpd/usage
Hostname for reports is 'deep.openna.com'
History file not found...
```

```
Previous run data not found...
Saving current run data... [03/06/2000 04:42:03]
Generating report for March 2000
Generating summary report
Saving history information...
81 records (2 ignored) in 0.31 seconds
```

Ở giai đoạn này, chúng ta nên kiểm tra xem webalizer có đang làm việc trên hệ thống không. Để làm việc đó, vào trình duyệt web của bạn và gõ vào địa chỉ: http://my-web-server/usage/

Trong đó <my-web-server> là địa chỉ nơi Apache web server hoạt động, và </usage> là thư mục chứa tất cả các tập tin báo cáo Webalizer.



Chạy Webalizer một cách tự động với một cron job

Việc sau cùng mà bạn có thể làm bây giờ là tự động hoá việc sinh ra các logfile mới để cập nhật các báo cáo Webalizer. Đặt một entry vào crontabs của root để chạy Webalizer như là một cron job: để thêm vào Webalizer trong cron job của bạn, bạn phải hiệu chỉnh crontab và thêm vào dòng sau để chạy nó, ví dụ, thiết đặt cứ mỗi 28 phút sẽ sinh ra các report (báo cáo) mới cho các website bị bận:

- Hiệu chỉnh tập tin crontab và thêm vào dòng sau:

```
# Run Webalizer to update Apache Log files every 28 minutes.
```

```
28 * * * * /usr/bin/webalizer
```

- Để hiệu chỉnh crontab tự động thực hiện công việc trên dùng lệnh (như root):

```
[root@deep /]# crontab -e
```

Các tập tin được cài đặt

```
> /etc/webalizer.conf.sample
> /usr/bin/webalizer
> /usr/man/man1/webalizer.1
```

Linux FAQ-O-Matic

Giới thiệu tổng quan

Như được giải thích trong web site của Faq-O-Matic:

Rất tiện lợi khi có một nơi lưu trữ danh sách các thư gởi (mailing list), bởi vì nó cho phép những người quan tâm đến các câu hỏi được hỏi thường xuyên (Frequently Asked Questions) có thể tự tìm ra một câu trả lời ngay lập tức, và tránh gây ra phiền toái cho những người có các câu trả lời. Thật không may là các câu trả lời trong nơi lưu trữ mailing list đã trở nên quá cũ theo thời gian, nó phá rối tổ chức, dẫn đến việc tách rời khỏi từ nơi thảo luận xôn xao của mailing list.

Một danh sách các câu hỏi và trả lời thường xuyên (FAQ) là sự kiện tốt, bởi vì với các câu hỏi, mọi người thường lười biếng ít tìm hiểu hơn và vẫn muốn tìm câu trả lời ngay của họ. Thật không may, việc duy trì một danh sách FAQ là yêu cầu phải cố gắng nỗ lực; nếu mọi người có các câu trả lời trở nên uể oải hay lười biếng thì danh sách FAQ trở nên cũ rích.

Faq-O-Matic là một hệ thống dựa trên CGI mà nó tự động xử lý việc duy trì một danh sách FAQ. Nó cho phép các người viếng thăm tới FAQ của bạn tham gia vào và lưu giữ cập nhật nó. Một hệ thống cũng chấp nhận nó như một ứng dụng trợ giúp (help-desk application), cơ sở dữ liệu theo dõi lỗi (bug-tracking database), hay tư liệu hệ thống.

Hướng dẫn cài đặt

Các dòng lệnh phải tương thích với Unix.

Đường dẫn nguồn là "/var/tmp" (cũng có thể chọn những đường dẫn khác).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt được thực hiện với người sử dụng có quyền cao nhất trong hệ thống (root).

Phiên bản FAQ-O-Matic là 2.709.

Các gói cần thiết (Packages)

FAQ-O-Matic Homepage Site: http://www.dartmouth.edu/~jonh/ff-serve/cache/1.html

FAQ-O-Matic FTP Site: ftp://ftp.cs.dartmouth.edu/pub/jonh

Bạn phải bảo đảm là tải về gói : FAQ-OMatic –2.709.tar.gz.

Yêu cầu ban đầu

- Apache web server nên được cài đặt sẵn trên hệ thống của bạn để có thể sử dụng phần mềm FAQ-O-Matic.

- Các công cụ quản lý phiên bản tập tin Revision Control System (RCS) cũng nên được cài đặt sẵn trên hệ thống của bạn để có thể sử dụng phần mềm FAQ-O-Matic.

- Để kiểm tra xem package RCS được cài đặt trên hệ thống của bạn chưa, dùng lệnh:

```
[root@deep /]# rpm -qi rcs
package rcs is not installed
```

- Để cài đặt package RCS trên hệ thống Linux của bạn, dùng các lệnh:

```
[root@deep /]# mount /dev/cdrom /mnt/cdrom/
[root@deep /]# cd /mnt/cdrom/Red Hat/RPMS/
[root@deep RPMS]# rpm -Uvh rcs-version.i386.rpm
rcs     ##################################################
[root@deep RPMS]# cd /; umount /mnt/cdrom/
```

Ghi chú: *Để biết thêm thông tin về phần mềm được yêu cầu, xem các chương liên quan trong quyển sách này.*

Sự biên dịch

Giải nén dạng lưu trữ tarball (tar.gz)

```
[root@deep /]# cp FAQ-O-Matic-version.tar.gz /var/tmp/
[root@deep /]# cd /var/tmp/
[root@deep tmp]# tar xzpf FAQ-O-Matic-version.tar.gz
```

Biên dịch

Để cài đặt chương trình Faq-O-Matic trên máy tính của bạn, chuyển vào trong thư mục mới của FAQ-O-Matic và gõ các lệnh sau trên terminal của bạn:

```
[root@deep FAQ-OMatic-2.709]# perl Makefile.PL
[root@deep FAQ-OMatic-2.709]# make
[root@deep FAQ-OMatic-2.709]# make install
[root@deep FAQ-OMatic-2.709]# mv fom /home/httpd/cgi-bin/ (hoặc bất cứ ở đâu các CGI của bạn đang có hiệu lực).
[root@deep FAQ-OMatic-2.709]# mkdir -p /home/httpd/cgi-bin/fom-meta
[root@deep FAQ-OMatic-2.709]# mkdir -p /home/httpd/faqomatic
[root@deep FAQ-OMatic-2.709]# chown root.www /home/httpd/cgi-bin/fom
[root@deep FAQ-OMatic-2.709]# chown -R www.www /home/httpd/cgi-bin/fom-meta/
[root@deep FAQ-OMatic-2.709]# chown -R www.www /home/httpd/faqomatic/
```

Lệnh "**make**" biên dịch tất cả các tập tin nguồn sang dạng nhị phân có thể thi hành, và lệnh "**make install**" cài đặt các tập tin nhị phân này và bất kỳ các tập tin hỗ trợ vào các nơi thích hợp. Lệnh "**mv**" sẽ dời chương trình CGI "form" chính của Faq-O-Matic sang thư mục "cgi-bin" của Apache web server của bạn. Lệnh "**mkdir**" sẽ tạo một thư mục mới có tên là "fom-meta" trong thư mục

"/home/httpd" nơi chúng ta sẽ xử lý tất cả các tập tin Faq-O-Matic liên quan. Cuối cùng lệnh **"chown"** sẽ đặt chương trình CGI "fom" được sở hữu riêng là super-user "root" và nhóm (group) người dùng Apache chạy như "www", và sẽ đặt các thư mục "fom-meta" và "faqomatic" là được sở hữu riêng bởi user "www" và group "www".

Ghi chú: *Bạn sẽ nhận một mật khẩu tạm thông qua mail trong suốt quá trình biên dịch phần mềm. mật khẩu này sẽ cần thiết để hoàn thành sau cùng việc cài đặt của Faq-O-Matic thông qua giao diện web.*

Báo cho Apache biết vị trí của các tập tin Faq-O-Matic

Một khi Faq-O-Matic đã được cài đặt trên hệ thống, chúng ta phải thêm vào các dòng sau vào tập tin "httpd.conf" của Apache để có thể định vị và dùng các đặc trưng của nó.

Bước 1

Soạn thảo tập tin cấu hình httpd.conf (vi /etc/conf/httpd.conf) và thêm vào các dòng sau giữa các phần tag <lfModule mod_alias.c> và </lfModule>:

```
Alias /faqomatic/ "/home/httpd/faqomatic/"
<Directory "/home/httpd/faqomatic">
    Options None
    AllowOverride None
    Order allow,deny
    Allow from all
    </Directory>

Alias /bags/ "/home/httpd/faqomatic/bags/"
<Directory "/home/httpd/faqomatic/bags">
    Options None
    AllowOverride None
        Order allow,deny
    Allow from all
    </Directory>

Alias /cache/ "/home/httpd/faqomatic/cache/"
<Directory "/home/httpd/faqomatic/cache">
    Options None
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

```
Alias /item/ "/home/httpd/faqomatic/item/"
<Directory "/home/httpd/faqomatic/item">
    Options None
    AllowOverride None
    Order allow,deny
    Allow from all
    </Directory>
```

Bước 2

Đừng quên khởi động lại Apache web server của bạn một khi bạn đã thêm vào các dòng trên vào tập tin "httpd.conf":

- Để khởi động lại Apache web server, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/httpd restart
Shutting down http:        [ OK ]
Starting httpd:            [ OK ]
```

Cấu hình phần mềm FAQ-O-Matic của bạn

Phần còn lại của việc cài đặt này sẽ được tạo thông qua việc dùng trình duyệt web của bạn. Với Netscape Communicator, theo các bước đơn giản được trình bày bên dưới:

Bước 1

Khởi động (start) trình duyệt web của bạn và dùng nó để cấu hình Faq-O-Matic.

1. Trỏ vào trình duyệt Nestcape tới địa chỉ: http://my-web-server/cgi-bin/fom.
2. Nhập vào mật khẩu tạm của bạn.
3. Tạo thư mục "/home/httpd/cgi-bin/fom-meta/" trước.
4. Cấu hình "**Define configuration parameters**" trong menu cấu hình chính.

Lấy ví dụ điền vào các phần phía dưới được đánh dấu **in đậm** theo thông tin sau:

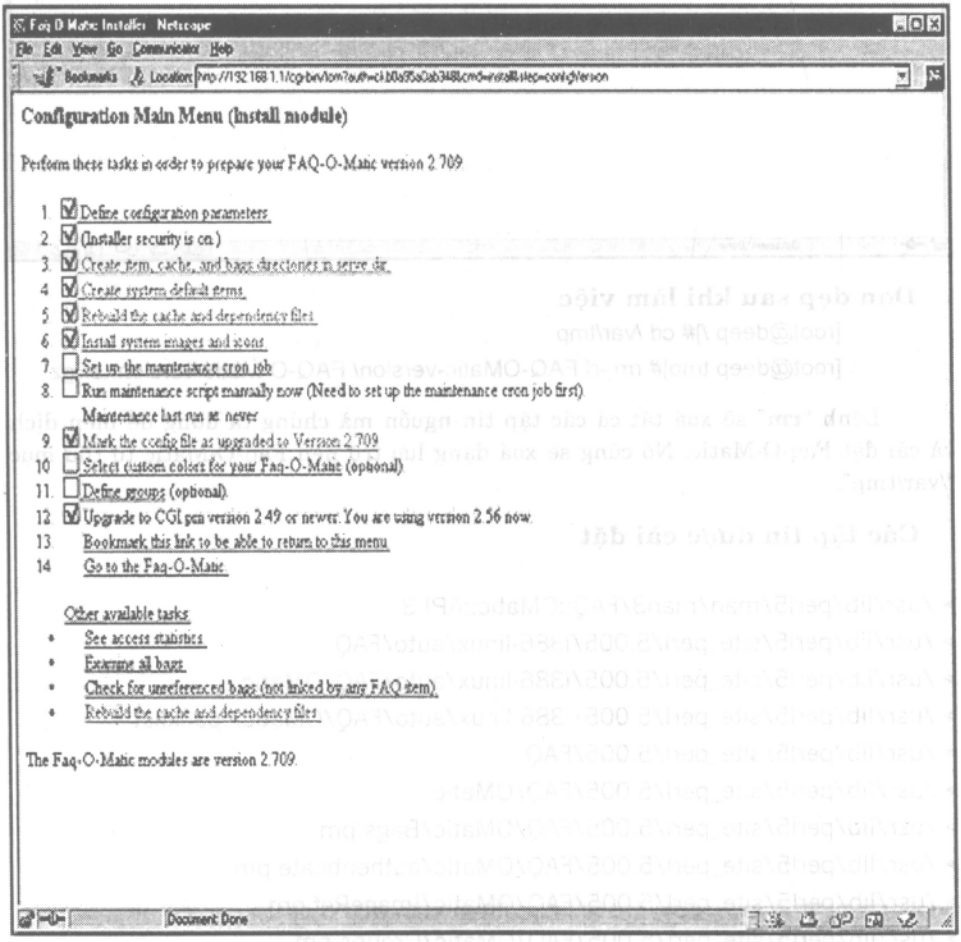
```
$adminAuth= admin@openna.com
$serverBase= http://www.openna.com
$cgiURL= /cgi-bin/fom
$serveDir= /home/httpd/faqomatic/
$serveURL= /faqomatic/
```

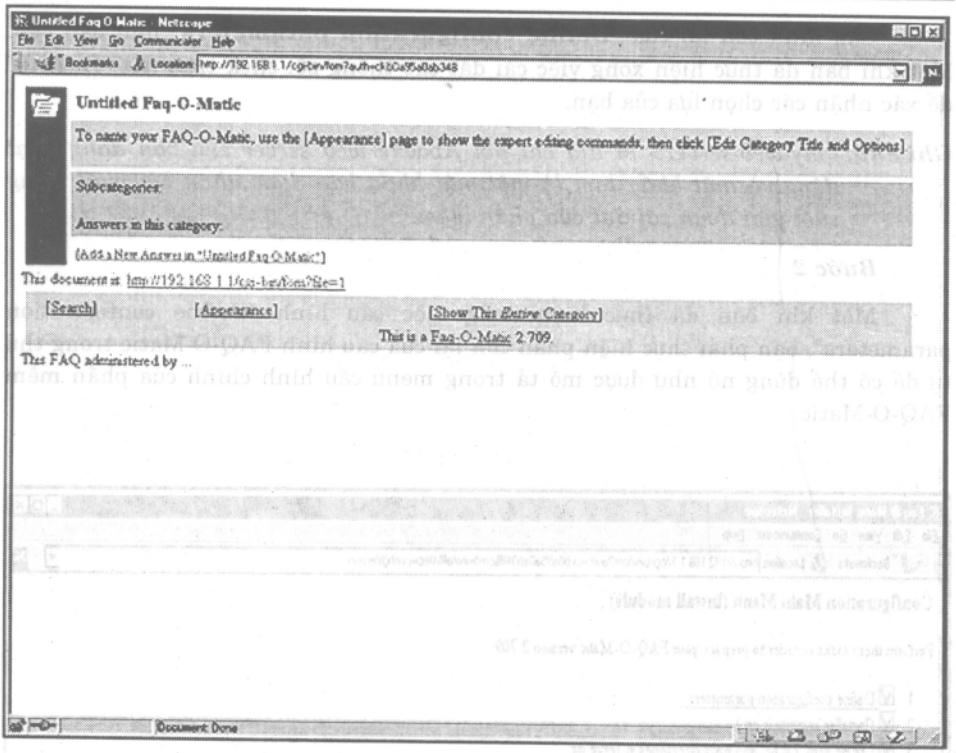
Cấu hình còn lại của "**Define configuration parameters**" như bạn cần. Một khi bạn đã thực hiện xong việc cài đặt các thông số, click trên nút "**Define**" để xác nhận các chọn lựa của bạn.

Ghi chú: *<my-web-server> là địa chỉ nơi Apache web server của bạn đang hoạt động và mật khẩu tạm là một mật khẩu bạn được nhận bởi mail trong suốt giai đoạn cài đặt của phần mềm.*

Bước 2

Một khi bạn đã thực hiện xong việc cấu hình "Define configuration parameters", bạn phải thực hiện phần còn lại của cấu hình FAQ-O-Matic trong thứ tự để có thể dùng nó như được mô tả trong menu cấu hình chính của phần mềm FAQ-O-Matic.





Dọn dẹp sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf FAQ-OMatic-version/ FAQ-O-Matic-version.tar.gz
```

Lệnh "**rm**" sẽ xoá tất cả các tập tin nguồn mà chúng ta dùng để biên dịch và cài đặt Faq-O-Matic. Nó cũng sẽ xoá dạng lưu trữ nén Faq-O-Matic từ thư mục "/var/tmp".

Các tập tin được cài đặt

```
> /usr/lib/perl5/man/man3/FAQ::OMatic::API.3
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/FAQ
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/FAQ/OMatic
> /usr/lib/perl5/site_perl/5.005/i386-linux/auto/FAQ/OMatic/.packlist
> /usr/lib/perl5/site_perl/5.005/FAQ
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Bags.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/authenticate.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/ImageRef.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Groups.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/submitGroup.pm
```

> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/recent.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/submitItem.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/maintenance.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Language_de_iso8859_1.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Slow.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/help.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/selectBag.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/submitPart.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/delPart.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/buildSearchDB.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/mirrorServer.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/editItem.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/search.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/SearchMod.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/addItem.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Versions.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/displaySlow.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Language_fr.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/img.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/editPart.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/AuthLocal.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/ColorPicker.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/ImageData.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/changePass.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/submitBag.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/submitModOptions.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/I18N.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Log.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/appearanceForm.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/moveItem.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/editGroups.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/HelpMod.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/searchForm.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/submitPass.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/submitMove.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Set.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/statgraph.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/stats.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Item.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Words.pm

> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Appearance.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/dispatch.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/editBag.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/submitCatToAns.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/submitAnsToCat.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/editModOptions.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Auth.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/install.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/Part.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/faq.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic/API.pm
> /usr/lib/perl5/site_perl/5.005/FAQ/OMatic.pm

Linux Webmail IMP

Giới thiệu tổng quan

Webmail IMP là phần mềm phổ biến, cho phép truy cập các IMAP/POP3 server dựa trên nền web và cung cấp một danh sách địa chỉ e-mail (address book). LDAP directory hỗ trợ tính năng tìm kiếm, hỗ trợ đầy đủ trong việc gửi và nhận thông tin theo dạng đính kèm, và rất nhiều các đặc trưng khác thông thường chỉ thấy nó xuất hiện trên desktop của các máy mail client. Nếu bạn đã cài đặt Apache với sự hỗ trợ của SSL, các máy client có thể truy cập và đọc mail trong một trạng thái bảo mật bằng cách mã hoá của SSL. Mặc định trong phần này, chúng ta cấu hình Webmail IMP dùng cơ sở dữ liệu PostgreSQL và các kết nối IMAP. Tuy nhiên, cũng có rất nhiều cơ sở dữ liệu khác được hỗ trợ bên trong Webmail IMP. Nếu như bạn thích, bạn có thể dùng MySQL, Oracle, Sysbase, hay các cơ sở dữ liệu nổi tiếng khác. Bạn cũng có thể chọn sử dụng POP3 thay cho kết nối IMAP tới các máy client của bạn.

Hướng dẫn cài đặt

Các dòng lệnh phải tương thích với Unix.

Đường dẫn nguồn là "/var/tmp" (cũng có thể chọn những đường dẫn khác).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong tiến trình cài đặt sẽ được thực hiện với người sử dụng có quyền cao nhất trong hệ thống (root).

Phiên bản Horde là 1.2.0

Phiên bản Webmail IMP là 2.2.0

Phiên bản PHPLib là 7.2b

Các gói cần thiết (Packages)

Webmail IMP Homepage Site: http://www.horde.org/imp/

Bạn phải bảo đảm là tải về gói : horde–1.2.0-pre11.tar.gz

Bạn phải bảo đảm là tải về gói : imp–2.2.0-pre11.tar.gz

PHPLib Homepage: http://phplib.netuse.de/index.php3

Bạn phải bảo đảm là tải về gói : phplib-7.2b.tar.gz

Yêu cầu ban đầu

- Apache web server nên được cài đặt sẵn trên hệ thống của bạn để có thể sử dụng phần mềm Webmail IMP.
- Hỗ trợ ngôn ngữ script PHP4 phía server nên được cài đặt sẵn trên hệ thống của bạn để có thể dùng phần mềm Webmail IMP.

- Cơ sở dữ liệu Postgresql hay database server nào khác nên được cài đặt trên hệ thống của bạn nếu như bạn có dự định sử dụng phần mềm Webmail IMP với sự hỗ trợ của SQL.
- Server OpenLDAP directory nên được cài đặt trên hệ thống của bạn nếu như bạn có dự định dùng phần mềm Webmail IMP với sự hỗ trợ của LDAP.
- Server IMAP/POP nên được cài đặt sẵn trên hệ thống của bạn để có thể dùng phần mềm Webmail IMP.
- Các tập tin PHPLIB phiên bản 7.2 hay cao hơn nên được cài đặt sẵn trên hệ thống của bạn để có thể dùng phần mềm Webmail IMP.

Ghi chú: *Để biết thêm thông tin về các phần mềm được yêu cầu, xem các chương liên quan trong quyển sách này.*

Cài đặt PHPLib, thành phần được yêu cầu bởi chương trình Horde của Webmail IMP

Để có thể chạy Webmail IMP trên máy Linux server của bạn, thì phần mềm PHPLib (một công cụ phát triển các ứng dụng Web dành cho các lập trình viên PHP) phải được cài đặt. Để cài đặt PHPLib, theo những bước đơn giản sau:

Các gói cần thiết (Packages)

PHPLib Homepage: http://phplib.netuse.de/index.php3

Bạn phải bảo đảm là tải về gói : phplib-7.2b.tar.gz

```
[root@deep /]# cp phplib-7.2b.tar.gz /home/httpd/
[root@deep /]# cd /home/httpd/
[root@deep httpd]# tar xzpf phplib-7.2b.tar.gz
```

Bước 1

Chuyển vào DocumentRoot của web server của bạn, tạo một thư mục "/home/httpd/php" bằng cách thi hành các lệnh sau:

```
[root@deep /]# cd /home/httpd/
[root@deep httpd]# mkdir php
```

Bước 2

Sao chép nội dung của thư mục "php" chứa đựng các PHPLib vào thư mục "php" mà bạn tạo trong DocumentRoot:

```
[root@deep /]# cd /home/httpd/phplib-7.2b/php/
[root@deep php]# cp * /home/httpd/php/
[root@deep php]# cd /home/httpd/
[root@deep httpd]# rm -f phplib-7.2b.tar.gz
```

```
[root@deep httpd]# rm -rf phplib-7.2b/
```

Ghi chú: *Xóa các dạng lưu trữ tar và thư mục "phplib-version" của PHPLib một khi đã thực hiện xong việc sao chép thư mục "php" của nó sang thư mục "php" mới mà chúng ta tạo được trong DocumentRoot.*

Biên dịch

Để cài đặt chương trình Webmail IMP trên server của bạn, thực hiện các bước đơn giản như bên dưới:

Bước 1

Sao chép horde-1.2.0-pre11.tar.gz vào DocumentRoot của web server của bạn (/home/httpd/), giải nén tar và chuyển dời thư mục từ "horde-version" sang "horde" bằng cách thi hành các lệnh sau:

```
[root@deep /]# cp horde-version.tar.gz /home/httpd/
[root@deep /]# cd /home/httpd/
[root@deep httpd]# tar xzpf horde-version.tar.gz
[root@deep httpd]# mv horde-version horde
[root@deep httpd]# rm -f horde-version.tar.gz
```

Ghi chú: *Xóa các dạng lưu trữ tar của Hordse một khi chúng ta đã thực hiện xong việc chuyển thư mục "horde-version" của nó sang thư mục "horde" mới.*

Bước 2:

Sao chép imp-2.2.0-pre11.tar.gz vào thư mục mới "horde" của bạn (/home/httpd/horde/), giải nén tập tin dạng tar trên và chuyển thư mục từ "imp-version" sang "imp" bằng cách thi hành các lệnh sau:

```
[root@deep /]# cp imp-version.tar.gz /home/httpd/horde/
[root@deep /]# cd /home/httpd/horde/
[root@deep horde]# tar xzpf imp-version.tar.gz
[root@deep horde]# mv imp-version imp
[root@deep horde]# rm -f imp-version.tar.gz
```

Ghi chú: *Một điều quan trọng ở đây là thư mục "imp" nằm trong thư mục "horde", nếu không Webmail sẽ không làm việc. Xóa các dạng lưu trữ tar của IMP một khi chúng ta đã thực hiện xong việc chuyển thư mục "imp-version" của nó sang thư mục "imp" mới.*

Bước 3:

Thay đổi thư mục "horde" và tất cả các thư mục con cùng các tập tin của nó được sở hữu riêng bởi super-user "root" vì một số lý do bảo mật.

```
[root@deep /]# chown -R 0.0 /home/httpd/horde/
```

Bước 4:

Sao chép các tập tin "/home/httpd/horde/phplib/*.ihtml" vào thư mục mới "php" (/home/httpd/php) bằng cách thi hành các lệnh sau:

```
[root@deep /]# cp /home/httpd/horde/phplib/*.ihtml /home/httpd/php/
```

Cấu hình và tạo cơ sở dữ liệu SQL cho Webmail IMP

Bây giờ, chúng ta phải cấu hình cơ sở dữ liệu của chúng ta để có thể dùng Webmail IMP với cơ sở dữ liệu SQL. Phương pháp dễ hơn là dùng các script được định nghĩa trước trong thư mục con "/home/httpd/horde/imp/config/scripts/". Đối với sự hỗ trợ của PostgreSQL, sau đây là các bước đơn giản để cấu hình:

Bước 1

Đầu tiên, chúng ta phải soạn thảo tập tin script "pgsql_create.sql" liên quan với PostgreSQL và định vị trong thư mục con "/home/httpd/horde/imp/config/scripts", thay đổi các giá trị mặc định cho username để chạy như từ "httpd" tới "www".

Soạn thảo tập tin pgsql_create.sql

Chạy lệnh vi /home/httpd/horde/imp/config/scripts/pgsql_create.sql và thay đổi dòng:

```
GRANT SELECT, INSERT, UPDATE ON imp_pref, imp_addr TO nobody;
```

Trở thành:

```
GRANT SELECT, INSERT, UPDATE ON imp_pref, imp_addr TO www;
```

Bước 2

Bây giờ, chúng ta phải định nghĩa username cho Apache với tên là "www" trong cơ sở dữ liệu PostgreSQL, để có thể tạo cơ sở dữ liệu Webmail IMP với username này.

- Để định nghĩa httpd username có tên là "www" trong cơ sở dữ liệu của bạn, chạy chương trình tiện ích createuser của PostgreSQL:

```
[root@deep /]# su postgres
[postgres@deep /]$ createuser
Enter name of user to add ---> www
Enter user's postgres ID or RETURN to use unix user ID: 80 -> [Press Enter]
Is user "www" allowed to create databases (y/n) y
Is user "www" a superuser? (y/n) n
createuser: www was successfully added
```

Bước 3

Một khi httpd user "www" đã được đưa vào trong PostgreSQL, thì log in vào như user của CSDL PostgreSQL của bạn thi hành (trong trường hợp "postgres") và chèn vào đoạn script nhỏ liên quan với PostgreSQL để tự động tạo một cơ sở dữ liệu Webmail IMP trong PostgreSQL.

- Để tự động tạo một cơ sở dữ liệu Webmail IMP trong PostgreSQL, dùng các lệnh sau:

```
[root@deep /]# cd /home/httpd/horde/imp/config/scripts/
[root@deep scripts]# su postgres
[postgres@deep scripts]$ psql template1 < pgsql_create.sql
// IMP database creation script for postgreSQL
// Author: barce@lines.edu
// Date: Aug-29-1998
// Notes: replace "nobody" with yours httpd username
// Run using: psql template1 < pgsql_create.sql

CREATE DATABASE horde;
CREATEDB

\connect horde
connecting to new database: horde

CREATE TABLE imp_pref (
username text,
sig text,
fullname text,
replyto text,
lang varchar(30)
);
CREATE

CREATE TABLE imp_addr (
username text,
address text,
nickname text,
fullname text
);
CREATE

GRANT SELECT, INSERT, UPDATE ON imp_pref, imp_addr TO www;
CHANGE
```

```
EOF
```

Bước 4

Chúng ta phải khởi động lại PostgreSQL server cho các thay đổi có hiệu lực:

- Để khởi động lại PostgreSQL, dùng các lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/postgresql restart
Stopping postgresql service:                [ OK ]
Checking postgresql installation: looks good!
Starting postgresql service: postmaster [13474]
```

Bước 5

Sao chép và đổi tên tập tin "/home/httpd/horde/phplib/horde_phplib.inc" thành "/home/httpd/php/local.inc", sau đó hiệu chỉnh tập tin mới "local.inc" mà tập tin cấu hình phplib của bạn chứa đựng các thông số cài đặt sẽ định nghĩa các hành vi của phplib, và theo các chỉ dẫn của nó để định nghĩa nơi lưu trữ mà bạn sẽ cần khi không muốn ghi chú nữa.

```
[root@deep /]# cp /home/httpd/horde/phplib/horde_phplib.inc /home/httpd/php/local.inc
cp: overwrite `/home/httpd/php/local.inc'? y
```

Soạn thảo tập tin local.inc (vi /home/httpd/php/local.inc), sau đó không cần ghi chú nữa, đặt các dòng sau để định nghĩa SQL như cơ sở dữ liệu mặc định của bạn:

```
/* To use an SQL database, uncomment and edit the following: */
class HordeDB extends DB_Sql {
var $Host = 'localhost';
var $Database = 'horde';
var $User = 'www';
var $Password = 'some-password';
var $Port = '5432';

function halt($msg) {
  printf("<b>Database error (HordeDB):</b> %s<br>\n", $msg);
}
}

class HordeCT extends CT_Sql {
  var $database_class = 'HordeDB'; // Which database class to use...
  var $database_table = 'active_sessions'; // and find our data in this table.
}
```

Ghi chú: *Đừng quên bỏ ghi chú (uncomment) trong tập tin này loại lưu trữ mà bạn cần dùng cho Webmail IMP. Hãy ghi nhớ là chỉ bỏ ghi chú một*

kiểu. Trong trường hợp của chúng ta, chúng ta chọn dùng SQL. Cũng như các thông số mà bạn phải cài đặt cho SQL database là "var $User = '"; "var $Password =" và "var $Port=". "var $User = '" tương ứng với httpd user của bạn (trong trường hợp này là "www"), "var $Password =" tương ứng với mật khẩu cho user "www" mà bạn định nghĩa trong PostgreSQL, và "var $Port=" là số cổng IP thường dùng để kết nối với SQL database của bạn.

Bước 6

Cuối cùng là soạn thảo tập tin "/home/httpd/php/prepend.php3" và chỉ ra loại database mặc định.

Soạn thảo tập tin prepend.php3 (vi /home/httpd/php/prepend.php3) sau đó thay đổi dòng sau để định nghĩa PostgreSQL như loại database của bạn:

```
require($_PHPLIB["libdir"] . "db_mysql.inc");
```

Trở thành:

```
require($_PHPLIB["libdir"] . "db_pgsql.inc");
```

Cấu hình tập tin "php.ini" cho PHP4 của bạn

Các thông số khác mà bạn cần cấu hình là trong tập tin cấu hình PHP4 "/etc/httpd/php.ini" của bạn. Sự điều chỉnh này được yêu cầu để định nghĩa các đặc trưng nào (chẳng hạn như IMAP, PostgreSQL và những thứ khác) được nạp tự động bởi PHP4. Vì đã quyết định dùng PostgreSQL như là cơ sở dữ liệu, và cần phải dùng các đặc trưng IMAP trong phần mềm Webmail, chúng ta phải định nghĩa chúng trong tập tin cấu hình "php.ini" của PHP4.

Bước 1

Soạn thảo tập tin php.ini (vi /etc/httpd/php.ini), và thêm vào trong phần Dynamic Extensions các chọn lựa mong muốn của bạn. Trong trường hợp này, chúng ta chọn IMAP và hỗ trợ PostgreSQL:

```
extension=imap.so ; Added for IMAP support
extension=pgsql.so ; Added for PostgreSql support
extension=mysql.so ; Added for MySql support
extension=ldap.so ; Added for LDAP support
```

Bước 2

Bạn phải chỉ cho "php" biết ở đâu có thể nhìn thấy các tập tin include, các tập tin không có các đường dẫn tuyệt đối đúng để tự động prepend nội dung "prepend.php3" của phplib tới mỗi tập tin, và tắt các magic quote.

Soạn thảo tập tin php.ini (vi /etc/httpd/php.ini) và thêm vào các thông số sau vào các dòng liên quan:

```
magic_quotes_gpc = Off
auto_prepend_file = "/home/httpd/php/prepend.php3"
```

include_path = "/home/httpd/horde:/home/httpd/php"

Cấu hình Apache để nhận biết Webmail IMP

Một khi Webmail IMP đã được cài đặt trên hệ thống, chúng ta phải thêm vào các dòng sau trong tập tin "httpd.conf" của Apache để có thể định vị và sử dụng các đặc trưng của nó.

Bước 1

Soạn thảo tập tin httpd.conf (vi /ect/httpd/conf/httpd.conf) và thêm vào các dòng sau giữa các phần tags <IfModule mod_alias.c> và </IfModule>:

```
Alias /horde/ "/home/httpd/horde/"
<Directory "/home/httpd/horde">
  Options None
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>

Alias /imp/ "/home/httpd/horde/imp/"
<Directory "/home/httpd/horde/imp">
  Options None
  AllowOverride None
  Order allow,deny
  Allow from all
  </Directory>
```

Bước 2

Bạn phải khởi động lại Apache web server cho các thay đổi có hiệu lực:

- Để khởi động lại Apache, dùng các lệnh:

```
[root@deep /]# /etc/rc.d/init.d/httpd restart
Shutting down http:     [ OK ]
Starting httpd:         [ OK ]
```

Cấu hình Webmail IMP thông qua trình duyệt web của bạn

Có rất nhiều cách để cấu hình Webmail IMP, cách mà chúng ta chọn là công cụ cài đặt mới có tên là "setup.php3", nó cung cấp cho mọi người khả năng cấu hình IMP thông qua một trình duyệt web.

Bước 1

Vì các lý do bảo mật, mặc nhiên là nó được vô hiệu hoá (disabled), nhưng bạn có thể cho phép (enable) nó với các lệnh:

- Để cho phép "setup.php", dùng lệnh sau:

```
[root@deep /]# cd /home/httpd/horde/
[root@deep horde]# sh ./install.sh

Your blank configuration files have been created, please go to
the configuration utitlity at :

your install path url/setup.php3
```

Bước 2

Một khi công cụ cài đặt mới của Webmail IMP đã được cho phép, trỏ vào trình duyệt của bạn tới URL sau. http://my-web-server/horde/setup.php. Vào lúc này, bạn có thể thông qua chương trình cài đặt đồ họa và cấu hình tất cả các ảnh hưởng của IMP.

Ghi chú: <my-web-server> là địa chỉ nơi Apache web server của bạn đang hoạt động, và thư mục "/horde/" là nơi tập tin "setup.php" định vị.

Bước 3

Khi bạn đã thực hiện xong với công cụ cài đặt mới của Webmail IMP, phải chắc chắn là vô hiệu hoá nó một lần nữa vì các lý do bảo mật.

- Để vô hiệu hóa "setup.php", dùng lệnh:

```
[root@deep /]# cd /home/httpd/horde/
[root@deep horde]# sh ./secure.sh

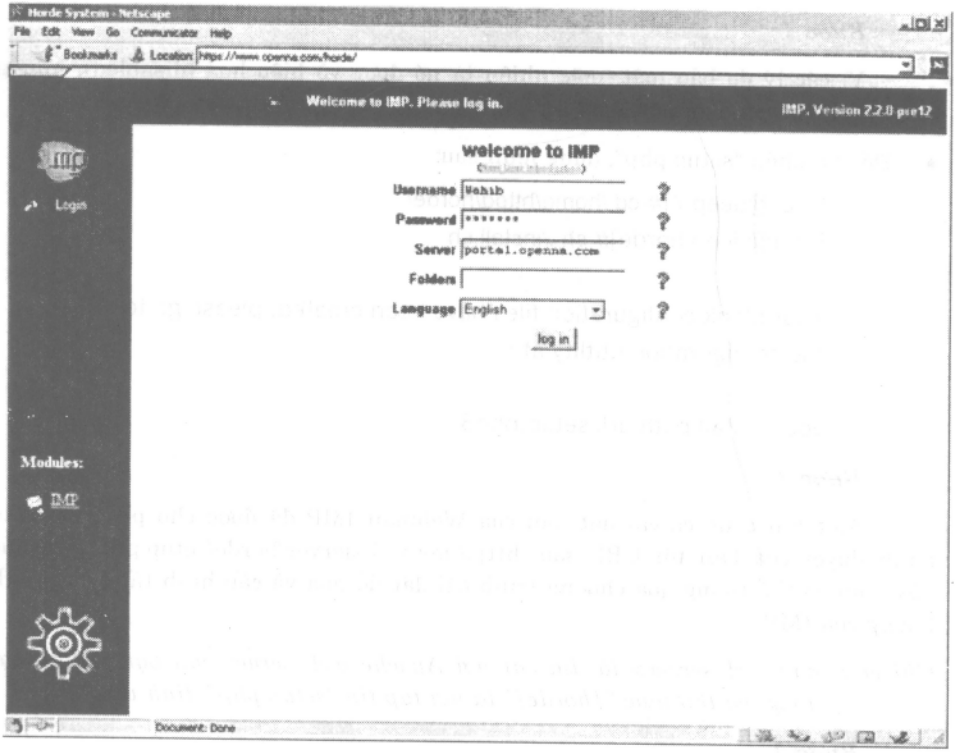
I have made your configuration files, and libraries mode 0555
which is read / execute for everyone.

And the setup.php is mode 0000 which is no access period.
```

Bước 4

Vào giai đoạn này, chúng ta phải kiểm tra xem Webmail IMP có đang làm việc trên hệ thống hay không? Để làm việc này, trỏ trình duyệt web của bạn tới địa chỉ: http://my-web-server/horde/.

Trong đó <my-web-server> là địa chỉ nơi Apache web server đang làm việc, và </horde> là thư mục chứa chương trình Webmail IMP.


Horde System - Netscape
File Edit View Go Communicator Help
Bookmarks Location: https://www.openna.com/horde/
Welcome to IMP. Please log in.
IMP, Version 2.2.8 pre12
welcome to IMP
Username Wahib
Password
Server portal.openna.com
Folders
Language English
log in
Logos
Modules:
IMP
Document: Done

Chương 21:

PHẦN MỀM SERVER
(Dịch vụ chia sẻ tập tin)

Trong chương này chúng ta sẽ đề cập đến các phần mềm sau:

- Linux Samba Server

  Cấu hình.

  Tạo một tập tin mật khẩu (password) Samba được mã hoá cho các máy client của bạn.

  Bảo mật Samba.

  Tối ưu Samba.

  Các công cụ quản trị Samba.

  Các công cụ của những người sử dụng Samba.

- Linux FTP Server

  Thiết lập một tài khoản FTP cho mỗi user không dùng shells.

  Cài đặt một môi trường sử dụng chroot.

  Cấu hình.

  Cấu hình ftpd để dùng tcp-wrappers inetd super server.

  Các công cụ quản trị FTP.

  Bảo mật FTP.



Linux Samba Server

Giới thiệu tổng quan

Các tổ chức kinh doanh lớn thường xử lý thông tin trên nhiều loại hệ điều hành khác nhau và có nhu cầu lưu giữ chúng trong một môi trường mạng trong việc chia sẻ các tập tin và máy in. Các nhân viên có thể làm việc trên các máy trạm như Linux, Microsoft Windows 95/98/NT, OS2 hay Novel và vẫn cần phải truy cập máy server trong các công việc thường ngày của họ. Một máy Linux server với sự hỗ trợ Samba có thể được dùng trong các hoạt động như thế này.

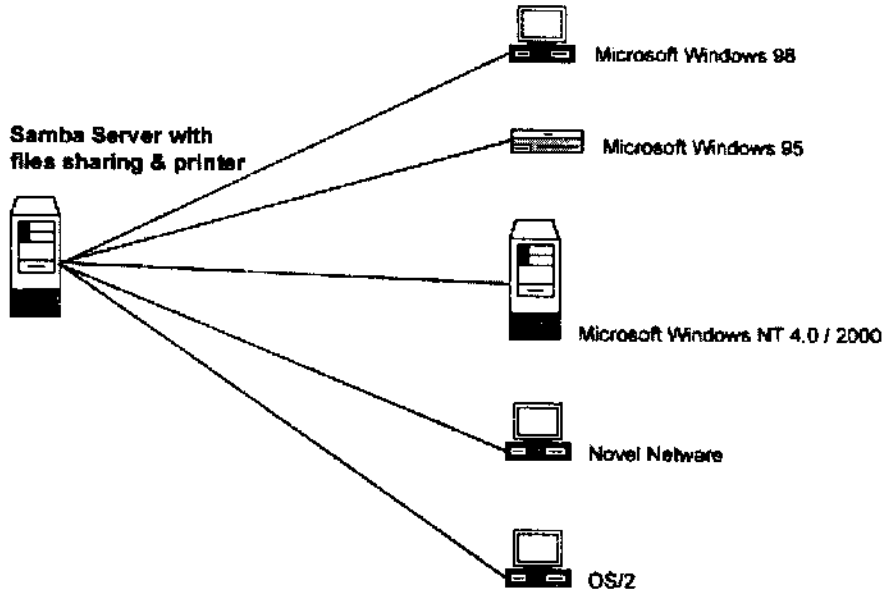
Samba là một dịch vụ mạng rất mạnh trong việc chia sẻ tập tin và máy in, nó làm việc tốt trên các hệ điều hành chủ yếu hiện nay. Khi đã được thực hiện tốt bởi người quản trị, nó sẽ nhanh hơn và bảo mật hơn các dịch vụ chia sẻ tập tin tự nhiên đã có sẵn trên các máy Microsoft Windows.

Như được giải thích trong tập tin README của Samba:

Samba là một giao thức có nhiều máy PC kết nối với nhau cùng chia sẻ các tập tin, các máy in, và các thông tin khác, chẳng hạn như liệt kê danh sách các tập tin và máy in. Các hệ điều hành mà nó hỗ trợ dịch vụ này một cách tự nhiên gồm có Windows 95/98/NT, OS/2 và Linux.

Các máy Apple Mac và một số trình duyệt web có thể giao tiếp giao thức này rất tốt. Xen kẽ với SMB gồm có Netware, NFS, AppleTalk, Banyan Vines, Decnet v.v..; Rất nhiều các dịch vụ này có các điểm thuận lợi nhưng mặc nhiên không có dịch vụ nào có cả hai, gồm sự phổ biến tài liệu và được thực hiện rộng rãi trên các máy để bàn. Phần mềm Samba gồm có một SMB server, để cung cấp Windows NT và LAN Manager-style, các dịch vụ tập tin và in ấn cho các SMB client chẳng hạn như Windows 95, Wrap Server, smbfs và những dịch vụ khác.

Xem hình vẽ bên dưới



Hướng dẫn cài đặt

Các dòng lệnh tương thích với Unix.

Đường dẫn nguồn sẽ là "/var/tmp" (cũng có thể chọn những đường dẫn khác).

Việc cài đặt được thử nghiệm trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong việc cài đặt sẽ được thực hiện với người sử dụng có quyền cao nhất trong hệ thống (root).

Phiên bản Samba là 2.0.7

Các gói cần thiết (Packages)

Samba Homepage Site: http://us1.samba.org/samba/samba.html

Samba FTP Site: 63.238.153.11

Bạn phải bảo đảm là tải về gói : samba–2.0.7.tar.gz

Dạng lưu trữ Tarballs

Trước hết hãy tạo ra danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt Samba, sau đó so sánh chúng dùng 'diff' để tìm ra các tập tin nào được đặt ở đâu. Đơn giản chạy lệnh 'find /* > Samba1' trước và 'find /* > Samba2' sau, khi bạn cài đặt phần mềm, sử dụng lệnh 'diff Samba1 Samba2 > Samba-Installed' để lấy một danh sách các thay đổi nào đó.

Sự biên dịch

Giải nén dạng lưu trữ tarball (tar.gz):

```
[root@deep /]# cp samba-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf samba-version.tar.gz
```

Cấu hình

Chuyển vào trong thư mục Samba mới, và chuyển vào trong thư mục con "source" của nó.

Bước 1

Soạn thảo tập tin **smbsh.in** (vi +3 smbwrapper/smbsh.in) và thay đổi dòng:

```
SMBW_LIBDIR=${SMBW_LIBDIR-@builddir@/smbwrapper}
```

Trở thành:

```
SMBW_LIBDIR=${SMBW_LIBDIR-/usr/bin}
```

Thay đổi này sẽ định vị lại thư mục "lib" của Samba ở trong thư mục "/usr/bin".

Bước 2

Soạn thảo tập tin Makefile.in (vi +28 Makefile.in) và thay đổi dòng:

SBINDIR = @bindir@

Trở thành:
SBINDIR = @sbindir@

VARDIR = @localstadir@

Trở thành:
VARDIR = /var/log/samba

Bước này sẽ chỉ ra rằng thư mục "sbin" của chúng ta dành cho các tập tin nhị phân sẽ được định vị trong thư mục "/usr/sbin", và thư mục "/var" cho các logfile Samba sẽ ở trong thư mục con "/var/log/samba".

Bước 3

Soạn thảo tập tin **convert_smbpasswd.in** (vi +10 convert_smbpasswd.in) và thay đổi dòng:

nawk 'BEGIN {FS=":"}
Trở thành:
gawk 'BEGIN {FS=":"}

Bước này sẽ chỉ ra việc dùng phiên bản GNU Linux của tiện ích xử lý văn bản awk thay cho phiên bản thử nghiệm của chương trình awk cho tập tin "smbpasswd".

Bước 4

Soạn thảo tập tin **smbmount.c** (vi +98 client/smbmount.c) và thay đổi dòng:

```
static void close_our_files(int client_fd)
{
int i;
for (i = 0; i < 256; i++) {
if (i == client_fd) continue;
close(i);
}
```

Trở thành:

```
static void close_our_files(int client_fd)
{
struct rlimit limits;
int i;
getrlimit(RLIMIT_NOFILE,&limits);
for (i = 0; i < limits.rlim_max; i++) {
if (i == client_fd) continue;
close(i);
}
```

Bước này sẽ tạo tập tin **smbmount.c** tương thích với thư viện glibc2.1 của Red Hat.

Biên dịch và tối ưu

Bước 1

Gõ các lệnh sau trên terminal của bạn:

```
CC="egcs" \
./configure \
--prefix=/usr \
--libdir=/etc \
--with-lockdir=/var/lock/samba \
--with-privatedir=/etc \
--with-swatdir=/usr/share/swat \
--with-pam \
--with-mmap \
--without-sambabook
```

Ghi chú: *Tùy chọn "--with-mmap" có thể cung cấp một khả năng thực hiện (performance) lớn trên một số máy, trong khi trên các máy khác nó không tạo ra sự khác nhau nào cả, và trên một số máy nó có thể giảm performance.*

Bước này bảo Samba tự thiết lập cấu hình cho việc cài đặt phần cứng riêng biệt với:

- Đưa vào cơ sở dữ liệu mật khẩu PAM hỗ trợ cho bảo mật tốt hơn.
- Đưa vào hỗ trợ MMAP dựa trên thí nghiệm để cải tiến khả năng thực hiện samba.
- Không cài đặt sách hướng dẫn mà đi kèm chung với sự phân bổ Samba.

Bước 2

Bây giờ, chúng ta phải cài đặt Samba trong máy Linux server:

```
[root@deep source]# make all
[root@deep source]# make install
[root@deep source]# install -m 755 script/mksmbpasswd.sh /usr/bin/
[root@deep source]# rm -rf /usr/share/swat/ (Làm việc này nếu bạn không thích cấu hình samba trong HTML).
[root@deep source]# rm -f /usr/sbin/swat
[root@deep source]# rm -f /usr/man/man8/swat.8
[root@deep source]# mkdir -p /var/lock/samba
[root@deep source]# mkdir -p /var/spool/samba (Chỉ yêu cầu nếu bạn có dự định sử dụng chia sẻ máy in).
```

[root@deep source]# chmod 1777 /var/spool/samba/ (Chỉ yêu cầu nếu bạn có dự định sử dụng chia sẻ máy in).

Lệnh "**install**" sẽ cài đặt script "**mksmbpasswd.sh**" trong thư mục "/usr/bin". Script này cần thiết cho việc cài đặt Samba, mọi người dùng được phép kết nối trên máy server của bạn thông qua tập tin "smbpasswd". Xem các phần sau trong tài liệu này để biết cách cài đặt và sử dụng mật khẩu Samba.

Lệnh "**rm**" sẽ xóa thư mục "/usr/share/swat" và tất cả các tập tin trong đó và nó cũng xóa chương trình nhị phân "swat" trong "/usr/sbin". Chương trình SWAT là một tiện ích cấu hình trên web, nó cho phép bạn cấu hình tập tin "smb.conf" của Samba thông qua giao diện của một trình duyệt web. Dĩ nhiên, để sử dụng tiện ích SWAT, bạn sẽ cần phải có một web server đang chạy, chẳng hạn như Apache. Tiện ích SWAT có thể mở ra một sự vi phạm bảo mật trên máy server của bạn và vì lý do bảo mật chúng ta được khuyên rằng nên xóa và không dùng nó.

Lệnh "**mkdir**" sẽ tạo một thư mục "/var/spool/samba/" trên hệ thống của bạn cho tất cả các công việc chia sẻ máy in mà bạn có thể có. Dĩ nhiên thư mục này chỉ cần thiết nếu bạn dự định dùng Samba chia sẻ máy in trên mạng LAN của bạn. Bởi vì chúng ta chưa cấu hình Samba server để dùng máy in chia sẻ, chúng ta không cần tạo thư mục này ("/var/spool/samba/") trên máy server của bạn, và chúng ta không cần phải dùng lệnh "chmod" để thay đổi bit "sticky" trong "/var/spool/samba" và như vậy chỉ có người sở hữu tập tin có thể xoá một tập cho sẵn trong thư mục này.

Dọn dẹp sau khi làm việc

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf samba-version/ samba-version.tar.gz
```

Lệnh "**rm**" sẽ xoá tất cả các tập tin nguồn mà chúng ta dùng để biên dịch và cài đặt Samba. Nó cũng sẽ xoá dạng lưu trữ nén Samba từ thư mục "/var/tmp".

Cấu hình

Tất cả các tập tin cấu hình cho các dịch vụ khác nhau thì sẽ được chỉ rõ, phụ thuộc vào nhu cầu và kiến trúc mạng của bạn. Một số người có thể cài đặt Samba server và chỉ có một client kết nối, một số người khác thì có thể cài đặt nó với 1000 kết nối.

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy Samba server, các tập tin sau được yêu cầu và phải được tạo hay được sao chép vào các thư mục thích hợp trên máy server của bạn:

  Sao chép các tập tin smb.conf và lmhosts vào trong thư mục "/etc/".

  Sao chép tập tin script smb vào trong thư mục "/etc/rc.d/init.d".

  Sao chép tập tin samba vào trong thư mục "/etc/logrotate.d".

Sao chép tập tin samba vào trong thư mục "/etc/pam.d".

Cấu hình cho tập tin "/etc/smb.conf"

Tập tin "/etc/smb.conf" là tập tin cấu hình chính cho Samba server, với tập tin này bạn có thể chỉ rõ thư mục nào mà bạn muốn truy cập từ các máy Windows, địa chỉ IP nào được cho phép ..v.v..Một ít dòng đầu tiên trong phần [global] chứa đựng các chỉ dẫn cấu hình tổng thể, phần này chung cho tất cả các chia sẻ (ngoại trừ chúng được chồng lên một việc chia sẻ cơ sở), được theo sau bởi các phần chia sẻ chung bạn muốn đưa ra. Có rất nhiều tùy chọn, điều quan trọng là đọc các tài liệu đi cùng với Samba để biết thêm chi tiết trên mỗi thông số và cài đặt khác nhau.

Ví dụ cấu hình theo sau là một tập tin cấu hình tối thiểu để Sanba làm việc với hỗ trợ mã hoá mật khẩu. Điều quan trọng cần chú ý rằng các diễn giải trong cấu hình samba này chỉ đề cập đến các tham số liên quan tới bảo mật và tối ưu, những tham số khác bạn tự nghiên cứu thêm trong tài liệu kèm theo của samba.

Trong ví dụ, chúng ta chỉ tạo một thư mục, "[tmp]", và chỉ các máy trong miền giá trị địa chỉ IP (class C) được phép kết nối tới Samba server. Chúng ta cũng không dùng khả năng chia sẻ máy in giữa Samba và Windows trên server này.

Soạn thảo tập tin smb.conf (vi /etc/smb.conf) và thêm vào/thay đổi các tham số sau:

```
[global]
workgroup = OPENNA
server string = R&D of Open Network Architecture Samba Server
encrypt passwords = True
security = user
smb passwd taäp tin = /etc/smbpasswd
log taäp tin = /var/log/samba/log.%rn
socket options = IPTOS_LOWDELAY TCP_NODELAY
domain master = Yes
local master = Yes
preferred master = Yes
os level = 65
dns proxy = No
name resolve order = lmhosts host bcast
bind interfaces only = True
interfaces = eth0 192.168.1.1
hosts deny = ALL
hosts allow = 192.168.1.4 127.0.0.1
debug level = 1
create mask = 0644
```

```
directory mask = 0755
level2 oplocks = True
read raw = no
write cache size = 262144

[homes]
comment = Home Directories
browseable = no
read only = no
invalid users = root bin daemon nobody named sys tty disk mem kmem users

[tmp]
comment = Temporary taäp tin Space
path = /tmp
read only = No
valid users = admin
invalid users = root bin daemon nobody named sys tty disk mem kmem
users
```

Bước này bảo Samba tự thiết lập chính nó cho cấu hình việc cài đặt cấu hình với:

```
[global]
```

workgroup = OPENNA

Tùy chọn "workgroup" chỉ ra workgroup của máy server của bạn sẽ xuất hiện trên màn hình khi được truy vấn bởi các máy client. Điều quan trọng là phải cùng tên workgroup trên cả hai gồm các máy client và các máy server.

server string = R&D of Open Network Architecture Samba Server

Tùy chọn "server string" chỉ rõ chuỗi mà bạn muốn trình bày cho các user của bạn trong hộp ghi chú máy in trong print manager, hoặc nhìn thấy trong windows explorer trên các máy Windows.

encrypt passwords = True

Tùy chọn "encrypt passwords" nếu được đặt là "True" thì ra lệnh cho Samba dùng các mật khẩu được mã hóa thay cho mật khẩu dạng plain-text khi đang điều đình với máy client. Chương trình Sniffer sẽ không thể phát hiện ra mật khẩu của bạn khi nó được mã hóa. Tùy chọn này phải luôn được đặt là "True" vì các lý do bảo mật.

security = user

Tùy chọn "security" nếu được đặt là "user" chỉ rõ rằng một máy client trước tiên phải "log on" với một username và password hợp lệ, bằng không kết nối sẽ bị từ chối. Điều này có nghĩa là một username và password hợp lệ cho máy client

phải tồn tại trong tập tin "/etc/passwd" trên Linux server và trong tập tin "/etc/smbpasswd" của Samba server, nếu không thì kết nối từ client sẽ bị thất bại. Xem phần "Bảo mật Samba" trong chương này để biết thêm chi tiết về tập tin "smbpasswd".

smb passwd tập tin = /etc/smbpasswd

Tùy chọn "smb passwd file" chỉ rõ đường dẫn tới tập tin "smbpasswd" được mã hóa. Tập tin "smbpasswd" là bản sao chép của tập tin "/etc/passwd" của Linux server chứa đựng username và password hợp lệ được phép kết nối tới Samba server. Phần mềm samba đọc tập tin này (smbpasswd) khi một kết nối được yêu cầu.

log tập tin = /var/log/samba/log.%m

Tùy chọn "*log file*" chỉ rõ vị trí và các tên của các logfile Samba. Với tên mở rộng "%m", nó cho phép bạn có các logfile riêng biệt cho mỗi user hay máy đã log on vào Samba server của bạn (ví dụ như log.machine.1).

socket options = IPTOS_LOWDELAY TCP_NODELAY

Tùy chọn "socket options" chỉ rõ các tham số mà bạn có thể đưa vào trong cấu hình Samba để điều chỉnh và cải tiến samba server của bạn cho khả năng thực hiện tối ưu. Mặc định chúng ta chọn để điều chỉnh kết nối cho một mạng cục bộ, và cải tiến khả năng thực hiện của Samba server trong việc truyền các tập tin.

domain master = Yes

Tùy chọn "domain master" chỉ rõ là phải đặt trình daemon Samba server "nmbd" như là một domain master browser cho workgroup được đưa ra của chính nó. Tùy chọn này thường phải đặt là "yes" chỉ trên một Samba server cho tất cả các Samba server khác trong cùng một hệ thống mạng và workgroup.

local master = Yes

Tùy chọn "local master" cho phép trình daemon Samba server "nmbd" để cố gắng trở thành một local master browser trên một subnet. Giống như tùy chọn trên, tùy chọn này thường phải đặt là "yes" trên một Samba server mà nó làm việc như một local master trên một subnet cho tất cả các Samba server khác trong hệ thống mạng của bạn.

preferred master = Yes

Tùy chọn "preferred master" định rõ và điều khiển trình daemon Samba server "nmbd" như là một master browser được ưu tiên cho workgroup của chính nó. Nhắc lại lần nữa tùy chọn này thường phải được đặt giá trị "yes" trên một server cho tất cả các samba server và client khác trong mạng.

os level = 65

Tùy chọn "os level" chỉ rõ bởi giá trị của nó có chăng trình daemon Samba server "nmbd" có một thay đổi trở thành một local master browser cho Workgroup

trong mạng nội bộ. Số 65 sẽ giành được quyền master từ bất kỳ NT server nào. Nếu bạn có một NT Server trong hệ thống của bạn, và muốn đặt Linux Samba server của bạn là local master browser cho Workgroup trong vùng local broadcast thì bạn phải set tùy chọn "os level" với giá trị là 65. Và tùy chọn này cũng phải chỉ set trên một Linux Samba server, và phải được vô hiệu hóa trên tất cả các Linux Samba server khác mà bạn có thể có trên hệ thống mạng của bạn.

dns proxy = No

Tùy chọn "dns proxy" nếu được set là "yes" chỉ ra rằng "nmbd" trình daemon khi được khởi tạo như một WINS server và tìm kiếm một tên Net BIOS mà chưa được đăng ký sẽ được xem như tên Net BIOS toàn là chữ như một tên DNS và thực hiện việc tìm kiếm với DNS server với tên mà nó đại diện cho tên máy client truy vấn. Khi chúng ta không cấu hình Samba server để làm việc như một WINS server, chúng ta không cần phải set tùy chọn này là "yes". Cài đặt tùy chọn này là "yes" sẽ làm giảm khả năng thực hiện Samba của bạn.

name resolve order = lmhosts host bcast

Tùy chọn "name resolve order" chỉ ra các dịch vụ tên (naming service) nào được dùng để mà phân giải các host name sang các địa chỉ IP, và theo thứ tự nào. Các thông số mà chúng ta chọn làm cho tập tin "lmhosts" cục bộ của samba được kiểm tra đầu tiên, và tiếp theo sau là các tập tin còn lại.

bind interfaces only = True

Tùy chọn "bind interfaces only" nếu được set là "True", thì cho phép bạn giới hạn những interface nào sẽ phục vụ các yêu cầu "smb". Đây là một đặc trưng bảo mật. Tùy chọn cấu hình "interfaces = eth0 192.168.1.1" bên dưới bổ sung cho tùy chọn này.

interfaces = eth0 192.168.1.1

Tùy chọn "interfaces" cho phép bạn chép chồng lên danh sách giao tiếp mạng mặc định mà Samba sẽ dùng trong việc duyệt (browsing), tên đăng ký và sự lưu thông NBT khác. Mặc nhiên, Samba sẽ truy vấn kernel để liệt kê tất cả các interface và dùng bất kỳ interface nào (ngoại trừ 127.0.0.1) mà nó là khả năng hỗ trợ broadcast. Với tùy chọn này, Samba chỉ sẽ lắng nghe interface "eth0" trên địa chỉ IP 192.168.1.1. Đây là một đặc trưng bảo mật, và bổ sung cho tùy chọn cấu hình "bind interfaces only = True" ở trên.

hosts deny = ALL

Tùy chọn "hosts deny" chỉ ra danh sách các host không được phép truy cập tới các dịch vụ Samba ngoại trừ các dịch vụ cụ thể nào đó có các danh sách riêng của họ được sao chép chồng lên. Để cho đơn giản, chúng ta từ chối truy cập tới tất cả các host mặc định, và cho phép các host chỉ định trong tùy chọn "hosts allow=" bên dưới.

hosts allow = 192.168.1.4 127.0.0.1

Tùy chọn "hosts allow" chỉ rõ các host nào được phép truy cập tới một dịch vụ Samba. Mặc nhiên, chúng ta cho phép các host thuộc lớp C của IP 192.168.1.4 và localhost 127.0.0.1 của chúng ta để truy cập Samba server. Ghi chú rằng localhost phải luôn được set, nếu không bạn sẽ nhận một số thông điệp lỗi.

debug level = 1

Tùy chọn "debug level" cho phép mức độ log được chỉ rõ trong tập tin "smb.conf". Nếu bạn set debug level cao hơn 2 khi đó bạn có thể bị hạn chế quan sát khả năng thực hiện. Đây là vì server làm tràn logfile sau mỗi thao tác, mà bạn có thể trả giá.

create mask = 0644

Tùy chọn "create mask" chỉ ra và set các quyền hạn cần thiết thông qua việc ánh xạ từ các mode DOS sang các quyền của UNIX. Với tùy chọn này được set là 0644, tất cả tập tin sao chép hay tạo ra từ một hệ thống Windows sang hệ thống Unix sẽ có một quyền của 0644 mặc định.

directory mask = 0755

Tùy chọn "directory mask" chỉ ra và set mode bát phân (octal) mà được dùng khi chuyển đổi từ các mode DOS sang các mode của UNIX khi tạo các thư mục UNIX. Với tùy chọn này được set là 0755, tất cả thư mục sao chép hay tạo ra từ một hệ thống Windows sang hệ thống Unix sẽ có một quyền của 0755 bởi mặc định.

level2 oplocks = True

Tùy chọn "level2 oplocks", nếu được set là "True", sẽ làm tăng tính năng thực hiện cho nhiều việc truy cập các tập tin mà không được viết một cách chung chung (chẳng hạn như các tập tin ứng dụng .EXE).

read raw = no

Tùy chọn "read raw" điều khiển có hay không server sẽ hỗ trợ các yêu cầu SMB đọc raw khi truyền dữ liệu tới các máy client. Chú ý rằng việc ánh xạ bộ nhớ là không được dùng bởi thao tác "read raw". Chính vì vậy, bạn có thể tìm việc ánh xạ bộ nhớ hiệu quả hơn nếu bạn vô hiệu hoá "read raw" dùng "read raw = no", giống như chúng ta làm.

write cache size = 262144

Tùy chọn "write cache size" cho phép Samba cải tiến tính năng thực hiện trên hệ thống nơi mà disk subsystem là một cổ nút chai (bottleneck). Giá trị của tùy chọn này được xác định rõ bằng byte, và một kích thước 262144 biểu diễn một kích thước cache 256k trên tập tin.

[tmp]

comment = Temporary File Space

Tùy chọn "comment" cho phép bạn chỉ rõ một chú thích sẽ xuất hiện kế bên một chia sẻ (share) khi một client không truy vấn tới server.

path = /tmp

Tùy chọn "path" chỉ ra một thư mục mà người sử dụng (user) dịch vụ có quyền truy cập. Trong ví dụ của chúng ta, đây là thư mục "tmp" của Linux server.

read only = No

Tùy chọn "read only" chỉ ra các user sẽ được phép chỉ đọc các tập tin hay là không. Trong ví dụ của chúng ta, đây là một tập tin cấu hình cho thư mục "tmp" cho Linux server, những người sử dụng có thể thực hiện nhiều việc hơn là chỉ đọc các tập tin.

valid users = admin

Tùy chọn "valid users" chỉ ra một danh sách các user sẽ được phép login vào dịch vụ này. Trong ví dụ này chỉ có user "admin" được phép truy cập dịch vụ.

invalid users = root bin daemon nobody named sys tty disk mem kmem users

Tùy chọn "invalid users" chỉ rõ một danh sách các user sẽ không được phép login vào dịch vụ này. Đây thật sự là việc kiểm tra để bảo đảm tuyệt đối các thông số cài đặt không vi phạm việc bảo mật của bạn. Một điều được khuyên là bạn đưa vào tất cả các user mặc định mà chạy trình daemon trên server.

Cấu hình cho tập tin "/etc/lmhosts"

Cấu hình tập tin "/etc/lmhosts" của bạn. Tập tin "lmhosts" là tập tin ánh xạ tên Samba Net BIOS tới địa chỉ IP. Nó cũng tương tự như dạng tập tin "/etc/hosts", ngoại trừ thành phần hostname phải tương ứng với dạng đặt tên Net BIOS.

Tạo tập tin **lmhosts** (touch /etc/lmhosts) và thêm vào client hosts:

```
# Sample Samba lmhosts file.
#
127.0.0.1       localhost
192.168.1.1     deep
192.168.1.4     win
```

Trong ví dụ của chúng ta, tập tin này chứa đựng ba địa chỉ IP cho việc ánh xạ tên Net BIOS. Localhost (127.0.0.1), client có tên deep (192.168.1.1) và client có tên win (192.168.1.4).

Cấu hình cho tập tin "/etc/pam.d/samba"

Cấu hình tập tin "/etc/pam.d/samba" để dùng pam authentication.

Tạo tập tin **samba** (touch /etc/pam.d/samba) và thêm vào các dòng sau:

```
Auth        required        /lib/security/pam_pwdb.so nullok shadow
Account     required        /lib/security/pam_pwdb.so
```

Cấu hình cho tập tin "/etc/logrotate.d/samba"

Cấu hình tập tin "/etc/logrotate.d/samba" để quay vòng tự động mỗi tuần các logfile của bạn.

Tạo tập tin **samba** (touch /etc/logrotate.d/samba) và thêm vào các dòng sau:

```
/var/log/samba/log.nmb {
    notifempty
    missingok
    postrotate
    /usr/bin/killall -HUP nmbd
    endrotate
}

/var/log/samba/log.smb {
    notifempty
    missingok
    postrotate
    /usr/bin/killall -HUP smbd
    endrotate
}
```

Tạo một tập tin mật khẩu (password) Samba được mã hoá cho các máy client của bạn

Tập tin "/etc/smbpasswd" là một tập tin mật khẩu được mã hoá. Nó chứa đựng username; Unix UID và các mật khẩu SMB được băm của người sử dụng được phép vào Samba server của bạn, cũng như thông tin cờ hiệu tài khoản sử dụng và thời gian mà mật khẩu thay đổi lần sau cùng. Điều quan trọng tạo tập tin mật khẩu này và đưa vào tất cả mọi người dùng được phép truy cập trước khi các máy client của bạn cố gắng kết nối vào Samba server. Không có bước này, sẽ không có một client nào có thể kết nối tới Samba server của bạn.

Bước 1

Để tạo một tài khoản sử dụng Samba, đầu tiên bạn phải có một tài khoản Linux hợp lệ cho chúng, vì vậy tạo trong tập tin "etc/passwd" của bạn tất cả các user mà bạn muốn kết nối tới Samba server của bạn đầu tiên, trước khi sinh ra tập tin "smbpasswd" của Samba.

- Để thêm một user mới vào tập tin "/etc/passwd", dùng các lệnh sau:

```
[root@deep /]# useradd smbclient
```

- Để thêm vào mật khẩu cho các user trong tập tin "/etc/passwd", dùng các lệnh sau:

```
[root@deep /]# passwd smbclient
Changing password for user smbclient
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully
```

Bước 2

Một khi đã thêm tất cả các client samba vào trong tập tin "/etc/passwd" của chúng ta trên máy Linux server, bây giờ chúng ta có thể sinh ra tập tin "smbpasswd" từ tập tin "/etc/passwd".

- Để sinh ra tập tin "smbpasswd" từ tập tin "passwd", dùng các lệnh sau:

```
[root@deep /]# cat /etc/passwd | mksmbpasswd.sh > /etc/smbpasswd
```

Bước 3

Bước cuối cùng chúng ta phải thực hiện là tạo một tài khoản người dùng Samba trong tập tin "/etc/smbpasswd" trước khi chúng ta có thể dùng nó.

- Để tạo tài khoản người sử dụng Samba. Dùng các lệnh sau:

```
[root@deep /]# smbpasswd -a smbclient (nên nhớ rằng "smbclient" phải là một tài khoản Linux hợp lệ).
New SMB password:
Retype new SMB password:
Added user smbclient.
Password changed for user smbclient.
```

Bước 4

Đừng quên thay đổi quyền của tập tin mới "smbpasswd" của bạn để chỉ có thể đọc và ghi bởi super-user "root", và không có quyền gì cho group và những người dùng khác (0600/-rw-------). Đây là một thước đo bảo mật.

```
[root@deep /]# chmod 600 /etc/smbpasswd
[root@deep /]# testparm
```

Lệnh trên sẽ phát hiện lỗi trong tập tin cấu hình smb.conf

Ghi chú: *Xem ENCRYPTION.txt trong samba/doc/texts/ để biết thêm thông tin.*

Cấu hình tập tin script "/etc/rc.d/init.d/smb"

Cấu hình tập tin script "/etc/rc.d/init.d/smb" của bạn để khởi động và dừng Samba smbd daemons Server một cách tự động.

Tạo tập tin script smb (touch /etc/rc.d/init.d/smb) và thêm vào các dòng sau:

```
#!/bin/sh
#
```

```
# chkconfig: - 91 35
# description: Starts and stops the Samba smbd and nmbd daemons \
# used to provide SMB network services.

# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0

# Check that smb.conf exists.
[ -f /etc/smb.conf ] || exit 0

RETVAL=0

# See how we were called.
case "$1" in
  start)
        echo -n "Starting SMB services: "
        daemon smbd -D
        RETVAL=$?
        echo
        echo -n "Starting NMB services: "
        daemon nmbd -D
        RETVAL2=$?
        echo
        [ $RETVAL -eq 0 -a $RETVAL2 -eq 0 ] && touch /var/lock/subsys/smb ||
        \
        RETVAL=1
        ;;
  stop)
        echo -n "Shutting down SMB services: "
        killproc smbd
        RETVAL=$?
        echo
        echo -n "Shutting down NMB services: "
        killproc nmbd
```

```
        RETVAL2=$?
        [ $RETVAL -eq 0 -a $RETVAL2 -eq 0 ] && rm -f /var/lock/subsys/smb
        echo ""
        ;;
restart)
        $0 stop
        $0 start
        RETVAL=$?
        ;;
reload)
        echo -n "Reloading smb.conf file: "
        killproc -HUP smbd
        RETVAL=$?
        echo
        ;;
status)
        status smbd
        status nmbd
        RETVAL=$?
        ;;
*)
        echo "Usage: $0 {start|stop|restart|status}"
        exit 1
esac

exit $RETVAL
```

Bây giờ, tạo script có thể thi hành và thay đổi các quyền mặc định:

```
[root@deep /]# chmod 700 /etc/rc.d/init.d/smb
```

Tạo các liên kết bằng biểu tượng rc.d cho Samba với lệnh:

```
[root@deep /]# chkconfig --add smb
```

Script samba sẽ không tự động khởi động smbd và nmbd daemon khi bạn khởi động lại server. Bạn có thể thay đổi nó để thực hiện việc này mặc định bằng cách thi hành lệnh sau:

```
[root@deep /]# chkconfig --level 345 smb on
```

Khởi động Samba server thực hiện bằng tay với lệnh:

```
[root@deep /]# /etc/rc.d/init.d/smb start
Starting SMB services:                     [ OK ]
Starting NMB services:                     [ OK ]
```

Bảo mật Samba

Làm cho các tập tin cấu hình quan trọng không thể thay đổi

Bit không thể thay đổi có thể được dùng để ngăn chặn ngẫu nhiên việc xóa hay chép chồng một tập tin mà nó phải được bảo vệ. Nó cũng ngăn chặn một số người tạo một biểu tượng liên kết tới tập tin này. Một khi các tập tin "smb.conf" và "lmhosts" đã được cấu hình, đó là một ý tưởng hay để chúng không thể thay đổi với lệnh:

```
[root@deep /]# chattr +i /etc/smb.conf
[root@deep /]# chattr +i /etc/lmhosts
```

Tối ưu Samba

Cài đặt tham số Samba "wide links=" trong tập tin cấu hình

Một thiếu sót lớn nếu cài đặt tham số Samba "wide links" là "no" trong tập tin cấu hình Samba "/etc/smb.conf". Tùy chọn này, nếu được đặt là "no", nó sẽ bảo Samba là không có các biểu tượng liên kết bên ngoài của một vùng được dự kiến để xuất ra một điểm chia sẻ.

Sự điều chỉnh buffuer cache

Việc hiệu chỉnh các thông số cache-tuning của filesystem có thể cải tiến khả năng thực hiện file-serving của Linux một cách đáng kể—gấp một đến hai lần. Linux sẽ cố sử dụng bộ nhớ mà không được dùng cho bất kỳ mục đích nào khác cho filesystem caching. Một trình daemon đặc biệt, được gọi là "bdflush", sẽ làm tràn các buffer "dirty" một cách định kỳ (các buffer chứa đựng dữ liệu filesystem được thay đổi hoặc metadata) vào đĩa.

Bí quyết để khả năng thực hiện tốt là lưu giữ nhiều dữ liệu trong bộ nhớ trong thời gian dài nếu có thể. Việc ghi ra đĩa là phần chậm nhất của bất kỳ filesystem nào. Nếu bạn biết rằng filesystem sẽ được dùng rất nhiều, khi đó bạn có thể điều chỉnh tiến trình này cho Linux Samba. Với nhiều tùy chọn có thể điều chỉnh kernel, việc này được thực hiện rất tốt bằng cách ghi vào các tập tin đặc biệt trong filesystem "/proc". Có mẹo vặt là bạn phải bảo Linux là bạn cần thực hiện điều đó. Thực hiện như vậy bằng cách thi hành lệnh trên một Linux kernel 2.2.

Cài đặt mặc định cho các thông số "bdflush" trên Red Hat Linux là:

```
"40 500 64 256 500 3000 500 1884 2"
```

- Để thay đổi các gía trị của bdflush, gõ lệnh sau vào terminal của bạn:

Trên hệ thống Linux 6.1

```
[root@deep /]# echo "80 500 64 64 15 6000 6000 1884 2"
>/proc/sys/vm/bdflush
```

Bạn có thể thêm các lệnh trên vào tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó lần nữa ở lần tới khi bạn khởi động lại hệ thống của bạn.

Trên hệ thống Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" tập tin và thêm vào dòng sau:

```
# Improve taap tin system performance
vm.bdflush = 80 500 64 64 15 6000 6000 1884 2
```

Bạn phải khởi động lại hệ thống mạng cho việc thay đổi có hiệu lực. Lệnh sau sẽ khởi động lại mạng :

- Để khởi động lại tất cả các thiết bị mạng thực hiện bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init d/network restart
Setting network parameters          [ OK ]
Bringing up interface lo            [ OK ]
Bringing up interface eth0          [ OK ]
Bringing up interface eth1          [ OK ]
```

Dòng này bảo với "bdflush" đừng lo lắng về việc ghi các dirty block ra đĩa cho tới khi buffer cache của filesystem đầy 80% (80). Các giá trị khác điều chỉnh nhiều điều khác như số lượng buffer để ghi ra trong một thao tác đĩa (500), cho phép các dirty buffer tồn tại bao lâu trong kernel (60*HZ), v.v.. Bạn có thể tìm đầy đủ các tài liệu chi tiết trong kernel 2.2 trong tập tin "linux/Documentation/sysctl/vm.txt". và bạn cũng có thể xem lại chương 4, "Tổng quan về tối ưu hệ thống" để biết thêm chi tiết.

Điều chỉnh buffermem

Gợi ý một điều chỉnh trợ giúp khác là bảo hệ điều hành Linux các việc sau: Sử dụng tối thiểu 60% bộ nhớ dành cho buffer cache; chỉ cắt xén bớt khi tỉ lệ phần trăm của bộ nhớ được dùng cho buffer cache chiếm hơn 10% (tham số này bây giờ không được dùng); và cho phép buffer cache tăng tới 60% của tất cả bộ nhớ (bây giờ tham số này cũng không được dùng).

Cài đặt mặc định cho các thông số "buffermem" trên Red Hat Linux là:

```
"2 10 60"
```

- Để thay đổi các gía trị của buffermem, gõ lệnh sau vào terminal của bạn:

Trên hệ thống Linux 6.1

```
[root@deep /]# echo "2 10 60" >/proc/sys/vm/buffermem
```

Bạn có thể thêm các lệnh trên vào tập tin script "/etc/rc.d/rc.local" và bạn sẽ không phải gõ nó lần nữa ở lần tới khi bạn khởi động lại hệ thống của bạn. Bạn có thể tìm đầy đủ các tài liệu chi tiết trong kernel 2.2 trong tập tin "linux/Documentation/sysctl/vm.txt". và bạn cũng có thể xem lại chương "Tổng quan về tối ưu hệ thống" để biết thêm chi tiết.

Trên hệ thống Linux 6.2

Soạn thảo tập tin "/etc/sysctl.conf" tập tin và thêm vào dòng sau:

```
# Improve taäp tin system performance
vm.bdflush = 60 10 60
```

Bạn phải khởi động lại hệ thống mạng cho việc thay đổi có hiệu lực. Lệnh sau sẽ khởi động lại mạng :

- Để khởi động lại tất cả các thiết bị mạng thực hiện bằng tay trên hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Setting network parameters          [ OK ]
Bringing up interface lo            [ OK ]
Bringing up interface eth0          [ OK ]
Bringing up interface eth1          [ OK ]
```

Nhắc lại rằng hai tham số sau cùng (10 và 60) thì không được dùng bởi hệ thống, như vậy chúng ta không cần phải thay đổi các tham số mặc định.

Tài liệu bổ sung

Để biết thêm chi tiết, có nhiều trang man mà bạn cần phải đọc:

```
$ man Samba (7)          - A Windows SMB/CIFS fileserver for UNIX
$ man smb.conf (5)       - The configuration taäp tin for the Samba suite
$ man smbclient (1)      - ftp-like client to access SMB/CIFS resources on servers
$ man smbd (8)           - server to provide SMB/CIFS services to clients
$ man smbmnt (8)         - mount smb taäp tin system
$ man smbmount (8)       - mount smb taäp tin system
$ man smbpasswd (5)      - The Samba encrypted password file
$ man smbpasswd (8)      - change a users SMB password
$ man smbrun (1)         - interface program between smbd and external programs
$ man smbsh (1)          - Allows access to Windows NT filesystem using UNIX commands
$ man smbstatus (1)      - report on current Samba connections
$ man smbtar (1)         - shell script for backing up SMB shares directly to UNIX tape drives
$ man smbumount (8)      - umount for normal users
$ man testparm (1)       - check an smb.conf configuration taäp tin for internal correctness
$ man testprns (1)       - check printer name for validity with smbd
```

Các công cụ quản trị Samba

Các lệnh được liệt kê bên dưới là một số lệnh mà chúng ta thường dùng, nhưng hiện tại đã có nhiều lệnh hơn. Xem lại các trang man và tài liệu để biết thêm thông tin chi tiết.

Smbstatus

Tiện ích smbstatus là một chương trình rất đơn giản để liệt kê các kết nối Samba hiện hành.

- Để báo cáo các kết nối Samba hiện hành, dùng lệnh sau:

```
[root@deep /]# smbstatus

Samba version 2.0.7
Service uid gid pid machine
----------------------------------------

tmp  webmaster webmaster 3995  gate (192.168.1.3) Sat Sep 25 19:40:54 1999

No locked files

Share mode memory usage (bytes):
1048464(99%) free + 56(0%) used + 56(0%) overhead = 1048576(100%) total
```

Các công cụ của những người sử dụng Samba

Các lệnh được liệt kê bên dưới là một số lệnh mà chúng ta thường dùng, nhưng hiện tại đã có nhiều lệnh hơn. Xem lại các trang man và tài liệu để biết thêm thông tin chi tiết.

smbclient

Chương trình tiện ích "smbclient" cho Samba làm việc rất giống như giao tiếp của chương trình FTP. Chương trình nhỏ này cho phép bạn lấy các tập tin từ server về máy cục bộ, chuyển các tập tin từ máy cục bộ tới máy server, nhận biết thông tin về thư mục từ server ... v.v..

- Để kết nối tới một máy Windows với tiện ích smbclient, dùng lệnh sau:

```
[root@deep /]# smbclient //sbmserver/sharename -U smbclient
[root@deep /]# smbclient //gate/tmp -U smbclient
Password:
Domain=[OPENNA] OS=[Windows NT 4.0] Server=[NT LAN Manager 4.0]
smb: \> ls
                                    D 0 Tue Mar 14 15:31:50 2000
  ..                                D 0 Tue Mar 14 15:31:50 2000
  PostgreSQL                        D 0 Tue Mar 14 15:32:22 2000
  Squid                             D 0 Tue Mar 14 15:32:28 2000
  Imap                              D 0 Tue Mar 14 15:32:38 2000
  E_comm                            D 0 Tue Mar 14 15:32:42 2000
  StackGuard.pdf                    A 61440 Tue Dec 21 20:41:34 1999
  installation-without-XFree86      A 448 Tue Dec 21 20:41:28 1999
  lcap-0_0_3-2_src.rpm              A 13481 Thu Jan 13 01:50:12 2000
  mirc561t.exe                      A 948224 Tue Dec 21 20:41:54 1999
  65510 blocks of size 32768. 5295 blocks available
smb: \>
```

Trong đó "//smbserver" là tên của server mà bạn muốn kết nối tới. "/sharename" là thư mục trên server mà bạn muốn kết nối tới, và "smbclient" là username trên máy này.

Các tập tin được cài đặt

> /etc/rc.d/init.d/smb
> /etc/rc.d/rc0.d/K35smb
> /etc/rc.d/rc1.d/K35smb
> /etc/rc.d/rc2.d/K35smb
> /etc/rc.d/rc3.d/S91smb
> /etc/rc.d/rc4.d/S91smb
> /etc/rc.d/rc5.d/S91smb
> /etc/rc.d/rc6.d/K35smb
> /etc/pam.d/samba
> /etc/logrotate.d/samba
> /etc/codepages
> /etc/codepages/codepage.437
> /etc/codepages/unicode_map.437
> /etc/codepages/codepage.737
> /etc/codepages/unicode_map.737
> /etc/codepages/codepage.775
> /etc/codepages/codepage.850
> /etc/codepages/unicode_map.850
> /etc/codepages/codepage.852
> /etc/codepages/unicode_map.852
> /etc/codepages/codepage.861
> /etc/codepages/unicode_map.861
> /etc/codepages/codepage.932
> /etc/codepages/unicode_map.932
> /etc/codepages/codepage.866
> /etc/codepages/unicode_map.866
> /etc/codepages/codepage.949
> /etc/gshadow
> /usr/bin/smbclient
> /usr/bin/smbspool
> /usr/bin/testparm
> /usr/bin/testprns
> /usr/bin/smbstatus
> /usr/bin/rpcclient
> /usr/bin/smbpasswd
> /usr/bin/make_smbcodepage
> /usr/bin/make_unicodemap
> /usr/bin/nmblookup
> /usr/bin/make_printerdef
> /usr/bin/smbtar
> /usr/bin/addtosmbpass
> /usr/bin/convert_smbpasswd
> /usr/bin/mksmbpasswd.sh
> /usr/man/man1/make_smbcodepage.1
> /usr/man/man1/make_unicodemap.1
> /usr/man/man1/nmblookup.1
> /usr/man/man1/smbclient.1
> /usr/man/man1/smbrun.1
> /usr/man/man1/smbsh.1
> /usr/man/man1/smbstatus.1
> /usr/man/man1/smbtar.1
> /usr/man/man1/testparm.1
> /usr/man/man1/testprns.1
> /usr/man/man5/lmhosts.5

> /etc/codepages/unicode_map.949
> /etc/codepages/codepage.950
> /etc/codepages/unicode_map.950
> /etc/codepages/codepage.936
> /etc/codepages/unicode_map.936
> /etc/codepages/codepage.1251
> /etc/codepages/unicode_map.ISO8859-1
> /etc/codepages/unicode_map.ISO8859-2
> /etc/codepages/unicode_map.ISO8859-5
> /etc/codepages/unicode_map.ISO8859-7
> /etc/codepages/unicode_map.KOI8-R
> /etc/lmhosts
> /etc/smb.conf
> /etc/smbpasswd
> /usr/man/man5/smb.conf.5
> /usr/man/man5/smbpasswd.5
> /usr/man/man7/samba.7
> /usr/man/man8/nmbd.8
> /usr/man/man8/smbd.8
> /usr/man/man8/smbmnt.8
> /usr/man/man8/smbmount.8
> /usr/man/man8/smbpasswd.8
> /usr/man/man8/smbspool.8
> /usr/man/man8/smbumount.8
> /usr/sbin/smbd
> /usr/sbin/nmbd
> /var/log/samba
> /var/lock/samba

Linux FTP Server

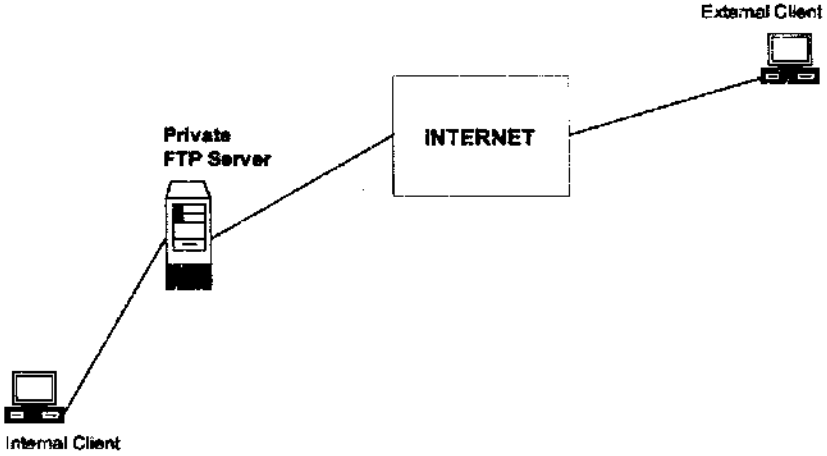
Tổng quan

Sử dụng File Transfer Protocol (FTP) là một trong những cách thông dụng nhất để truyền nhận tập tin từ máy này sang máy khác. Thay vì các máy khách và máy chủ được lập trình dựa trên platform thông dụng để truyền nhận tập tin, thì việc sử dụng FTP là cách truyền thông thông dụng nhất để thực hiện việc truyền tập tin mà không dựa trên platform nào cả.

Có rất nhiều cách cấu hình cho máy chủ FTP của bạn. Một trong số ấy là private user-only site, đây là kiểu cấu hình mặc định của FTP server, một private FTP server chỉ cho phép các users trên hệ thống Linux kết nối thông qua FTP và truy xuất những tập tin của chính user đó mà thôi.

Cách thứ hai là anonymous FTP server. Một anonymous FTP server cho phép bất cứ ai trên mạng kết nối tới nó và truyền nhận tập tin không cần một tài khoản nào cả. Mặc dù đây là cách tiềm ẩn nguy cơ về mặt an toàn mạng khi cài đặt theo cách này, nhưng có thể phòng tránh bằng cách chỉ cho phép truy xuất đến những thư mục nào đó trên hệ thống.

Cấu hình chúng ta đề cập trong quyển sách này là một FTP server cho phép FTP đến những vùng bán an toàn (semi-secure) của một hệ thống tập tin Unix (chroot'd Guest FTP access). Cấu hình này cho phép các users được quyền truy xuất những thư mục trên FTP server mà không cho phép chúng vào những nơi có quyền cao hơn. Đây là kiểu cài đặt an toàn nhất dành cho FTP server.



Hướng dẫn cài đặt

Những lệnh này tương thích với hệ điều hành Unix.

Đường dẫn nguồn sẽ là "/var/tmp" (cũng có thể chọn những đường dẫn khác).

Sự cài đặt này đã được kiểm tra trên Red Hat Linux 6.1 và 6.2.

Tất cả các bước trong quá trình cài đặt sẽ dùng tài khoản của super-user "root".

Phiên bản wu-ftpd là 2.6.0.

Các gói dữ liệu(packages)

Trang chủ của Wu-ftpd: http://www.wu-ftpd.org/.

FTP Site của Wu-ftpd: 205.133.13.68.

Bạn phải chắc chắn đã tải xuống : Wu-ftpd-2.6.0.tar.gz.

Biên Dịch

Giải nén tập tin tarball (tar.gz)

```
[root@deep /]# cp wu-ftpd-version.tar.gz /var/tmp
[root@deep /]# cd /var/tmp
[root@deep tmp]# tar xzpf wu-ftpd-version.tar.gz
```

Cấu hình và tối ưu

Di chuyển vào thư mục mới của Wu-ftpd và gõ lệnh sau trên terminal của bạn.

Bước 1

Hiệu chỉnh tập tin **ftpcount.c** (vi +241 src/ftpcount.c) và thay đổi dòng.

```
#if defined (LINUX)
```

Để thành:

```
#if defined (LINUX_BUT_NOT_RED HAT_6_0)
```

Bước 2

Hiệu chỉnh tập tin **pathnames.h.in** (vi +42 src/pathnames.h.in) và thay đổi dòng:

```
#define _PATH_EXECPATH "/bin/ftp-exec"
```

Để thành:

```
#define _PATH_EXECPATH "/usr/bin/ftp-exec"
```

Với Red Hat Linux, chúng ta thay đổi thư mục "/bin" của "ftp-exec" thành "/usr/bin"

Bước 3

Gõ những lệnh sau trên terminal để cấu hình Wu-ftpd:

```
CC="egcs" \
CFLAGS="-O9 -funroll-loops -ffast-math -malign-double -mcpu=pentiumpro -
march=pentiumpro -fomitframe-
pointer -fno-exceptions" \
./configure \
--prefix=/usr \
```

```
--sysconfdir=/etc \
--localstatedir=/var \
--disable-dnsretry \
--enable-quota \
--enable-pam \
--disable-daemon \
--disable-newlines \
--disable-virtual \
--disable-plsm \
--disable-pasvip \
--disable-anonymous \
--enable-ls \
--enable-numericuid
```

Điều này chứng tỏ rằng Wu-ftpd tự thiết lập phần cứng dành cho nó:

+ Không thử lại khi gặp thất bại trong quá trình lookup DNS để cải thiện việc thực thi.
+ Thêm QUOTA dùng để hỗ trợ tính bảo mật (nếu hệ điều hành của bạn hỗ trợ).
+ Thêm PAM dùng để hỗ trợ tính bảo mật.
+ Không cho phép chạy như là standalone daemon để cho phép FTPD được điều khiển bởi TCP-Wrappers.
+ Không để những dòng trống.
+ Không hỗ trợ những máy chủ ảo.
+ Không hiển thị những thông điệp PID lock sleep (dùng khi máy bận).
+ Không yêu cầu cùng một địa chỉ IP dành cho kiểu kết nối thụ động.
+ Sử dụng lệnh "ls" mới của Wu-ftpd thay vì "ls" mặc định của Linux để được an toàn hơn.
+ Lệnh "ls" liệt kê danh sách UID thay vì username để thực hiện được tốt hơn (nhanh hơn).

Bước 4

Bây giờ, chúng ta cài đặt Wu-ftpd trên hệ thống Linux server:

```
[root@deep wu-ftpd-2.6.0]# make
[root@deep wu-ftpd-2.6.0]# make install
[root@deep wu-ftpd-2.6.0]# install -m 755 util/xferstats /usr/sbin/
[root@deep wu-ftpd-2.6.0]# touch /var/log/xferlog
[root@deep wu-ftpd-2.6.0]# chmod 600 /var/log/xferlog
[root@deep wu-ftpd-2.6.0]# cd /usr/sbin/
```

```
[root@deep sbin]# ln -sf in.ftpd /usr/sbin/wu.ftpd
[root@deep sbin]# ln -sf in.ftpd /usr/sbin/in.wuftpd
[root@deep sbin]# strip /usr/bin/ftpcount
[root@deep sbin]# strip /usr/bin/ftpwho
[root@deep sbin]# strip /usr/sbin/in ftpd
[root@deep sbin]# strip /usr/sbin/ftpshut
[root@deep sbin]# strip /usr/sbin/ckconfig
[root@deep sbin]# strip /usr/sbin/ftprestart
```

Những lệnh trên "**make**" và "**make install**" sẽ cấu hình phần mềm để chắc rằng hệ thống của bạn cần những chức năng và thư viện để biên dịch các gói một cách thành công, biên dịch các tập tin nguồn trở thành những tập tin nhị phân có thể thực thi được và sau đó cài đặt những tập tin nhị phân này và các tập tin hỗ trợ khác vào những vị trí thích hợp.

"**install -m**" sẽ cài đặt chương trình "xferstats" dùng để xem trạng thái thông tin về việc truyền nhận tập tin và lệnh "**touch**" sẽ tạo tập tin log "xferstats" trong thư mục "/var/log". "**chmod**" sẽ thay đổi mode của các tập tin "xferlog" để những tập tin chỉ có thể được đọc và được ghi bởi superuser "root". Sau đó chúng ta tạo biểu tượng kết nối cho các tập tin nhị phân "in.ftpd" và sau cùng gỡ bỏ tất cả các tập tin nhị phân liên quan tới Wu-ftpd để làm giảm kích thước chương trình và cũng để thực thi tốt hơn.

Dọn dẹp sau quá trình cài đặt

```
[root@deep /]# cd /var/tmp
[root@deep tmp]# rm -rf wu-ftpd-version/ wu-ftpd-version.tar.gz
```

Lệnh "**rm**" sẽ xoá tất cả các tập tin nguồn mà chúng ta dùng để cài đặt Wu-ftpd. Nó cũng sẽ gỡ bỏ tập tin nén Wu-ftpd khỏi thư mục "/var/tmp".

Thiết lập một tài khoản FTP cho mỗi user không có shells

Đây là điều quan trọng mang lại cho các user FTP không có real shell account trên hệ thống Linux. Theo cách này nếu bất kỳ lý do gì một ai đó có thể thoát ra môi trường FTP Chrooted một cách thành công. Nó không có thể có khả năng thực thi bất kỳ các tác vụ người dùng đưa ra vì không chứa bash shell. Đầu tiên, hãy tạo các user mới cho mục đích này. Những user này sẽ được phép kết nối tới FTP server. Môi trường "chroot" hoạt động bằng cách phân chia một tài khoản người dùng bình thường với một tài khoản có quyền truy xuất không hạn chế.

Bước 1

Sử dụng lệnh sau để tạo users trong tập tin "/etc/passwd". Bước này phải được thực hiện mỗi khi bạn thêm mới một user và cho phép nó truy xuất tới FTP server.

```
[root@deep /]# mkdir /home/ftp
```

```
[root@deep /]# useradd -d /home/ftp/ftpadmin/ -s /dev/null ftpadmin > /dev/null
2>&1

[root@deep /]# passwd ftpadmin

Changing password for user ftpadmin

New UNIX password:

Retype new UNIX password:

passwd: all authentication tokens updated successfully
```

Lệnh "**mkdir**" sẽ tạo thư mục "ftp" dưới thư mục "/home" để điều khiển tất cả các thư mục gốc dành cho các user được đặt trên máy chủ. Lệnh "useradd" sẽ tạo một user mới có tên là "ftpadmin" trên máy chủ Linux. Sau đó lệnh "passwd" sẽ đặt mật khẩu cho user này. Nếu thư mục "/home/ftp" đã được tạo, bạn sẽ không dùng lệnh này nữa mỗi khi thêm mới một user FTP.

Bước 2

Hiệu chỉnh tập tin "/etc/shell" (vi /etc/shells) và thêm một shell không tồn tại (non-existent shell) có tên là "null", ví dụ shell giả mạo này sẽ giới hạn quyền truy xuất đến FTP server.

```
[root@deep /]# vi /etc/shells
/bin/bash
/bin/sh
/bin/ash
/bin/bsh
/bin/tcsh
/bin/csh
/dev/null  (Đây là shell không tồn tại chúng ta đã thêm vào).
```

Ghi chú: *Với Red Hat Linux, tồn tại một thiết bị đặc biệt mang tên (/dev/null) dành cho mục đích như thế này.*

Bước 3

Bây giờ, hãy hiệu chỉnh tập tin "/etc/passwd" và thêm vào dòng "/./" dùng để phân chia thư mục "/home/ftp" với thư mục "/ftpadmin", nơi mà user "/ftpadmin" ắt hẳn sẽ chuyển đổi tới một cách tự động. Bước này phải được thực hiện cho mỗi user FTP bạn thêm vào tập tin "passwd".

Hiệu chỉnh tập tin **passwd** (vi /etc/passwd) và thêm mới/ thay đổi dòng dành cho user "ftpadmin":

```
ftpadmin:x:502:502::/home/ftp/ftpadmin/:/dev/null
```

Để thành:

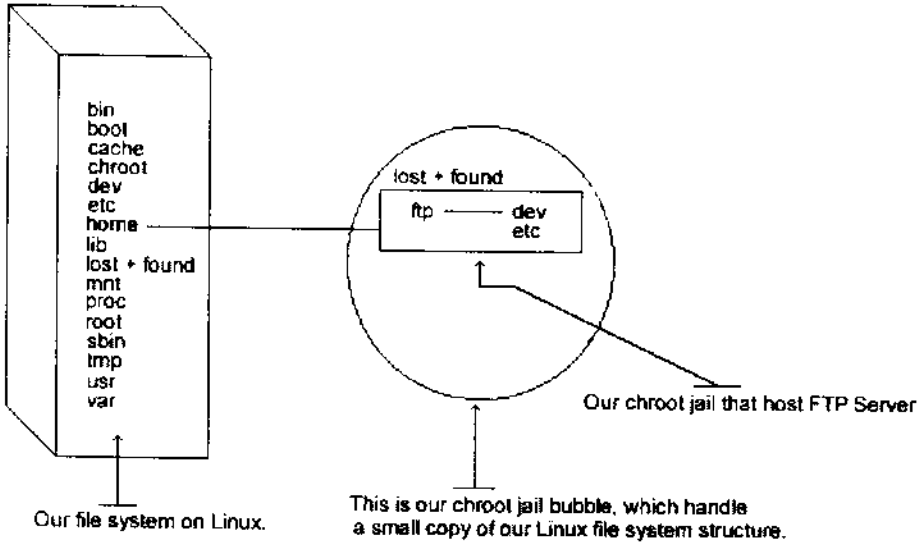
```
ftpadmin:x:502:502::/home/ftp/./ftpadmin/:/dev/null
```

^^

Tài khoản là "ftpadmin", nhưng bạn sẽ lưu ý đường dẫn đến thư mục gốc là một bit lẻ. Phần đầu của "/home/ftp/" chỉ rõ tập tin hệ thống ắt hẳn xem như thư mục gốc mới. Dấu chấm "." phân chia thư mục để chúng tự chuyển đổi thư mục thành "/ftpadmin". Một lần nữa phần "/dev/null" làm mất quyền đăng nhập của nó để trở thành một user bình thường. Với sự thay đổi này bây giờ user "ftpadmin" có một fake shell thay vì kết quả phải là một real shell với quyền truy cập giới hạn trên hệ thống.

Cài đặt một môi trường sử dụng chroot

Về cơ bản những gì mà chúng ta đang làm là tạo một bộ khung tập tin hệ thống root (skeleton root tập tin system) với đầy đủ các thành phần cần thiết (các tập tin nhị phân, mật khẩu ...) để cho phép Unix tạo một chroot khi user đăng nhập vào. Lưu ý rằng, nếu bạn dùng tùy chọn "**--enable-ls**" trong quá trình biên dịch, các thư mục "/home/ftp/bin" và "/home/ftp/lib" không được yêu cầu từ khi tùy chọn mới này cho phép Wu-ftpd sử dụng chức năng "ls" của chúng. Chúng tôi vẫn sử dụng phương pháp cũ dành cho những ai yêu thích, bằng cách sao chép từ thư mục "/bin/ls" đến thư mục của FTP ("/home/ftp/bin") và tạo các tập tin thư viện có liên quan của "ls".



Sau đây là những bước cần thiết để thực thi chương trình Wu-ftpd trong một chroot jail:

Bước 1

Đầu tiên hãy tạo tất cả các thư mục cần thiết dành cho môi trường chrooted, như trình bày dưới đây:

```
[root@deep /]# mkdir /home/ftp/dev
[root@deep /]# mkdir /home/ftp/etc
```

```
[root@deep /]# mkdir /home/ftp/bin
[root@deep /]# mkdir /home/ftp/lib
```

Bước 2

Thay đổi quyền 0511 cho những thư mục mới, dùng cho mục đích an toàn:

```
[root@deep /]# chmod 0511 /home/ftp/dev/
[root@deep /]# chmod 0511 /home/ftp/etc/
[root@deep /]# chmod 0511 /home/ftp/bin
[root@deep /]# chmod 0511 /home/ftp/lib
```

Lệnh "**chmod**" sẽ tạo các thư mục "dev", "etc", "bin", và "lib" của chúng ta có quyền đọc và có thể thực thi bởi super-user "root" và quyền có thể thực thi cho user-group và tất cả users.

Bước 3

Sao chép tập tin thư viện "/bin/ls" đến thư mục "/home/ftp/bin" và thay đổi quyền 0111 cho chương trình "ls". (Bạn không muốn các user có thể thay đổi những tập tin thư viện này):

```
[root@deep /]# cp /bin/ls /home/ftp/bin
[root@deep /]# chmod 0111 /bin/ls /home/ftp/bin/ls
```

Lưu ý: *Bước này là cần thiết chỉ khi bạn không dùng tùy chọn "--enable-ls" trong quá trình cấu hình Wu-ftpd. Để biết thêm chi tiết hãy xem phần "Biên dịch và tối ưu trong chương này".*

Bước 4

Hãy tìm các tập tin thư viện dùng chung của chương trình "ls" trên Linux:

```
[root@deep /]# ldd /bin/ls

libc.so.6 => /lib/libc.so.6 (0x00125000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x00110000)
```

Sao chép các tập tin trên vào thư mục mới trong thư mục "/home/ftp" :

```
[root@deep /]# cp /lib/libc.so.6 /home/ftp/lib/
[root@deep /]# cp /lib/ld-linux.so.2 /home/ftp/lib/
```

Lưu ý: *Những tập tin này cần thiết để "ls" có thể thực thi. Bước 3 và 4 ở trên được yêu cầu chỉ khi nào bạn muốn sử dụng chương trình "ls" trên Linux thay vì tùy chọn "--enable-ls" dùng trong Wu-ftpd, đây là một tính năng mới của Wu-ftpd.*

Bước 5

Hãy tạo tập tin "/home/ftp/dev/null":

```
[root@deep /]# mknod /home/ftp/dev/null c 1 3
[root@deep /]# chmod 666 /home/ftp/dev/null
```

Bước 6

Hãy sao chép các tập tin "group" và "passwd" trong thư mục "/home/ftp/etc". Với lí do này, chúng ta sẽ xoá tất cả non FTP users ngoại trừ super-user "root" trong cả hai tập tin (passwd and group).

```
[root@deep /]# cp /etc/passwd /home/ftp/etc/
[root@deep /]# cp /etc/group /home/ftp/etc/
```

Hiệu chỉnh tập tin **passwd** (vi /home/ftp/etc/passwd) và xoá tất cả các thành phần ngoại trừ super-user "root" và những FTP users mà chúng ta đã cho phép trước đó. Đây là điều quan trọng rằng tập tin "passwd" trong môi trường chroot có các thành phần như:

```
root:x:0:0:root:/:/dev/null
ftpadmin:x:502:502::/ftpadmin/:/dev/null
```

Lưu ý: *Ở đây chúng ta cần lưu ý hai điều: đầu tiên, thư mục gốc của các user trong tập tin "passwd" được thay đổi tới thư mục FTP mới (chrooted) nghĩa là thư mục /home/ftp/./ftpadmin/ trở thành thư mục /ftpadmin/, tên shell login của người dùng cho user "root" được đổi thành "/dev/null".*

Hiệu chỉnh tập tin **group** (vi /home/ftp/etc/group) và xoá toàn bộ các thành phần ngoại trừ super-user "root" và các FTP users mà chúng ta đã cấp trước đó. Tập tin "group" nên đổi thành tập tin group bình thường:

```
root:x:0:root
ftpadmin:x:502:
```

Bước 7

Bây giờ chúng ta phải thiết lập bit không thay đổi cho các tập tin "passwd" và "group" trong chroot jail, điều này rất tốt cho việc bảo mật.

- Thiết lập bit không biến đổi trên tập tin "passwd":

```
[root@deep /]# cd /home/ftp/etc/
[root@deep /]# chattr +i passwd
```

- Thiết lập bit không biến đổi trên tập tin "group":

```
[root@deep /]# cd /home/ftp/etc/
[root@deep /]# chattr +i group
```

Cấu hình

Xin nhắc lại là tập tin cấu hình cho phần mềm này bạn có thể lấy từ đĩa mềm kèm theo sách.

- Để chạy một FTP server, sau đây là những tập tin được yêu cầu và phải được tạo hoặc sao chép đến những thư mục thích hợp trên máy chủ của bạn.

  Sao chép tập tin ftpaccess trong thư mục "/etc/"

  Sao chép tập tin ftpusers trong thư mục "/etc/"

Sao chép tập tin ftphosts trong thư mục "/etc/"

Sao chép tập tin ftpgroups trong thư mục "/etc/"

Sao chép tập tin ftpconversion trong thư mục "/etc/"

Sao chép tập tin ftp trong thư mục "/etc/pam.d/"

Sao chép tập tin ftpd trong thư mục "/etc/logrotate.d/"

Cấu hình của tập tin "/etc/ftpaccess"

Tập tin "/etc/ftpaccess" là tập tin cấu hình chính dùng để cấu hình việc thực thi của Wuftpd server. Tập tin này được dùng để điều khiển những user nào, có bao nhiêu user có thể truy cập server của bạn, và những điểm quan trọng khác của vấn đề bảo mật. Mỗi dòng trong tập tin này hoặc là định nghĩa một thuộc tính hoặc là tự thiết lập giá trị cho chính nó.

Bước 1

Hiệu chỉnh tập tin **ftpaccess** (vi /etc/ftpaccess) và thêm mới/thay đổi các dòng trong tập tin này:

```
class openna guest 208.164.186.*

limit openna 20 MoTuWeTh,Fr0000-1800 /home/ftp/.too_many.msg
email admin@openna.com

loginfails 3
readme README* login
readme README* cwd=*

message /home/ftp/.welcome.msg          login
message .message      cwd=*

compress        yes             all
tar             yes             all
chmod           yes             guest
delete          yes             guest
overwrite       yes             guest
rename          yes             guest

log commands  real,guest
log transfers    reat,guest inbound,outbound

guestgroup ftpadmin
guestgroup webmaster

# We don't want users being able to upload into these areas.
upload /home/ftp/* /       no
upload /home/ftp/* /etc    no
upload /home/ftp/* /dev    no

# We'll prevent downloads with noretrieve.
noretrieve /home/ftp/etc
noretrieve /home/ftp/dev
```

```
log security real,guest

guest-root /home/ftp ftpadmin webmaster
restricted-uid ftpadmin webmaster
restricted-gid ftpadmin webmaster

greeting terse
keepalive yes
noretrieve .notar
```

Bước 2

Bây giờ hãy thay đổi quyền mặc định là 600:

```
[root@deep /]# chmod 600 /etc/ftpaccess
```

Điều này bảo tập tin ftpaccess tự thiết lập chính nó cho việc cài đặt cấu hình quen thuộc với:

class openna guest 208.164.186.*

Tùy chọn "class" chỉ rõ một lớp (class) của các user, ai có thể truy xuất máy chủ FTP của bạn. Bạn có thể định nghĩa nhiều lớp khi bạn muốn trong tập tin "ftpaccess". Trong ví dụ chúng ta định nghĩa tên class là <openna> và chúng ta chỉ cho phép user guest <guest> với các tài khoản trên FTP server được truy xuất vào thư mục home của họ thông qua FTP nếu họ kết nối từ những máy có dãy địa chỉ IP là 208.164.186.*. Đây là một lưu ý quan trọng rằng có ba loại user: **anonymous, guest, and real**. Các user **anonymous** là bất cứ ai trên mạng, người đang kết nối tới server và truyền nhập các tập tin mà không cần một tài khoản nào cả. Các user **guest** là các user thật (real users) trên hệ thống được cài đặt như là anonymous FTP (nó là một trong những tài khoản chúng ta cài đặt trong các ví dụ), và **Real** users phải có tài khoản và shells trên máy chủ để có thể truy xuất vào.

limit openna 20 MoTuWeTh,Fr0000-1800 /home/ftp/.too_many.msg

Tùy chọn "limit" chỉ rõ số users được phép đăng nhập vào máy chủ FTP bởi lớp và thời gian trong ngày. Trong ví dụ, chúng ta giới hạn quyền truy xuất tới máy chủ FTP dành cho class tên <openna> là 20 users <20> từ thứ hai đến thứ năm là cả ngày nhưng thứ sáu là nửa đêm đến 6:00 p.m <Fr0000-1800>. Nếu giới hạn của 20 users đã đạt tới thì nội dung tập tin </home/ftp/.too_many.msg> được hiện thị khi có user kết nối. Đây có thể là một tham số hữu dụng khi bạn cần điều khiển các tài nguyên trên máy chủ.

loginfails 3

Tùy chọn "loginfails" chỉ rõ số lần đăng nhập bị thất bại của user trước khi thoát kết nối. Trong ví dụ của chúng ta, chúng ta ngưng kết nối (disconnect) một user từ server sau ba lần kết nối thất bại.

readme README* login

readme README* cwd=

Tùy chọn "readme" dùng để thông báo các client ở thời điểm đăng nhập hoặc khi thay đổi thư mục làm việc, tập tin này chắc chắn được lưu tại thư mục hiện hành. Trong ví dụ chúng ta thiết lập tên của tập tin là <README*> và điều kiện khi thể hiện thông điệp là đăng nhập thành công <login> hoặc khi một client vào thư mục mặc định <cwd=*>.

message /home/ftp/.welcome.msg login

message .message cwd=*

Tùy chọn "message" dùng để trình bày một thông điệp đặc biệt khi user đăng nhập vào hoặc khi user thay đổi thư mục làm việc. Trong ví dụ, vị trí và tên tập tin sẽ thể hiện là </home/ftp/.welcome.msg or .message> và điều kiện để thể hiện là khi user đăng nhập thành công <login>, hoặc khi một client gõ vào thư mục mới <cwd=*>. Với những tùy chọn "readme" và "message" ở trên, nhớ rằng khi bạn định rõ một đường dẫn dành cho anonymous users, thì đường dẫn đó phải có quan hệ với thư mục FTP của anonymous

compress	yes	all
tar	yes	all
chmod	yes	guest
delete	yes	guest
overwrite	yes	guest
rename	yes	guest

Những tùy chọn "compress", "tar", "chmod", "delete", "overwrite", và "rename", định rõ quyền mà bạn muốn gán cho user của bạn, trong ví dụ, chúng ta gán cho nhóm guest <guest> quyền xoá, ghi chồng, đổi tên tập tin và cho phép mọi người sử dụng các lệnh nén tập tin và lệnh tar <all>. Nếu bạn không chỉ rõ những chỉ dẫn thì mặc định sẽ là "yes" dành cho mọi người.

log commands real,guest

Tùy chọn "log commands" chỉ định ghi lại các lệnh riêng lẻ bởi các user dành cho mục đích bảo mật. Trong ví dụ, chúng ta sẽ log tất cả các lệnh riêng lẻ của tất cả các user **real và guest** <real,guest>. Kết quả các log được lưu trữ trong tập tin "/var/log/message" .

log transfers real,guest inbound,outbound

Tùy chọn "log transfers" chỉ định để ghi nhận tất cả việc truyền tải tập tin trên FTP dành cho mục đích bảo mật. Trong ví dụ, chúng ta ghi nhận tất cả quá trình truyền nhận tập tin của tất cả các user **real** và **guest** <real,guest> và ghi nhận cả hai **inbound** và **outbound** <inbound,outbound> chỉ định việc truyền nhận phải trong thứ tự để log. Thông tin về kết quả được lưu trữ trong tập tin "/var/log/xferlog".

guestgroup ftpadmin

guestgroup webmaster

Tùy chọn "guestgroup" chỉ định tất cả các nhóm guest của bạn là những real users trên hệ thống được cài đặt như là anonymous FTP <ftpadmin and webmaster>. Tập tin "/home/ftp/etc/group" có các thành phần cho mỗi nhóm và mỗi thành viên. Đây là điều rất quan trọng rằng, guestgroup xuất hiện một lần trên mỗi dòng trong tập tin cấu hình.

log security real,guest

Tùy chọn "log security" định rõ cho phép ghi lại sự vi phạm những qui tắc bảo mật của các client là user thật (real), guest (khách) và/hoặc anonymous FTP (user bất kỳ). Trong ví dụ của chúng ta, chúng ta chỉ định ghi lại sự vi phạm cho những user là real và guest <real,guest>.

```
guest-root /home/ftp ftpadmin webmaster
restricted-uid ftpadmin webmaster
restricted-gid ftpadmin webmaster
```

Những điều này, "guest-root", "restricted-uid", "restricted-gid" định rõ và điều khiển có hay không những người dùng guest sẽ được quyền truy cập những nơi trên máy chủ FTP mà nằm ngoài những thư mục home của họ (đây là một đặc trưng quan trọng về an toàn). Trong ví dụ của chúng ta, chúng ta chỉ rõ đường dẫn chroot() cho những người dùng <ftpadmin và webmater> ở thư mục </home/ftp>, điều đó làm cho họ không thể truy cập những tập tin lẫn nhau vì chúng bị giới hạn bởi những thư mục home của họ <retricted-uid ftpadmin webmaster>, <retricted-gid ftpadmin webmaster>. Nếu một guestroot được chọn cho user, thư mục home của user trong tập tin "<root-dir>/etc/passwd" được dùng để xác định thư mục ban đầu và thư mục home của chúng, trong system-wide "/etc/passwd", không sử dụng. Đây là một đặc trưng an toàn.

greeting terse

"greeting" định rõ những thông tin hệ thống nào sẽ được hiển thị trước khi user đăng nhập từ xa. Có ba tham số bạn có thể chọn: <full> đây là giá trị mặc định và hiển thị hostname và phiên bản daemon của máy chủ, <brief> nó chỉ hiển thị hostname, và <terse> đơn giản chỉ hiển thị dòng "FTP server ready" trên terminal của bạn.

keepalive yes

"keepalive" chỉ rõ có hay không hệ thống sẽ gửi các message keepalive tới FTP server ở xa, nếu chọn giá trị "yes" thì sự chấm dứt của connection hoặc sự cố trên máy từ xa sẽ được thông báo một cách chính xác.

Cấu hình của tập tin "/etc/ftphosts"

Tập tin "/etc/ftphosts" được dùng để định nghĩa có hay không những user đã được phép đăng nhập từ host nào đó hoặc nó bị từ chối truy cập.

Bước 1

Tạo tập tin **ftphosts** (touch /etc/ftphosts) và thêm vào trong tập tin ví dụ này những dòng sau:

```
# Example host access file
#
# Everything after a '#' is treated as comment,
# empty lines are ignored
allow ftpadmin 208.164.186.1 208.164.186.2 208.164.186.4
deny ftpadmin 208.164.186.5
```

Trong ví dụ trên, chúng ta cho phép user <ftpadmin> kết nối qua FTP từ danh sách địa chỉ rõ ràng <208.164.186.1 208.164.186.2 08.164.186.4>, và không cho phép user <ftpadmin> kết nối từ địa chỉ <208.164.186.5>.

Bước 2

Bây giờ, thay đổi quyền mặc định là 600 :

Cấu hình của tập tin "/etc/ftpusers"

Tập tin "/etc/ftpusers" định rõ không cho phép những user này kết nối đến máy chủ FTP.

Bước 1

Tạo tập tin **ftpusers** (touch /etc/ftpusers) và thêm vào trong tập tin này những user sau vì những lý do an toàn :

```
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
news
uucp
operator
games
nobody
```

Bước 2

Bây giờ, thay đổi quyền mặc định là 600 :

```
[root@deep /]# chmod 600 /etc/ftpusers
```

Cấu hình của tập tin "/etc/ftpconversions"

Đây là tập tin "/etc/ftpconversions" bao gồm những hướng dẫn cho phép bạn nén những tập tin theo yêu cầu trước khi truyền.

Bước 1

Thay đổi tập tin **ftpconversions** (vi /etc/ftpconversions) và thêm vào trong tập tin này những dòng sau :

```
:.Z: : :/bin/compress -d -c %s:T_REG|T_ASCII:O_UNCOMPRESS:UNCOMPRESS
: : :.Z:/bin/compress -c %s:T_REG:O_COMPRESS:COMPRESS
:.gz: : :/bin/gzip -cd %s:T_REG|T_ASCII:O_UNCOMPRESS:GUNZIP
: : :.gz:/bin/gzip -9 -c %s:T_REG:O_COMPRESS:GZIP
: : :.tar:/bin/tar -c -f - %s:T_REG|T_DIR:O_TAR:TAR
: : :.tar.Z:/bin/tar -c -Z -f - %s:T_REG|T_DIR:O_COMPRESS|O_TAR:TAR+COMPRESS
: : :.tar.gz:/bin/tar -c -z -f - %s:T_REG|T_DIR:O_COMPRESS|O_TAR:TAR+GZIP
: : :.crc:/bin/cksum %s:T_REG::CKSUM
: : :.md5:/bin/md5sum %s:T_REG::MD5SUM
```

Bước 2

Bây giờ, thay đổi quyền mặc định là 600 :

```
[root@deep /]# chmod 600 /etc/ftpconversions
```

Cấu hình của tập tin "/etc/pam.d/ftp"

Tập tin cấu hình "/etc/pam.d/ftp" sử dụng kỹ thuật xác thực pam (pam authentication).

Tạo tập tin **ftp** (touch /etc/pam.d/ftp) và thêm vào những dòng sau :

```
#%PAM-1.0
auth required /lib/security/pam_listfile.so item=user sense=deny file=/etc/ftpusers
auth required /lib/security/pam_pwdb.so shadow nullok
auth required /lib/security/pam_shells.so
account required /lib/security/pam_pwdb.so
session required /lib/security/pam_pwdb.so
```

Cấu hình của tập tin "/etc/logrotate.d/ftpd"

Cấu hình tập tin "/etc/logrotate.d/ftpd" để tự động quay vòng những tập tin log mỗi tuần.

Tạo tập tin **ftpd** (touch etc/logrotate.d/ftpd) và thêm vào những dòng sau:

```
/var/log/xferlog {
# ftpd doesn't handle SIGHUP properly
nocompress
}
```

Cấu hình ftpd để dùng tcp-wrappers inetd super server

Tcp-wrappers sẽ cho phép khởi động và dừng máy chủ ftpd. Dựa trên sự thực hiện này, inetd đọc thông tin cấu hình của nó từ một tập tin cấu hình này, mặc nhiên là tập tin "/etc/inetd.conf". Phải có một mục nhập cho mỗi trường (field)

của tập tin cấu hình, những mục cho mỗi field sẽ được phân biệt bởi một tab hay khoảng trắng.

Bước 1

Thay đổi tập tin **inetd.conf** (vi /etc/inetd.conf) và thêm vào hoặc kiểm tra lại đã có dòng sau :

```
ftp stream tcp nowait root /usr/sbin/tcpd in.ftpd -l –a
```

Chú ý: *Cập nhật tập tin "inetd.conf" bằng cách một lệnh SIGHUP (killall –HUP inetd) sau khi thêm trong tập tin này dòng trên.*

- Để cập nhật tập tin "inetd.conf", dùng lệnh sau :

```
[root@deep /]# killall -HUP inetd
```

Bước 2

Thay đổi tập tin **hosts.allow** (vi /etc/hosts.allow) và thêm vào dòng sau:

```
in.ftpd: 192.168.1.4 win.openna.com
```

Cho phép client mang địa chỉ IP "192.168.1.4" với host name "win.openna.com" được phép FTP trên máy chủ.

Các công cụ quản trị FTP (FTP Administrative Tools)

ftpwho

Chức năng chương trình ftpwho hiển thị tất cả những users ftp đang hoạt động, và thông tin tiến trình hiện tại của chúng trên hệ thống. Kết quả của lệnh này trong định dạng của lệnh "/bin/ps". Định dạng của lệnh này là :

- Để hiển thị tất cả những ftp user đang hoạt động và tiến trình hiện tại của chúng, dùng lệnh sau :

```
[root@deep /]# ftpwho
Service class openna:
5443 ?          S       0:00  ftpd: win.openna.com: ftpadmin: IDLE
        1 users ( 20 maximum)
```

Ở đây, bạn có thể thấy một user đã truy cập vào, với tổng số 20 users được phép truy cập, và user này có tên là "ftpadmin" những yêu cầu của user này từ win.openna.com.

ftpcount

Chức năng chương trình ftpcount là một phiên bản đơn giản của ftpwho, chỉ hiển thị số users hiện tại truy cập đến hệ thống, và số lượng user được phép truy cập.

- Chỉ hiển thị số users hiện tại truy cập đến hệ thống, và số lượng user được phép truy cập, dùng lệnh sau :

```
[root@deep /]# ftpcount
```

Service class openna - 1 users (20 maximum)

Bảo mật FTP

Tập tin ftpusers

Điều quan trọng để chắc rằng bạn cài đặt tập tin "/etc/ftpusers" này xác định rõ những user không được phép kết nối đến máy chủ FTP. Đây sẽ được xem như một yêu cầu tối thiểu (MINIMUM request), những mục nhập sau: root, bin, daemon, adm, lp, sync, shutdown, halt, mail, news, uucp, operator, games, nobody và tất cả linh tinh mặc nhiên khác cung cấp những tài khoản sẵn có trong tập tin "/etc/passwd".

Chương trình anonymous FTP

Không cho phép anonymous FTP, Xóa user anonymous "**ftp**" từ tập tin mật khẩu và chắc rằng gói tin **anon-version.i386.rpm** không cài đặt trên hệ thống của bạn.

- Để xóa user "ftp" từ tập tin mật khẩu, dùng lệnh sau :

  [root@deep /]# **userdel ftp**

- Để chắc rằng gói tin RPM của chương trình anonymous FTP không cài đặt trên hệ thống Linux, dùng lệnh sau:

  [root@deep /]# rpm -q anonftp
  package anonftp is not installed

Lệnh upload

Mặc nhiên, máy chủ Wu-ftpd sẽ cấp cho tất cả user quyền upload. Tham số upload cho phép những client từ xa tải và đặt những tập tin trên máy chủ FTP. An toàn nhất là chúng ta không muốn những user upload vào trong "bin", "etc", "dev", và "lib" là những thư mục con trong thư mục "/home/ftp", trong tập tin "/etc/ftpaccess" chúng ta có những chroot'd user đến "/home/ftp" và họ không thể truy cập bất kỳ nơi nào của những tập tin hệ thống ngoài cấu trúc thư mục đó. Nhưng trong vài trường hợp xảy ra, bạn sẽ không cho phép những quyền upload trong tập tin "/etc/ftpaccess" vào trong những vùng như (/home/ftp, /home/ftp/bin, /home/ftp/dev, và /home/ftp/lib).

Thay đổi tập tin **ftpaccess** (vi /etc/ftpaccess) và thêm vào những dòng sau để không cho phép quyền upload đến những vùng này.

```
# We don't want users being able to upload into these areas.
upload /home/ftp/* / no
upload /home/ftp/* /etc no
upload /home/ftp/* /dev no
upload /home/ftp/* /bin no (require only if you are not using the "--enable-ls" option)
upload /home/ftp/* /lib no (require only if you are not using the "--enable-ls" option)
```

Những dòng trên định rõ không cho phép upload vào trong những thư mục "/", "/etc", "/dev", "/bin", và "/lib" của cấu trúc thư mục chroot'd "/home/ftp".

Tập tin đặc biệt ".notar"

Mục đích của việc này là không cho phép dùng lệnh tar trong tất cả các vùng mà tham số upload không cho phép, cho dù bạn cho phép dùng lệnh tar trên các thư mục hoặc tập tin.

Bước 1

Để làm như vậy, tạo tập tin đặc biệt '.notar' trong mỗi thư mục và trong thư mục FTP.

```
[root@deep /]# touch /home/ftp/.notar
[root@deep /]# touch /home/ftp/etc/.notar
[root@deep /]# touch /home/ftp/dev/.notar
[root@deep /]# touch /home/ftp/bin/.notar (chỉ yêu cầu nếu bạn không dùng tùy chọn "--enable-ls)
[root@deep /]# touch /home/ftp/lib/.notar (chỉ yêu cầu nếu bạn không dùng tùy chọn "--enable-ls)
[root@deep /]# chmod 0 /home/ftp/.notar
[root@deep /]# chmod 0 /home/ftp/etc/.notar
[root@deep /]# chmod 0 /home/ftp/dev/.notar
[root@deep /]# chmod 0 /home/ftp/bin/.notar (chỉ yêu cầu nếu bạn không dùng tùy chọn "--enable-ls)
[root@deep /]# chmod 0 /home/ftp/lib/.notar (chỉ yêu cầu nếu bạn không dùng tùy chọn "--enable-ls)
```

Bước 2

Tập tin ".notar" không có gì cả, có thể làm nhầm lẫn một vài web client và những FTP proxy, do đó lưu ý đến nó và tránh tổn thất khi giải quyết vấn đề đó. Thêm những dòng sau trong tập tin "/etc/ftpaccess".

Hiệu chỉnh tập tin **ftpaccess** (vi /etc/ftpaccess) và thêm vào những dòng sau để đánh dấu những tập tin ".notar".

```
noretrieve .notar
```

Lệnh noretrieve

Tham số noretrieve của Wu-ftpd server cho phép bạn ngăn không cho truyền những thư mục hay tập tin đã chọn. Đây là một ý hay để ngăn không cho tải những thư mục con (bin, etc, dev, và lib) này xuống trong thư mục "/home/ftp" với lệnh "noretrieve" trong tập tin "/etc/ftpaccess".

Hiệu chỉnh tập tin **ftpaceess** (vi /etc/ftpaccess) và thêm vào những dòng sau để không cho truyền nhận đến những vùng này.

```
# We'll prevent downloads with noretrieve.
noretrieve /home/ftp/etc
noretrieve /home/ftp/dev
noretrieve /home/ftp/bin (chỉ yêu cầu nếu bạn không dùng tùy chọn "--enable-ls)
```

noretrieve /home/ftp/lib (chỉ yêu cầu nếu bạn không dùng tùy chọn "--enable-ls)

Những tập tin được cài đặt

> /etc/pam.d/ftp
> /etc/logrotate.d/ftpd
> /etc/ftpaccess
> /etc/ftpconversions
> /etc/ftpgroups
> /etc/ftphosts
> /etc/ftpusers
> /home/ftp/
> /usr/man/man5/ftpconversions.5
> /usr/man/man5/xferlog.5
> /usr/man/man8/ftpd.8
> /usr/man/man8/ftpshut.8
> /usr/man/man8/ftprestart.8
> /usr/sbin/in.ftpd
> /usr/sbin/ftpshut
> /usr/sbin/ckconfig
> /usr/bin/ftpcount
> /usr/bin/ftpwho
> /usr/man/man1/ftpcount.1
> /usr/man/man1/ftpwho.1
> /usr/man/man5/ftpaccess.5
> /usr/man/man5/ftphosts.5
> /usr/sbin/ftprestart
> /usr/sbin/xferstats
> /usr/sbin/wu.ftpd
> /usr/sbin/in.wuftpd
> /var/log/xferlog

PHẦN 5

CÁC VẤN ĐỀ LIÊN QUAN VỀ SAO LƯU DỮ LIỆU

Chương 22:

CÁC THỦ TỤC VỀ SAO LƯU VÀ KHÔI PHỤC DỮ LIỆU

Trong chương này chúng ta sẽ đề cập đến các vấn đề sau:

- Chương trình sao chép dự phòng tar.
- Tạo các sao chép dự phòng với tar.
- Các công việc sao lưu tự động được tạo với tar.
- Phục hồi các tập tin với tar.
- Chương trình sao chép dự phòng dump.
- Tạo các sao lưu với dump.
- Phục hồi lại các tập tin với dump.
- Sao chép dự phòng và phục hồi thông qua mạng.



Sao lưu và khôi phục dữ liệu

Giới thiệu tổng quan

Một server bảo mật và đáng tin cậy thì có mối quan hệ chặt chẽ với việc thường xuyên thực hiện sao chép dự phòng (backup). Đôi khi việc sao chép dự phòng cũng xảy ra thất bại. Nguyên nhân gây ra việc này có thể là do sự tấn công của các tin tặc, phần cứng bị hỏng, lỗi do người gây ra, mất điện v.v..Phương pháp an toàn nhất để thực hiện sao chép dự phòng là ghi chúng ở một nơi tách biệt với hệ thống Linux của bạn như thông qua một mạng máy tính, với băng từ, ổ đĩa có thể xóa, CD-ROM có khả năng ghi, v.v..

Đã có rất nhiều phương pháp thực hiện việc sao chép dự phòng với Linux, chẳng hạn như "dump", "tar", "cpio", và "dd" mà mỗi phương pháp đã có sẵn mặc nhiên trên hệ thống Linux của bạn. Cũng đã có sẵn các tiện ích dựa trên văn bản, ví dụ như "Amanda", nó được thiết kế để thêm vào một giao diện người dùng thân thiện cho các thủ tục sao chép dự phòng và phục hồi (restore procedures). Cuối cùng, các trình tiện ích sao chép dự phòng thương mại thì cũng đã có, chẳng hạn như "BRU".

Các thủ tục cho việc thực hiện một sao chép dự phòng và phục hồi sẽ không giống nhau phụ thuộc vào chọn lựa một giải pháp sao chép dự phòng của bạn. Vì lý do này chúng ta sẽ thảo luận các phương pháp cho việc thực hiện sao chép dự phòng với các công cụ UNIX cổ điển: "tar" và "dump" mà nó là một công cụ sao chép dự phòng bằng dòng lệnh.

Những gì cần sao chép dự phòng:

Có ý kiến cho rằng tạo một sao chép dự phòng là để phòng hờ rất nhiều trường hợp có thể xảy ra trên hệ thống của bạn, nhưng có một số ngoại lệ như được trình bày bên dưới. Điều đó cũng không hợp lý để đưa vào những ngoại lệ này trong việc sao chép dự phòng của bạn bởi vì bạn sẽ mất thời gian và không gian để thực hiện mà không được gì cả.

Một số ngoại lệ chính yếu không cần đưa vào trong việc sao chép dự phòng là:

- Hệ thống tập tin "/proc": Bởi vì nó chỉ chứa đựng dữ liệu mà nhân của hệ điều hành (kernel) sinh ra một cách tự động, Việc này cũng chưa bao giờ là một ý kiến hay để sao chép dự phòng nó.
- File hệ thống "/mnt", bởi vì nó là nơi bạn gắn vào hệ thống các phương tiện thông tin như CD-ROM, đĩa mềm và những thiết bị khác lưu trữ dữ liệu khác.
- Thư mục sao chép dự phòng hay các dạng media nơi bạn đã đặt các tập tin bakup, ví dụ như một băng từ (tape), CD-ROM, NFS được đặt vào tập tin hệ thống, thư mục nội bộ/từ xa hoặc các dạng khác của media.
- Phần mềm mà có thể dễ dàng cài đặt lại, thông qua đó người ta có thể có các tập tin cấu hình mà quan trọng để sao chép dự phòng, e rằng bạn phải làm

tất cả các công việc để cấu hình tất cả chúng một lần nữa. Tôi khuyên bạn là nên đặt chúng (các tập tin cấu hình cho phần mềm) trên đĩa mềm.

Chương trình sao chép dự phòng tar

Chương trình sao chép dự phòng tar là một chương trình lưu trữ được thiết kế để lưu trữ và trích ra một số tập tin từ một tập tin lưu trữ được biết như là một tarfile. Một tarfile có thể được tạo trên một băng từ (tape drive); tuy nhiên, nó cũng dùng chung cho cả việc ghi tarfile thành một tập tin bình thường.

Một sự sắp xếp sao chép dự phòng đơn giản

Khi bạn quyết định tạo một sao chép dự phòng một số tập tin trên hệ thống của bạn, bạn phải chọn một kế hoạch sao lưu trước khi bắt đầu các thủ tục sao lưu. Có rất nhiều các chiến lược sao lưu và phụ thuộc vào các chính sách sao lưu mà bạn muốn dùng. Sau đây, tôi sẽ trình bày một kế hoạch sao lưu, bạn có thể dùng để tận dụng được các khả năng có thể có của chương trình tar. Kế hoạch này là: đầu tiên sao chép dự phòng tất cả mọi thứ ở lần đầu, sau đó sao chép dự phòng mọi thứ đã được hiệu chỉnh từ sao lưu trước. Việc sao lưu đầu tiên được gọi là sao lưu đầy đủ; các sao lưu tiếp theo sau là các sao lưu gia tăng.

Tạo các sao chép dự phòng với tar

Với sáu đĩa băng từ (tape) bạn có thể tạo các sao chép dự phòng mỗi ngày; Thủ tục là dùng tape 1 trong lần sao lưu đầy đủ đầu tiên (Thứ 6 - ứng với tape thứ 1), và tape thứ 2 tới 5 cho các sao lưu gia tăng (Thứ 2 tới Thứ 5). Sau đó, bạn tạo một sao lưu đầy đủ trên tape 6 (Thứ 6 lần 2) và bắt đầu thực hiện các sao lưu gia tăng với tape 2 tới 5 lần nữa. Một điều quan trọng để giữ tape1 ở trạng thái của nó cho tới khi bạn có một sao lưu đầy đủ với tape 6. Trong ví dụ bên dưới, giả sử rằng chúng ta ghi sao lưu tới một ổ tape SCSI có tên là (/dev/st0), và chúng ta sao lưu thư mục home (/home) của hệ thống của chúng ta.

Trước tiên, chúng ta phải chuyển tới partition tập tin hệ thống "/". Khi tạo một tập tin lưu trữ, "tar" sẽ tháo gỡ các ký tự đứng đầu "/" (slash) từ các tên đường dẫn tập tin. Điều này nghĩa là những tập tin được phục hồi có thể không phải kết thúc ở cùng các vị trí mà chúng được sao lưu từ trước. Chính vì vậy, để giải quyết vấn đề, giải pháp là thay đổi sang thư mục gốc "/" trước khi tạo tất cả các sao lưu và các sự phục hồi.

- Để chuyển tới thư mục gốc "/", dùng lệnh :

```
[root@deep]# cd /
```

Điều quan trọng là luôn bắt đầu với một sao lưu đầy đủ (full backup) (như đã nói, là vào thứ 6), ví dụ:

- Friday 1, (Sử dụng tape thứ 1 cho lần backup đầy đủ lần đầu tiên).

```
[root@deep /]# cd /
[root@deep /]# tar cpf /dev/st0 --label=" full-backup created on `date '+%d-
%B-%Y'`." \
```

```
--directory /home
```

- Monday, (Sử dụng tape thứ 2 cho lần backup gia tăng).

```
[root@deep /]# cd /
[root@deep /]# tar cpNf /dev/st0 --label=" full-backup created on \
`date '+%d-%B-%Y'`." --directory /home
```

- Tuesday, (Sử dụng tape thứ 3 cho lần backup gia tăng).

```
[root@deep /]# cd /
[root@deep /]# tar cpNf /dev/st0 --label=" full-backup created on \
`date '+%d-%B-%Y'`." --directory /home
```

- Wednesday, (Sử dụng tape thứ 4 cho lần backup gia tăng).

```
[root@deep /]# cd /
[root@deep /]# tar cpNf /dev/st0 --label=" full-backup created on \
`date '+%d-%B-%Y'`." --directory /home
```

- Thursday, (Sử dụng tape thứ 5 cho lần backup gia tăng).

```
[root@deep /]# cd /
[root@deep /]# tar cpNf /dev/st0 --label=" full-backup created on \
 date '+%d-%B-%Y'`." --directory /home
```

- Friday 2, (Sử dụng tape thứ 6 cho lần backup gia tăng).

```
[root@deep /]# cd /
[root@deep /]# tar cpf /dev/st0 --label=" full-backup created on \
`date '+%d-%B-%Y'`." --directory /home
```

Tiếp tục thực hiện việc sao lưu gia tăng cho các tape từ thứ hai đến thứ 5 trở lại.

Tùy chọn "c" chỉ rõ rằng một tập tin lưu trữ bắt đầu được tạo.

Tùy chọn "p" thì giữ nguyên các quyền; thông tin bảo vệ tập tin sẽ "được ghi nhớ".

Tùy chọn "N" thực hiện một sao lưu gia tăng và chỉ lưu trữ các tập tin mới hơn DATE (ngày backup so với lần trước).

Tùy chọn "f" trạng thái mà mọi thông số tiếp theo sẽ có tên của tập tin lưu trữ hay thiết bị được ghi chép.

Chú ý: *Làm thế nào một filename, cái chứa đựng một ngày hiện hành, được tìm thấy, đơn giản bằng cách đính kèm theo lệnh "date" giữa hai ký tự back-quote. Một nguyên tắc đặt tên chung là thêm vào một hậu tố "tar" cho các dạng lưu trữ chưa được nén, và một hậu tố "tar.gz"cho các dạng tập tin lưu trữ được nén. Bởi vì chúng ta không thể chỉ rõ một tên file cho tập backup, tùy chọn "-label" có thể được dùng để ghi một số thông tin về tập sao lưu sang dạng lưu trữ chính nó. Sau cùng, chỉ các tập tin chứa đựng trong "/home" là được ghi vào tape.*

Bởi vì các ổ đĩa tape là một thiết bị ký tự, nó không thể được chỉ rõ một tên tập tin thật sự. Chính vì vậy, tên tập tin được dùng như là một thông số mà có thể ghi nhiều tập trên một tape. Bạn cũng có thể kể đến thiết bị như "/dev/st0" là thiết bị tape đầu tiên. Thiết bị "/dev/st0" không quay lại từ đầu sau khi tập sao lưu được ghi; Chính vì vậy, có thể ghi nhiều tập trên một tape. Bạn cũng có thể tham chiếu tới thiết bị như "/dev/st0", mà trong trường hợp tape được quay trở lại từ đầu một cách tự động sau khi bộ sao lưu được ghi. Khi làm việc với các tape, bạn có thể dùng các lệnh sau để quay lại từ đầu và lấy tape của bạn ra:

```
[root@deep /]# mt -f /dev/st0 rewind
[root@deep /]# mt -f /dev/st0 offline
```

Nguyên nhân: *Để giảm bớt khoảng không cần thiết trên một dạng lưu trữ tar, các sao lưu có thể được nén với tùy chọn "z" của chương trình tar. Thật không may, sử dụng tùy chọn này để nén các sao lưu có thể gây ra phiền toái. Bởi vì tính tự nhiên của các công việc nén như thế nào, nếu một bit riêng rẽ trong sao lưu được nén là hỏng, tất cả các dữ liệu được nén sẽ bị mất hết. Một điều được khuyên là KHÔNG dùng dạng nén (Tùy chọn "z") để tạo các sao lưu với lệnh tar.*

- Nếu việc sao lưu của bạn không vừa trên một tape, bạn cần phải dùng tùy chọn –multi-volume (-M):

```
[root@deep /]# cd /
[root@deep /]# tar cMpf /dev/st0 /home
Prepare volume #2 for /dev/st0 and hit return:
```

- Sau khi tạo một backup, bạn nên kiểm tra xem nó có đúng không, dùng tùy chọn --compare (-d) như được trình bày bên dưới:

```
[root@deep /]# cd /
[root@deep /]# tar dvf /dev/st0
```

- Để thực hiện một sao lưu toàn bộ hệ thống của bạn, dùng lệnh sau:

```
[root@deep /]# cd /
[root@deep /]# tar cpf /archive/full-backup-`date '+%d-%B-%Y'`.tar \
--directory / --exclude=proc --exclude=mnt --exclude=archive \
--exclude=cache --exclude=*/lost+found .
```

Tùy chọn "--directory" bảo tar đầu tiên chuyển sang đường dẫn thư mục sau (thư mục "/" trong ví dụ này) ưu tiên để bắt đầu sao lưu. Các tùy chọn "--exclude" bảo tar không sao chép dự phòng cả hai bao gồm các tập tin và thư mục được chỉ định. Sau cùng, ký tự "." ở cuối dòng lệnh bảo tar rằng nó sẽ sao chép dự phòng mọi thứ trong thư mục hiện hành.

Nguyên nhân: *Khi sao chép dự phòng các tập tin hệ thống của bạn, không bao gồm "/proc' pseudo-file-system! Các tập tin trong "/proc" không phải các tập tin thật sự nhưng là các tập tin liên kết mô tả và trỏ*

tới các cấu trúc dữ liệu kernel. Cũng không bao gồm các thư mục "/mnt", "/archive", và tất cả "lost+found".

Các công việc sao lưu tự động được tạo với tar

Điều này luôn thú vị là các công việc tự động của một sao chép dự phòng. Việc tự động đưa ra một số các cơ hội to lớn cho việc dùng Linux server để lưu trữ các mục đích mà bạn đặt. Ví dụ ở dưới là script sao lưu của chúng ta, có tên là "backup.cron". Script này được thiết kế để chạy trên bất kỳ máy tính nào bằng cách chỉ thay đổi bốn biến: COMPUTER, DIRECTORY, BACKUPDIR, và TIMEDIR. Tôi đề nghị rằng bạn đặt script này và chạy nó ở đầu tháng trong lần đầu tiên, và sau đó chạy nó trong một tháng trước khi tạo các thay đổi chủ yếu. Trong ví dụ bên dưới, chúng ta sao lưu một thư mục trên server cục bộ (BACKUPDIR), nhưng bạn có thể hiệu chỉnh script này để thực hiện nó thành một tape trên server cục bộ hay thông qua một NFS được mount tập tin hệ thống.

Bước 1

Tạo một tập tin script sao lưu backup.cron (touch /etc/cron.daily/backup.cron) và thêm các dòng sau vào tập tin backup:

```
#!/bin/sh
# full and incremental sao lõu script
# created 07 February 2000
# Based on a script by Daniel O'Callaghan <danny@freebsd.org>
    # and modified by Gerhard Mourani <gmourani@videotron.ca>

#Change the 5 variables below to fit your computer/backup

COMPUTER=deep                   # name of this computer
DIRECTORIES="/home"             # directoris to backup
BACKUPDIR=/backups              # where to store the backups
TIMEDIR=/backups/last-full      # where to store time of full backup
TAR=/bin/tar                    # name and locaction of tar

#You should not have to change anything below here

PATH=/usr/local/bin:/usr/bin:/bin
DOW=`date +%a`                  # Day of the week e.g. Mon
DOM=`date +%d`                  # Date of the Month e.g. 27
    DM=`date +%d%b              # Date and Month e.g. 27Sep

# On the 1st of the month a permanet full sao lõu is made
# Every Sunday a full sao lõu is made - overwriting last Sundays backup
# The rest of the time an incremental sao lõu is made. Each incremental
```

```
# sao lõu overwrites last weeks incremental sao lõu of the same name.
#
# if NEWER = "", then tar backs up all files in the directories
# otherwise it backs up files newer than the NEWER date. NEWER
    # gets it date from the taäp tin written every Sunday.

# Monthly full backup
if [ $DOM = "01" ]; then
      NEWER=""
      $TAR $NEWER -cf $BACKUPDIR/$COMPUTER-$DM.tar $DIRECTORIES
    fi

# Weekly full backup
if [ $DOW = "Sun" ]; then
      NEWER=""
      NOW=`date +%d-%b`
      # Update full sao lõu date
        echo $NOW > $TIMEDIR/$COMPUTER-full-date
        $TAR $NEWER -cf $BACKUPDIR/$COMPUTER-$DOW.tar $DIRECTORIES

# Make incremental sao lõu - overwrite last weeks
else

      # Get date of last full backup
          NEWER="--newer `cat $TIMEDIR/$COMPUTER-full-date`"
      $TAR $NEWER -cf $BACKUPDIR/$COMPUTER-$DOW.tar $DIRECTORIES
    fi
```

Đây là một tóm tắt của thư mục sao lưu sau một tuần:

```
[root@deep /]# ls -l /backups/
total 22217
-rw-r--r-- 1 root root      10731288  Feb 7 11:24 deep-01Feb.tar
-rw-r--r-- 1 root root          6879  Feb 7 11:24 deep-Fri.tar
-rw-r--r-- 1 root root          2831  Feb 7 11:24 deep-Mon.tar
-rw-r--r-- 1 root root          7924  Feb 7 11:25 deep-Sat.tar
-rw-r--r-- 1 root root      11923013  Feb 7 11:24 deep-Sun.tar
-rw-r--r-- 1 root root          5643  Feb 7 11:25 deep-Thu.tar
-rw-r--r-- 1 root root          3152  Feb 7 11:25 deep-Tue.tar
-rw-r--r-- 1 root root          4567  Feb 7 11:25 deep-Wed.tar
drwxr-xr-x 2 root root          1024  Feb 7 11:20 last-full
```

Ghi chú: *Thư mục lưu trữ các sao lưu (BACKUPDIR) và thư mục nơi lưu trữ thời gian của sao lưu đầy đủ (TIMEDIR) phải tồn tại hay phải được tạo trước khi dùng backup-script, bằng không bạn sẽ nhận một thông điệp lỗi.*

Bước 2

Nếu bạn không chạy script sao lưu này từ đầu tháng (01-month-year), các sao lưu gia tăng sẽ cần thời gian của sao lưu Sunday để có thể làm việc chính xác. Nếu bạn bắt đầu ở giữa tuần, bạn sẽ cần phải tạo tập tin thời gian trong TIMEGDIR.

- Để tạo một tập tin thời gian trong thư mục TIMEDIR, dùng lệnh sau:

```
[root@deep /]# date +%d%b > /backups/last-full/myserver-full-date
```

Trong đó </backups/last-full> là biến TIMEDIR, nơi mà chúng ta muốn lưu trữ thời gian của full backup và <myserver-full-date> là tên của server của chúng ta (ví dụ như deep) và tập tin thời gian của chúng ta gồm có một dòng riêng với ngày hiện tại (ví dụ là 15-Feb).

Bước 3

Tạo tập tin script có thể thi hành và thay đổi các quyền mặc định của nó để chỉ có thể ghi bởi super-user "root" (755).

```
[root@deep /]# chmod 755 /etc/cron.daily/backup.cron
```

Ghi chú: *Bởi vì script này là trong thư mục "/etc/cron.daily", nó sẽ chạy tự động như một cron job lúc đúng một giờ sáng mỗi ngày.*

Phục hồi các tập tin với tar

Quan trọng hơn việc thực hiện sao lưu thường xuyên là chúng đã có sẵn khi chúng ta cần phải lấy lại các tập tin quan trọng! Trong phần này, sẽ thảo luận về các phương pháp phục hồi (restore) tập tin đã được sao chép dự phòng với lệnh "tar".

Lệnh sau sẽ phục hồi tất cả các tập tin từ dạng lưu trữ "full-backup-Day-Month-Year.tar", nó là một ví dụ sao lưu của thư mục "home" được tạo từ ví dụ các lệnh tar được trình bày ở trên.

- Để phục hồi một full backup của thư mục "home", dùng lệnh sau:

```
[root@deep /]# cd /
[root@deep /]# tar xpf /dev/st0/full-backup-Day-Month-Year.tar
```

Lệnh trên trích ra tất cả các tập tin được chứa đựng trong dạng lưu trữ nén, duy trì các quyền và quyền sở hữu các tập tin gốc.

Tùy chọn "x" thay cho extract (trích ra các tập tin từ dạng lưu trữ nén).

Tùy chọn "p" duy trì các quyền hạn sử dụng; Thông tin bảo vệ tập tin sẽ được ghi nhớ.

Tùy chọn "f" trạng thái mà mọi thông số kế tiếp sẽ là tên của tập tin lưu trữ hay thiết bị.

Nếu bạn không cần phục hồi tất cả các tập tin được chứa đựng trong tập tin lưu trữ, bạn có thể chỉ ra một hay nhiều tập tin mà bạn muốn phục hồi:

- Để chỉ rõ một hay nhiều tập tin bạn muốn phục hồi, dùng lệnh sau:

```
[root@deep]# cd /
[root@deep]# tar xpf /dev/st0/full-backup-Day-Month-Year.tar \
home/wahib/Personal/Contents.doc home/quota.user
```

Lệnh trên phục hồi các tập tin "/home/wahib/Personal/Contents.doc" và "/home/quota.user" từ dạng lưu trữ.

- Nếu bạn chỉ muốn xem những tập tin nào là thuộc về bản backup, Dùng tùy chọn –list (-t):

```
[root@deep /]# tar tf /dev/st0
```

Nguyên nhân: *Nếu bạn có các tập tin trên hệ thống của bạn đã được set với bit không thay đổi, thì dùng lệnh "chattr", các tập tin này sẽ không được ghi nhớ với bit không thay đổi từ sao lưu của bạn được phục hồi. Bạn phải khởi tạo lại nó với lệnh "chattr + i" sau khi sao lưu được hoàn thành.*

Kiểm tra khả năng phục hồi từ các sao chép dự phòng.

Đối với nhiều người quản trị hệ thống, việc phục hồi một tập tin từ một sao lưu là một việc làm không thường xuyên. Bước này bảo đảm rằng nếu bạn cần phải phục hồi lại một tập tin, các công cụ và các tiến trình sẽ làm việc tốt. Thực hiện việc kiểm tra này một cách định kỳ sẽ giúp cho bạn khám phá ra các lỗi với các thủ tục sao lưu và như vậy bạn có thể chữa các lỗi đó trước khi bị mất dữ liệu. Một số phần mềm phục hồi sao lưu không phục hồi một cách chính xác đúng bảo vệ tập tin và quyền điều khiển riêng. Kiểm tra các thuộc tính của các tập tin được phục hồi để bảo đảm chúng là đã được set đúng. Việc thử nghiệm một cách định kỳ là để bảo đảm rằng bạn có thể thực hiện một việc phục hồi đầy đủ hệ thống từ các sao lưu của bạn.

Tài liệu bổ sung

Để biết thêm chi tiết, có trang man mà bạn có thể đọc:

tar (1) - The GNU version of the tar archiving utility

Chương trình sao chép dự phòng dump

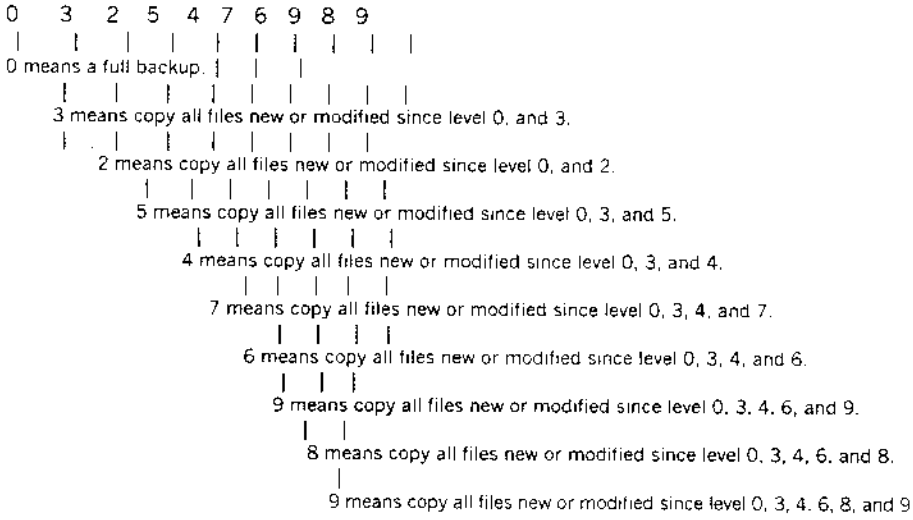
Mô tả

Dump là một công cụ backup khác hoàn toàn so với tar; Nó là một chương trình dành cho việc sao chép dự phòng và phục hồi toàn bộ tập tin hệ thống (không bao gồm các tập tin thường). Dump không xem trọng tập tin hệ thống nào đang trên ổ đĩa cứng, hay thậm chí là có hay không các tập tin trong hệ thống tập tin. Nó kiểm tra xem các tập tin trên một hệ thống tập tin ext2, xác định là các

tập tin nào được sao chép dự phòng và sao chép các tập tin đó vào một đĩa cụ thể nào đó, tape, tập tin hay nơi lưu trữ trung gian nào khác. Nó dump một tập tin hệ thống tại một thời điểm một cách nhanh chóng và hiệu quả. Thật không may, nó không thực hiện trên các thự mục riêng lẻ, và như vậy nó ngốn một không gian lưu trữ nhiều hơn tar. Nó cũng được ghi một cách đặc biệt cho các backup. Lệnh phục hồi thực hiện chức năng đảo ngược lại của dump; Nó có thể phục hồi một full backup của một hệ thống tập tin. Và các sao lưu gia tăng liền theo sau có thể khi đó được phân lớp trên đỉnh đầu của full backup. Các tập tin và cây thư mục đơn lẻ có thể cũng được phục hồi từ full backup hay một phần backup. Bạn có thể dùng dump nếu bạn cần một thủ tục cho cả hai sao chép dự phòng các tập tin hệ thống và phục hồi các tập tin hệ thống sau khi backup.

Các mức độ Dump

Dump có rất nhiều mức độ của các thủ tục backup. Các mức độ thuộc miền giá trị từ 0 tới 9, trong đó số mức độ 0 có nghĩa là một full backup và bảo đảm toàn bộ các hệ thống tập tin được sao chép. Một số mức độ lớn hơn 0, là các sao lưu gia tăng (incremental backup), báo cho dump biết để sao chép tất cả các tập tin mới hay đã được hiệu chỉnh từ lần dump cuối có mức độ thấp hơn hoặc bằng. Để được rõ hơn, ở mỗi mức sao lưu gia tăng, bạn sao chép dự phòng mọi thứ đã được thay đổi từ sao lưu trước đó, ở cùng hay ở một mức độ trước đó. Có các điểm thuận lợi và các lý do để tạo và dùng nhiều mức độ để tạo một backup? Chúng tôi thử giải thích câu hỏi này với lược đồ sau :



Các điểm thuận lợi và các lý do để tạo và dùng nhiều mức độ để tạo một backup, quá trình lưu lại sao lưu có thể được mở rộng hơn. Một quá trình lưu lại sao lưu dài hơn thì tiện lợi, từ khi xoá hay làm hỏng các tập tin thì thông thường không được thông báo trong một khoảng thời gian dài. Thậm chí một phiên bản của một tập tin mà nó không được cập nhật thì cũng tốt hơn là không có tập tin

Tùy chọn "f" trạng thái mà mọi thông số kế tiếp sẽ là tên của tập tin lưu trữ hay thiết bị.

Nếu bạn không cần phục hồi tất cả các tập tin được chứa đựng trong tập tin lưu trữ, bạn có thể chỉ ra một hay nhiều tập tin mà bạn muốn phục hồi:

- Để chỉ rõ một hay nhiều tập tin bạn muốn phục hồi, dùng lệnh sau:

```
[root@deep]# cd /
[root@deep]# tar xpf /dev/st0/full-backup-Day-Month-Year.tar \
home/wahib/Personal/Contents.doc home/quota.user
```

Lệnh trên phục hồi các tập tin "/home/wahib/Personal/Contents.doc" và "/home/quota.user" từ dạng lưu trữ.

- Nếu bạn chỉ muốn xem những tập tin nào là thuộc về bản backup, Dùng tùy chọn –list (-t):

```
[root@deep /]# tar tf /dev/st0
```

Nguyên nhân: *Nếu bạn có các tập tin trên hệ thống của bạn đã được set với bit không thay đổi, thì dùng lệnh "chattr", các tập tin này sẽ không được ghi nhớ với bit không thay đổi từ sao lưu của bạn được phục hồi. Bạn phải khởi tạo lại nó với lệnh "chattr + i" sau khi sao lưu được hoàn thành.*

Kiểm tra khả năng phục hồi từ các sao chép dự phòng.

Đối với nhiều người quản trị hệ thống, việc phục hồi một tập tin từ một sao lưu là một việc làm không thường xuyên. Bước này bảo đảm rằng nếu bạn cần phải phục hồi lại một tập tin, các công cụ và các tiến trình sẽ làm việc tốt. Thực hiện việc kiểm tra này một cách định kỳ sẽ giúp cho bạn khám phá ra các lỗi với các thủ tục sao lưu và như vậy bạn có thể chữa các lỗi đó trước khi bị mất dữ liệu. Một số phần mềm phục hồi sao lưu không phục hồi một cách chính xác đúng bảo vệ tập tin và quyền điều khiển riêng. Kiểm tra các thuộc tính của các tập tin được phục hồi để bảo đảm chúng là đã được set đúng. Việc thử nghiệm một cách định kỳ là để bảo đảm rằng bạn có thể thực hiện một việc phục hồi đầy đủ hệ thống từ các sao lưu của bạn.

Tài liệu bổ sung

Để biết thêm chi tiết, có trang man mà bạn có thể đọc:

tar (1) - The GNU version of the tar archiving utility

Chương trình sao chép dự phòng dump

Mô tả

Dump là một công cụ backup khác hoàn toàn so với tar; Nó là một chương trình dành cho việc sao chép dự phòng và phục hồi toàn bộ tập tin hệ thống (không bao gồm các tập tin thường). Dump không xem trọng tập tin hệ thống nào đang trên ổ đĩa cứng, hay thậm chí là có hay không các tập tin trong hệ thống tập tin. Nó kiểm tra xem các tập tin trên một hệ thống tập tin ext2, xác định là các

nào cả. Các mức độ sao lưu thường được giữ cho cả hai thời gian việc sao lưu và phục hồi là tối thiểu.

Trang hướng dẫn dump đề nghị một chiến lược tốt thực hiện bằng tay để tận dụng hoàn toàn các thuận lợi của các mức độ sao lưu : 3,2 ,5,4,7, 6, 9, 8, 9, v.v.. như được mô tả trong bảng bên dưới. Hầu như bạn phải sao lưu là công việc của hai ngày. Số lượng các tape cho việc phục hồi phụ thuộc vào bạn lưu giữ bao lâu giữa các full backup.

Tape	Level	Backup (days)	Restore tapes
1	0	N/a	1
2	3	1	1, 2
3	2	2	1, 3
4	5	1	1, 2, 4
5	4	2	1, 2, 5
6	7	1	1, 2, 5, 6
7	6	2	1, 2, 5, 7
8	9	1	1, 2, 5, 7, 8
9	8	2	1, 2, 5, 7, 9
10	9	1	1, 2, 5, 7, 9, 10

Tạo các sao lưu với dump

Điều thú vị là sử dụng chương trình sao lưu dump nếu bạn muốn tận dụng các thuận lợi của nhiều mức độ của các thủ tục backup. Dưới đây, tôi trình bày cho bạn một thủ tục để có một quá trình lưu lại sao lưu dài hơn, và để giữ thời gian của cả hai việc sao lưu và phục hồi là nhỏ. Trong ví dụ sau, giả sử rằng chúng ta ghi sao lưu vào một ổ đĩa tape có tên là (/dev/st0) và sao lưu thư mục home (/home) của hệ thống của chúng ta.

Điều quan trọng là luôn bắt đầu với một mức độ sao lưu 0, ví dụ:

- Friday 1, (use tape 1 for the first full backup).

```
[root@deep /]# dump -0u -f /dev/st0 /home
DUMP: Date of this level 0 dump: Fri Jan 28 21:25:12 2000
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/sda6 (/home) to /dev/st0
DUMP: mapping (Pass I) [regular files]
DUMP: mapping (Pass II) [directories]
DUMP: estimated 18582 tape blocks on 0.48 tape(s).
DUMP: Volume 1 started at: Fri Jan 28 21:25:14 2000
DUMP: dumping (Pass III) [directories]
DUMP: dumping (Pass IV) [regular files]
DUMP: DUMP: 18580 tape blocks on 1 volumes(s)
```

DUMP: finished in 4 seconds, throughput 4645 KBytes/sec
DUMP: Volume 1 completed at: Fri Jan 28 21:25:18 2000
DUMP: Volume 1 took 0:00:04
DUMP: Volume 1 transfer rate: 4645 KB/s
DUMP: level 0 dump on Fri Jan 28 21:25:12 2000
DUMP: DUMP: Date of this level 0 dump: Fri Jan 28 21:25:12 2000
DUMP: DUMP: Date this dump completed: Fri Jan 28 21:25:18 2000
DUMP: DUMP: Average transfer rate: 4645 KB/s
DUMP: Closing /dev/st0
DUMP: DUMP IS DONE

- Monday, (Sử dụng tape thứ hai cho backup gia tăng).
  [root@deep /]# dump -3u -f /dev/st0 /home
- Tuesday, (Sử dụng tape thứ ba cho backup gia tăng).
  [root@deep /]# dump -2u -f /dev/st0 /home
- Wednesday, (Sử dụng tape thứ tư cho backup gia tăng).
  [root@deep /]# dump -5u -f /dev/st0 /home
- Thursday, (Sử dụng tape thứ năm cho backup gia tăng).
  [root@deep /]# dump -4u -f /dev/st0 /home
- Friday 2, (Sử dụng tape thứ sáu cho backup gia tăng).
  [root@deep /]# dump -7u -f /dev/st0 /home
- Monday, (Sử dụng tape thứ hai cho backup gia tăng).
  [root@deep /]# dump -3u -f /dev/st0 /home
- Tuesday, (Sử dụng tape thứ ba cho backup gia tăng).
  [root@deep /]# dump -2u -f /dev/st0 /home
- Wednesday, (Sử dụng tape thứ tư cho backup gia tăng).
  [root@deep /]# dump -5u -f /dev/st0 /home
- Thursday, (Sử dụng tape thứ năm cho backup gia tăng).
  [root@deep /]# dump -4u -f /dev/st0 /home
- Friday 3, (Sử dụng tape thứ bảy cho backup gia tăng).
  [root@deep /]# dump -6u -f /dev/st0 /home
- Monday, (Sử dụng tape thứ hai cho backup gia tăng).
  [root@deep /]# dump -3u -f /dev/st0 /home
- Tuesday, (Sử dụng tape thứ ba cho backup gia tăng).
  [root@deep /]# dump -2u -f /dev/st0 /home
- Wednesday, (Sử dụng tape thứ tư cho backup gia tăng).
  [root@deep /]# dump -5u -f /dev/st0 /home
- Thursday, (Sử dụng tape thứ năm cho backup gia tăng).
  [root@deep /]# dump -4u -f /dev/st0 /home

- Friday 4, (Sử dụng tape thứ tám cho backup gia tăng chỉ khi có năm ngày Thứ Sáu trong một tháng).

```
[root@deep /]# dump -9u -f /dev/st0 /home
```

- Monday, (Sử dụng tape thứ hai cho backup gia tăng chỉ khi có năm ngày Thứ Sáu trong một tháng).

```
[root@deep /]# dump -3u -f /dev/st0 /home
```

- Tuesday, (Sử dụng tape thứ ba cho backup gia tăng chỉ khi có năm ngày Thứ Sáu trong một tháng).

```
[root@deep /]# dump -2u -f /dev/st0 /home
```

- Wednesday, (Sử dụng tape thứ tư cho backup gia tăng chỉ khi có năm ngày Thứ Sáu trong một tháng).

```
[root@deep /]# dump -5u -f /dev/st0 /home
```

- Thursday, (Sử dụng tape thứ năm cho backup gia tăng chỉ khi có năm ngày Thứ Sáu trong một tháng).

```
[root@deep /]# dump -4u -f /dev/st0 /home
```

- Month, (Dùng tape khác cho một sao lưu (backup) đầy đủ mới khi tháng thay đổi).

```
[root@deep /]# dump -0u -f /dev/st0 /home
```

Trong đó "-0 tới -9" là tùy chọn mức độ sao lưu mà bạn muốn dùng, tùy chọn "-u" có nghĩa là cập nhật tập tin "/etc/dumpdates" sau lệnh dump thành công, tùy chọn "-f" là ghi backup vào tập tin; tập tin có thể là tập tin thiết bị đặc biệt tựa như "/dev/st0" (một ổ đĩa tape), "/dev/rsd1c" (một ổ đĩa), một tập tin gốc, hay "-" (một output chuẩn). Sau cùng, bạn phải chỉ rõ bạn cần sao lưu cái gì. Trong ví dụ của chúng ta, đó là thư mục home (/home).

Bạn có thể thấy rằng chúng ta dùng cùng các tape 2 tới 5 cho các công việc sao lưu hàng ngày (Thứ 2 tới thứ 6 = 4 tape), tape 6, 7 và 8 cho việc sao lưu hàng tuần (các thứ 6 khác, 6+7+8 = 3 tape; ghi chú là có thể có 5 thứ 6 trong một tháng) và tape 1 và bất kỳ một cái mới nào tiếp theo cho việc sao lưu hàng tháng (Thứ 6 đầu tiên mỗi tháng, 1+ bất kỳ cái tiếp theo nào "11 tháng" = 12 tape). Tóm lại, nếu chúng ta dùng 8 tape (4+3+1=8), chúng ta có thể có một sao lưu đầy đủ và lặp lại thủ tục với 8 tape để lấy 11 tháng tiếp theo sau để được tổng cộng 1 năm các backup đầy đủ riêng rẽ.

Việc backup đầy đủ sẽ được thực hiện ở các khoảng cách thiết lập (interval), như một lần trên một tháng, và trên một tập các tape mà được lưu lại mãi mãi. Với loại thủ tục này, bạn sẽ có 12 tape cho 12 tháng mà xử lý quá trình lưu lại và thay đổi hệ thống của bạn trong một năm. Sau đó, bạn có thể sao chép 12 tape sao lưu trên một máy tính khác được thiết kế để lưu giữ các sao lưu hàng năm trong một thời gian dài và có thể sử dụng lại chúng (12 tape) để lặp lại thủ tục cho một năm mới.

Phục hồi lại các tập tin với dump

Lệnh phục hồi thực hiện chức năng ngược lại của dump (8). Nó phục hồi các tập tin hay các hệ thống tập tin từ các sao lưu được tạo với dump. Một sao lưu đầy đủ của một hệ thống tập tin có thể được phục hồi, và các sao lưu gia tăng tiếp theo sau được phân lớp trên đỉnh đầu của nó. Các cây thư mục và các tập tin có thể được phục hồi từ các sao lưu đầy đủ hay một phần. Bạn có một số các lệnh và các tùy chọn có thể phục hồi các dữ liệu được sao lưu với chương trình dump. Bên dưới, tôi trình bày cho bạn một thủ tục mà sử dụng hoàn toàn tiềm năng của một chương trình phục hồi với hầu hết các tùy chọn có thể. Nó cũng được thực hiện trong chế độ tương tác.

Trong một ảnh hưởng với việc phục hồi của các tập tin từ một dump, chương trình phục hồi cung cấp một tập lệnh giống như giao diện cho phép người sử dụng chuyển xung quanh cây thư mục các tập tin đang chọn để trích ra (extract), sau khi đọc thông tin thư mục từ dump. Sau đây là những gì mà chúng ta sẽ thấy nếu chúng ta cố gắng phục hồi thư mục "/home" của chúng ta:

Trước tiên, phải chuyển tới partition tập tin hệ thống nơi mà bạn muốn phục hồi các sao lưu của chúng ta. Đây là một yêu cầu, bởi vì mode tương tác của chương trình phục hồi sẽ phục hồi các sao lưu của chúng ta từ partition tập tin hệ thống hiện hành, nơi ta phải thi hành lệnh phục hồi từ đó.

- Để chuyển vào partition tập tin hệ thống cần phục hồi (thư mục "/home" trong trường hợp của chúng ta), dùng lệnh sau:

```
[root@deep /]# cd /home
```

- Để phục hồi các tập tin từ một dump trong mode tương tác, dùng lệnh sau:

```
[root@deep /home]# restore -i -f /dev/st0
restore >
```

Một dấu nhắc sẽ xuất hiện trong terminal của bạn để liệt kê hay chỉ rõ thư mục hiện hành. Dùng lệnh "ls" như được trình bày bên dưới:

```
restore > ls
.:
admin/ lost+found/ named/ quota.group quota.user wahib/

restore >
```

Để thay đổi thư mục làm việc hiện hành tới một thư mục được chỉ rõ, dùng lệnh "cd" (trong ví dụ, chúng ta sẽ chuyển tới thư mục *wahib*) như được trình bày bên dưới:

```
restore > cd wahib
restore > ls
./wahib:
.Xdefaults .bash_logout .bashrc
.bash_history .bash_profile Personal/

restore >
```

Để thêm thư mục hay tập tin hiện hành vào danh sách các tập tin có thể được trích ra, dùng lệnh "add" (nếu một thư mục được chỉ rõ, sau đó nó và tất cả tập tin có nguồn gốc của nó được thêm vào danh sách được trích) như được trình bày bên dưới:

```
restore > add Personal/
restore >
```

Các tập tin nằm trên danh sách trích thì được bắt đầu với một dấu "*" khi chúng được liệt kê bởi lệnh "ls":

```
restore > ls
./wahib:
.Xdefaults .bash_logout .bashrc
.bash_history .bash_profile *Personal/
```

Để xóa thư mục hiện hành hay thông số chỉ định từ danh sách các tập tin có thể được trích ra, dùng lệnh "**delete**" (nếu một thư mục được chỉ rõ, thì nó và tập tin có nguồn gốc của nó bị xóa từ danh sách trích) như được trình bày bên dưới:

Ghi chú: *Cách tiện lợi nhất để trích ra hầu hết các tập tin từ một thư mục là thêm thư mục vào danh sách trích và sau đó xóa các tập tin không cần thiết.*

```
restore > cd Personal/
restore > ls
./wahib/Personal:
*Ad?le_Nakad.doc          *Overview.doc
*BIMCOR/                  *Resume/
*My Webs/                 *SAMS/
*Contents.doc             *Templates/
*Divers.doc               *bruno universite.doc
*Linux/                   *My Pictures/

restore > delete Resume/
restore > ls
./wahib/Personal:
*Ad?le_Nakad.doc          *Overview.doc
*BIMCOR/                  Resume/
*My Webs/                 *SAMS/
*Contents.doc             *Templates/
*Divers.doc               *bruno universite.doc
*Linux/                   *My Pictures/
```

Để trích ra tất cả các tập tin trong danh sách trích từ dump, dùng lệnh "extract" (Phục hồi sẽ hỏi phiên bản nào mà user muốn mount. Cách nhanh nhất để trích ra một ít tập tin là khởi đầu với volume sau cùng và làm việc hướng về volume đầu tiên) như được trình bày bên dưới:

```
restore > extract
You have not read any tapes yet.
Unless you know which volume your file(s) are on you should start
with the last volume and work towards the first.
Specify next volume #: 1
set owner/mode for '.'? [yn] y
```

Để thoát từ mode tương tác phục hồi sau khi bạn đã kết thúc xong việc trích ra các thư mục hay các tập tin của bạn, dùng lệnh "quit' như được trình bày bên dưới:

```
/sbin/restore > quit
```

Ghi chú: *Đã có các phương pháp phục hồi khác với chương trình dump; xem các trang man của dump để biết thêm chi tiết.*

Tài liệu bổ sung

Để biết thêm thông tin chi tiết, có một số trang man bạn có thể đọc:

```
$ man dump (8) - ext2 filesystem backup
$ man restore (8) - "restore files or taäp tin systems from backups made with dump"
```

Sao chép dự phòng và phục hồi thông qua mạng

Các sao lưu cho phép bạn phục hồi lại các tài nguyên thông tin chính xác và toàn vẹn do các rủi ro và các sự vi phạm bảo mật gây nên. Không có quá trình backup, bạn không thể phục hồi dữ liệu của một máy tính sau khi hệ thống bị hỏng và các vi phạm bảo mật. Điều quan trọng là đưa ra một kế hoạch đủ bao phủ tất cả các server mà bạn có kế hoạch triển khai. Chúng ta phải xác định các nhóm tập tin nào sẽ được backup. Lấy ví dụ, ban có thể chọn chỉ sao lưu các tập tin dữ liệu của user (nghĩa là /home) bởi vì tập tin hệ thống bị hư hỏng sẽ được nạp lại từ các CR-ROM của chương trình nguồn.

Có nhiều kỹ thuật phổ biến gắn liền với các sao lưu tập tin. Đối với các server mạng, một phiên bản có căn cứ xác thực nội dung thông tin của server được tạo và được bảo quản trên một máy bảo mật mà nó được backup. Nếu server bị làm hại thì nội dung của nó bị hỏng, nó có thể được nạp lại từ một hệ thống bảo mật và duy trì phiên bản có căn cứ xác thực. Việc thăm dò này là tiêu biểu được dùng cho các server public, ví dụ như các Web server, bởi vì nội dung thay đổi ở nhiều khoảng cách có thể dự báo trước.

Điều quan trọng là bảo đảm rằng các sao lưu được thực hiện trong một tình trạng bảo mật và nội dung của các sao lưu còn lại phải bảo mật. Chúng tôi khuyên bạn là kế hoạch phải chỉ rõ:

- Nguồn dữ liệu được mã hoá trước khi được truyền tới nơi lưu trữ trung gian.
- Dữ liệu còn lại được mã hóa trên phương tiện lưu trữ dự phòng.
- Phương tiện lưu trữ được giữ trong môi trường bảo mật vật lý và tránh được các tai hoạ có thể gây ra bởi con người hoặc tự nhiên.

Truyền tải các sao lưu của bạn trong một tình trạng bảo mật thông qua mạng

Trong các phần trước đó, tôi đã trình bày cho bạn phải tạo sao lưu như thế nào trên cả hai một tape và các tập tin từ cùng hệ thống nơi bạn thi hành thủ tục

backup, với các tiện ích như tar và dump. Các chương trình này (tar và dump) cũng có khả năng tạo sao lưu tốt thông qua mạng. Để có thể sao lưu thông qua mạng, bạn phải bảo đảm rằng các package có tên "rmt" và "rsh" được cài đặt trên hệ thống của bạn. Tiện ích "rmt" cung cấp truy cập từ xa tới các thiết bị tape dành cho các chương trình như dump và tar. Để bổ sung phần này, package "rsh" chứa đựng một tập các chương trình cho phép các user thi hành các lệnh trên các máy từ xa, login vào các máy khác và sao chép các tập tin giữa các máy (rsh, rlogin và rcp là tập hợp các chương trình này).

Bởi vì "rsh" có thể bị tấn công dễ dàng, và "rmt" phụ thuộc vào "rsh" để có thể làm việc được, chúng ta đã chọn không cài đặt chúng trong tiến trình cài đặt của chúng ta (xem chương 2, "Cài đặt Linux Server" để biết thêm chi tiết) vì các lý do bảo mật. Chính vì vậy, chúng ta phải tìm một cách khác để tạo các sao lưu thông qua mạng trong một tình trạng bảo mật. Công nghệ SSH là giải pháp cho vấn đề (xem chương 11, Phần mềm bảo mật và Bảo mật mạng) bởi vì nó cũng có khả năng sao chép dữ liệu thông qua mạng với lệnh "scp" của nó, thông qua việc mã hóa. Sau đây là phương pháp cho phép chúng ta dùng tiềm năng của phần mềm SSH để chuyển giao các sao lưu được tạo của bạn với tar hay dump trong một tình trạng bảo mật thông qua tiện ích "scp" SSH.

Dùng lệnh SSH "scp" để chuyển giao các sao lưu thông qua mạng

Lệnh SSH "scp" sao chép các tập tin giữa các host trên một mạng. Nó dùng SSH cho việc truyền dữ liệu, và sử dụng cùng tính năng kiểm tra tính xác thực (authentication), và cung cấp việc bảo mật giống nhau, như SSH. Không giống như tiện ích "rcp" mà đi cùng với package "rsh", "scp" sẽ đòi mật mã. Trong ví dụ bên dưới, chúng ta truyền một tập tin sao lưu với chương trình lưu trữ tar; và thủ tục truyền một tập tin sao lưu hay tape được tạo với chương trình dump là như nhau.

- Dùng scp để sao chép một tập tin sao lưu hay tape tới một hệ thống bảo mật từ xa, dùng lệnh:

```
[admin@deep /]# scp <localdir/to/filelocation> <user@host:/dir/for/file>
```

Trong đó <localdir/to/filelocation> là thư mục ở đó tập tin sao lưu của bạn nằm trên máy server cục bộ, và <user@host:/dir/for/file> biểu diễn theo thứ tự username (user) trên site ở nơi sẽ lưu giữ tập tin backup, hostname (host) của host ở xa nơi bạn muốn gửi tập tin backup và thư mục của host từ xa nơi bạn muốn đặt tập sao lưu được truyền.

Một ví dụ thật sẽ giống như sau:

```
[admin@deep /]# scp -Cp /backups/deep-01Feb.tar
admin@backupserver:/archive/deep/deep-01Feb.tar
admin@backupserver's password:
deep-01Feb.tgz | 10479 KB | 154.1 kB/s | ETA: 00:00:00 | 100%
```

Ghi chú: *Tùy chọn "C" hỗ trợ tính năng nén dữ liệu cho việc truyền dữ liệu nhanh thông qua phiên (session) được mã hóa, tùy chọn "p" chỉ rõ việc hiệu*

chỉnh và thời gian truy cập cũng như các mode của tập tin nguồn sẽ được bảo vệ khi sao chép. Đây là một mong muốn thường xuyên. Điều quan trọng đáng lưu ý là thư mục "dir/for/file" trên host ở xa (trong ví dụ này là "/archive/deep") phải được sở hữu riêng bởi "username" mà bạn chỉ rõ trong lệnh scp của bạn ("admin" là username trong ví dụ này) bằng không. bạn có thể nhận thông điệp lỗi như : scp: /archive/deep/deep-01Feb.tar: Permission denied.

- Dùng lệnh scp để sao chép một tape hay tập tin ở xa tới hệ thống cục bộ, dùng lệnh:

```
[admin@deep /]# scp <user@host:/dir/for/file> <localdir/to/filelocation>
```

Trong đó user@host:/dir/for/file ngầm hiểu username (user) của người trên site ở xa nơi sẽ lưu giữ tập tin backup, hostname (host) là tên của host ở xa nơi bạn muốn gửi tập tin backup và thư mục của host từ xa nơi bạn muốn lấy tập backup, thư mục từ xa của host này là nơi mà tập tin sao lưu được lưu giữ, và <localdir/to/filelocation> là thư mục trên hệ thống của bạn nơi bạn muốn đặt tập tin sao lưu mà bạn lấy từ máy host ở xa.

Một ví dụ thật sẽ giống như sau:

```
[admin@deep /]# scp -Cp admin@backupserver:/archive/deep/deep-01Feb.tar /backups
admin@backupserver's password:
deep-01Feb.tgz | 10479 KB | 154.1 kB/s | ETA: 00:00:00 | 100%
```

Ghi chú: *Điều quan trọng đáng lưu ý là thư mục "location/to/filelocation" trên host cục bộ (trong ví dụ này là "/backups") phải được sở hữu riêng bởi "username" được chỉ rõ trong lệnh scp ("admin" là username trong ví dụ của chúng ta) bằng không bạn có thể nhận một thông điệp lỗi như: scp: /archive/deep/deep-01Feb.tar: Permission denied.*

Các web site tham khảo lệnh tar và dump

AMANDA

AMANDA Homepage: http://www.cs.umd.edu/projects/amanda/

BRU

BRU Homepage: http://www.bru.com/

PHẦN 6

CÀI ĐẶT VÀ CẤU HÌNH MỘT SỐ PHẦN MỀM TRÊN RED HAT LINUX 7.2

Trong phần này chúng ta sẽ đề cập đến các chương sau:

- **Cài đặt Web Server Apache 1.3.20 trên Red Hat Linux 7.2.**
- **Cài đặt LINUX FREES/WAN VPN trên Red Hat Linux 7.2.**
- **Cài đặt Squid Proxy Server trên Red Hat Linux 7.2.**
- **Cài đặt OpenSSH Client/Server trên Red Hat Linux 7.2.**
- **Cài đặt và sử dụng Samba trong Red Hat Linux 7.2.**

Chương 23:

WEB SERVER APACHE 1.3.20 TRÊN Red Hat LINUX 7.2

Trong chương này chúng ta sẽ đề cập đến các vấn đề chính như sau:

- Biên dịch và cài đặt apache.
- Cài đặt Apache từ tập tin apache-1.3.20-16.i386.rpm.
- Cấu hình tập tin /etc/httpd/conf/httpd.conf.

Tổng quan:

Apache vẫn là một webserver rất mạnh và cũng là một phần mềm được chọn làm web server để chạy trên hầu hết các server Linux. Do đáp ứng được nhiều yêu cầu của người dùng như phát triển ứng dụng, kết nối với các hệ database thông dụng, khả năng bảo mật, chạy trên nhiều hệ điều hành khác nhau và một điều mà chúng ta đều biết là nó miễn phí. Trong phần này chúng tôi sẽ cung cấp những thay đổi của apache1.3.20 so với các phiên bản trước cũng như cách cài đặt và một vài tính năng mới của chúng.

Trong phiên bản mới này chủ yếu sửa những lỗi về bảo mật cùng một vài cấu hình quan trọng trong phiên bản 1.3.19. Một vài sự thay đổi cùng những đặc trưng mới có thể được tóm tắt như sau:

Những sửa lỗi về bảo mật.

Một URI được xây dựng một cách cẩn trọng là nguyên nhân làm an toàn các apache web server chạy trên Linux, Win32 và OS/2. Chúng sẽ từ chối các yêu cầu truy cập từ user đến khi các lỗi được tìm ra, khả năng nhận biết tác động của DOS nên tránh được sự nguy hiểm của dữ liệu trên server.

Những sửa lỗi chung:

Loại bỏ những lỗi quan trọng liên quan đến server nếu một giá trị không hợp lệ xảy ra trong hàm ap_snprintf() trên những môi trường hỗ trợ các hàm isnan() và insinf().

Sửa lại chương trình ở quá trình khởi động khi tìm kiếm một server name hoặc IP khi không có server name được chỉ định.

Những đặc trưng mới bao gồm:

Quay vòng logfile với thời gian được chỉ định, định dạng các tên logfile có thể tương ứng với tên của người.

Thêm ký tự '\' để' viết lại những qui tắc cho phép đưa vào những ký tự đặc biệt. Cho phép đưa vào hai ký tự '$' và '%' ví dụ như 'foo\$1' sẽ được xem như 'foo$1'.

Download và cài đặt apache

Bạn có thể download Apache 1.3.20 tại địa chỉ sau:

http://httpd.apache.org/dist/httpd/

Hoặc bạn có thể download tại các website khác như:

http://httpd.apache.org/dist/httpd/binaries/ cho Apache dạng nhị phân

hoặc http://www.apache.org/mirrors/ nếu bạn cần Apache ở dạng binary hoặc source.

Để đơn giản nhất bạn có thể download tập tin apache_1.3.20.src.rpm hoặc apache-1.3.20-16.i386.rpm trên đĩa cài đặt Red Hat Linux7.2 hoặc từ ftp server

của Red Hat. Bạn cũng có thể download về tập tin apache_1.3.20.tar.gz từ FTP server hoặc một trong những web site đã cung cấp ở trên.

Biên dịch và cài đặt Apache

Mặc định Apache 1.3.20 trên Linux 7.2 có những thay đổi về vị trí so với Linux 6.2. Các tập tin cấu hình bây giờ sẽ được đặt tại vị trí /etc/httpd/conf/httpd.conf và các thư mục con nằm dưới thư mục /home/httpd sẽ được đặt trong /var/www.

Nếu bạn cài đặt Apache từ tập tin apache-1.3.20-16.i386.rpm, thì vị trí của các tập tin cấu hình và thư mục chứa các tập tin .html (htdocs) theo mặc định sẽ nằm ở /etc/httpd/conf/httpd.conf và /var/www.

Biên dịch apache

Giả sử bạn đã download về tập tin apache_1.3.20.src.rpm và lưu ở thư mục /tmp. Trước khi biên dịch chạy lệnh sau để giải nén:

```
root # cd tmp
root # rpm –ivh apache_1.3.20.src.rpm
root # cd /usr/src/Red Hat/SOURCES
root # tar zxf apache_1.3.20.tar.gz
root # cd apache_1.3.20
```

Quá trình biên dịch Apache trên Ret Hat Linux có thể bao gồm ba bước chính: Đầu tiên chọn các module nào bạn muốn đưa vào trong web server, thứ hai tạo một cấu hình cho hệ thống của bạn, thứ ba biên dịch thành tập tin có thể thực thi được.

Tất cả cấu hình của Apache được thực hiện trong thư mục src của thư mục Apache sau khi giải nén. Chuyển vào trong thư mục này.

```
root # cd src
```

Chọn các module để biên dịch trong Apache trong tập tin Configuration. Những dòng không có dấu # tương ứng với những module tùy chọn bạn muốn đưa vào (giữa các dòng AddModule ở cuối của tập tin hoặc thêm những những dòng mới tương ứng với các module bạn viết hoặc download).

Cấu hình Apache cho hệ thống của bạn. Bình thường bạn có thể chạy script configure như trình bày dưới đây. Tuy nhiên, nếu việc này không thành công hoặc bạn có bất kỳ những yêu cầu đặc biệt (ví dụ đưa thêm vào thư viện được yêu cầu bởi một module tùy chọn) bạn có thể cần hiệu chỉnh một hoặc nhiều tùy chọn trong tập tin Configuration: EXTRA_CFLAGS, LIBS, LDFLAGS, INCLUDES.

Chạy script Configure trên màn hình có dạng sau:

```
% Configure
  Using 'Configuration' as config file
  + configured for <whatever> platform
  + setting C compiler to <whatever> *
```

```
+ setting C compiler optimization-level to <whatever> *
+ Adding selected modules
+ doing sanity check on compiler and options
Creating Makefile in support
Creating Makefile in main
Creating Makefile in os/unix
Creating Makefile in modules/standard
```

Lệnh trên sẽ sinh ra một tập tin Makefile để dùng cho giai đoạn 3. Nó cũng tạo một tập tin Makefile trong thư mục support cho việc biên dịch của những chương trình hỗ trợ tùy chọn.

Nếu bạn có nhiều tập tin cấu hình bạn có thể báo cho script Configure biết nó đọc tập tin cấu hình nào với tùy chọn –file, ví dụ như Configure –file Configuration.ai).

```
Root # make
```

Chạy lệnh make để biên dịch apache.

Các module trong package Apache đã được kiểm tra và sử dụng hằng ngày của các thành viên trong nhóm phát triển apache, một vài module khác được phát triển bởi một số người cho các mục đích riêng của họ có thể tham khảo tại địa chỉ http://www.apache.org/dist/httpd/contrib/modules.

Cài đặt Apache

Chúng ta đã có tập tin nhị phân httpd trong thư mục src, bây giờ sẽ cài đặt chúng vào hệ thống Linux của chúng ta. Cài đặt Apache đơn giản thực hiện lệnh:

```
root # make install
```

Apache được thiết kế để cấu hình và chạy từ cùng thư mục ở đó nó được biên dịch. Nếu bạn muốn chạy Apache từ một nơi nào đó, tạo một thư mục và copy các thư mục conf, logs và icons vào trong thư mục vừa tạo, trong trường hợp này bạn cũng cần thiết đặt permission cho thư mục đã tạo cho những nguyên nhân bảo mật (security).

Để start/stop apache, chuyển vào thư mục bạn đã biên dịch Linux và thực hiện các lệnh sau:

```
root # cd apache-version/bin
root # cd bin
root # apachectl start (start apache)
root # apachectl stop (stop apache)
```

Bước kế tiếp là bạn hiệu chỉnh tập tin httpd.conf để làm việc cho web site của bạn.

Cài đặt Apache từ tập tin apache-1.3.20-16.i386.rpm

Bạn có thể nhận tập tin này từ CDROM cài đặt, các bước tiến hành như sau:

```
root #mount /mnt/cdrom
root # cd /mnt/cdrom/Red Hat/RPMS
root # rpm –ivh apache-1.3.20-16.i386.rpm
```

Sau khi thực hiện các lệnh trên Apache đã được cài đặt trên Linux với vị trí mặc định như sau:

Các tập tin cấu hình : /etc/httpd/conf/*.conf

Script để start/stop apache: /etc/rc.d/init.d/httpd

Cấu hình tập tin /etc/httpd/conf/httpd.conf

Hiệu chỉnh tập tin "/etc/httpd/conf/httpd.conf" (vi /etc/httpd/conf/httpd.conf) thay đổi hoặc thêm vào tập tin httpd.conf những nội dung sau.

Trong tập tin cấu hình dưới đây tôi không có ý định diễn giải ý nghĩa từng tùy chọn, bạn có thể tham khảo các diễn giải này trong phần cài đặt và cấu hình Apache trên Red Hat Linux6.2 trong cuốn sách này.

```
### Section 1: Global Environment
#
ServerType standalone

# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
ServerRoot "/etc/httpd"

# The LockFile directive sets the path to the lockfile used when Apache
# is compiled with either USE_FCNTL_SERIALIZED_ACCEPT or
# USE_FLOCK_SERIALIZED_ACCEPT. This directive should normally be left at
LockFile /var/run/httpd.lock

# PidFile: The file in which the server should record its process
# identification number when it starts.
PidFile /var/run/httpd.pid

# ScoreBoardFile: taäp tin used to store internal server process information.
ScoreBoardFile logs/apache_runtime_status

# Timeout: The number of seconds before receives and sends time out.
Timeout 300

# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
```

```
KeepAlive Off

MaxKeepAliveRequests 100

# KeepAliveTimeout: Number of seconds to wait for the next request from the
# same client on the same connection.
KeepAliveTimeout 15

MinSpareServers 5
MaxSpareServers 20

#
StartServers 8

# Limit on total number of servers running, i.e., limit on the number
# of clients who can simultaneously connect --- if this limit is ever
# reached, clients will be LOCKED OUT, so it should NOT BE SET TOO LOW.
# It is intended mainly as a brake to keep a runaway server from taking
# the system with it as it spirals down...
MaxClients 150

MaxRequestsPerChild 1000

# Dynamic Shared Object (DSO) Support

#LoadModule mmap_static_module        modules/mod_mmap_static.so
LoadModule vhost_alias_module        modules/mod_vhost_alias.so
<IfDefine HAVE_BANDWIDTH>
LoadModule bandwidth_module          modules/mod_bandwidth.so
</IfDefine>
<IfDefine HAVE_THROTTLE>
LoadModule throttle_module           modules/mod_throttle.so
</IfDefine>
LoadModule env_module                modules/mod_env.so
LoadModule config_log_module         modules/mod_log_config.so
LoadModule agent_log_module          modules/mod_log_agent.so
LoadModule referer_log_module        modules/mod_log_referer.so
#LoadModule mime_magic_module        modules/mod_mime_magic.so
LoadModule mime_module               modules/mod_mime.so
LoadModule negotiation_module        modules/mod_negotiation.so
LoadModule status_module             modules/mod_status.so
```

```
LoadModule info_module                  modules/mod_info.so
LoadModule includes_module              modules/mod_include.so
LoadModule autoindex_module             modules/mod_autoindex.so
LoadModule dir_module                   modules/mod_dir.so
LoadModule cgi_module                   modules/mod_cgi.so
LoadModule asis_module                  modules/mod_asis.so
LoadModule imap_module                  modules/mod_imap.so
LoadModule action_module                modules/mod_actions.so
#LoadModule speling_module              modules/mod_speling.so
LoadModule userdir_module               modules/mod_userdir.so
LoadModule alias_module                 modules/mod_alias.so
LoadModule rewrite_module               modules/mod_rewrite.so
LoadModule access_module                modules/mod_access.so
LoadModule auth_module                  modules/mod_auth.so
LoadModule anon_auth_module             modules/mod_auth_anon.so
LoadModule db_auth_module               modules/mod_auth_db.so
#LoadModule auth_any_module             modules/mod_auth_any.so
#LoadModule dbm_auth_module             modules/mod_auth_dbm.so
#LoadModule auth_ldap_module            modules/mod_auth_ldap.so
#LoadModule mysql_auth_module           modules/mod_auth_mysql.so
#LoadModule auth_pgsql_module           modules/mod_auth_pgsql.so
#LoadModule digest_module               modules/mod_digest.so
#LoadModule proxy_module                modules/libproxy.so
#LoadModule cern_meta_module            modules/mod_cern_meta.so
LoadModule expires_module               modules/mod_expires.so
LoadModule headers_module               modules/mod_headers.so
#LoadModule usertrack_module            modules/mod_usertrack.so
#LoadModule example_module              modules/mod_example.so
#LoadModule unique_id_module            modules/mod_unique_id.so
LoadModule setenvif_module              modules/mod_setenvif.so
<IfDefine HAVE_PERL>
LoadModule perl_module                  modules/libperl.so
</IfDefine>
<IfDefine HAVE_PHP>
LoadModule php_module                   modules/mod_php.so
</IfDefine>
<IfDefine HAVE_PHP3>
LoadModule php3_module                  modules/libphp3.so
</IfDefine>
<IfDefine HAVE_PHP4>
LoadModule php4_module                  modules/libphp4.so
```

```
</IfDefine>
<IfDefine HAVE_DAV>
LoadModule dav_module                modules/libdav.so
</IfDefine>
<IfDefine HAVE_ROAMING>
LoadModule roaming_module            modules/mod_roaming.so
</IfDefine>
<IfDefine HAVE_SSL>
LoadModule ssl_module                modules/libssl.so
</IfDefine>
<IfDefine HAVE_PUT>
LoadModule put_module                modules/mod_put.so
</IfDefine>
<IfDefine HAVE_PYTHON>
LoadModule python_module             modules/mod_python.so
</IfDefine>

# Reconstruction of the complete module list from all available modules
# (static and shared ones) to achieve correct module execution order.
# [WHENEVER YOU CHANGE THE LOADMODULE SECTION ABOVE UPDATE THIS, TOO]
ClearModuleList
#AddModule mod_mmap_static.c
AddModule mod_vhost_alias.c
<IfDefine HAVE_BANDWIDTH>
AddModule mod_bandwidth.c
</IfDefine>
<IfDefine HAVE_THROTTLE>
AddModule mod_throttle.c
</IfDefine>
AddModule mod_env.c
AddModule mod_log_config.c
AddModule mod_log_agent.c
AddModule mod_log_referer.c
#AddModule mod_mime_magic.c
AddModule mod_mime.c
AddModule mod_negotiation.c
AddModule mod_status.c
AddModule mod_info.c
AddModule mod_include.c
AddModule mod_autoindex.c
AddModule mod_dir.c
```

```
AddModule mod_cgi.c
AddModule mod_asis.c
AddModule mod_imap.c
AddModule mod_actions.c
#AddModule mod_speling.c
AddModule mod_userdir.c
AddModule mod_alias.c
AddModule mod_rewrite.c
AddModule mod_access.c
AddModule mod_auth.c
AddModule mod_auth_anon.c
AddModule mod_auth_db.c
#AddModule mod_auth_any.c
#AddModule mod_auth_dbm.c
#AddModule auth_ldap.c
#AddModule mod_auth_mysql.c
#AddModule mod_auth_pgsql.c
#AddModule mod_digest.c
#AddModule mod_proxy.c
#AddModule mod_cern_meta.c
AddModule mod_expires.c
AddModule mod_headers.c
#AddModule mod_usertrack.c
#AddModule mod_example.c
#AddModule mod_unique_id.c
AddModule mod_so.c
AddModule mod_setenvif.c
<IfDefine HAVE_PERL>
AddModule mod_perl.c
</IfDefine>
<IfDefine HAVE_PHP>
AddModule mod_php.c
</IfDefine>
<IfDefine HAVE_PHP3>
AddModule mod_php3.c
</IfDefine>
<IfDefine HAVE_PHP4>
AddModule mod_php4.c
</IfDefine>
<IfDefine HAVE_DAV>
AddModule mod_dav.c
```

```
</IfDefine>
<IfDefine HAVE_ROAMING>
AddModule mod_roaming.c
</IfDefine>
<IfDefine HAVE_SSL>
AddModule mod_ssl.c
</IfDefine>
<IfDefine HAVE_PUT>
AddModule mod_put.c
</IfDefine>
<IfDefine HAVE_PYTHON>
AddModule mod_python.c
</IfDefine>


### Section 2: 'Main' server configuration
Port 80

##  SSL Support
<IfDefine HAVE_SSL>
Listen 80
Listen 443
</IfDefine>

User apache
Group apache

ServerAdmin root@localhost
ServerName localhost
DocumentRoot "/var/www/html"

<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>

<Directory "/var/www/html">

    Options Indexes FollowSymLinks

    AllowOverride None
```

```
#
# Controls who can get stuff from this server.
#
    Order allow,deny
    Allow from all
</Directory>

<IfModule mod_userdir.c>
    UserDir public_html
</IfModule>

<IfModule mod_dir.c>
    DirectoryIndex index.html index.htm index.shtml index.php index.php4 index.php3
index.phtml index.cgi
</IfModule>
AccessFileName .htaccess

<Files ~ "^\.ht">
    Order allow,deny
    Deny from all
</Files>

UseCanonicalName On

<IfModule mod_mime.c>
    TypesConfig /etc/mime.types
</IfModule>

DefaultType text/plain

<IfModule mod_mime_magic.c>
#   MIMEMagicFile /usr/share/magic.mime
    MIMEMagicFile conf/magic
</IfModule>

HostnameLookups Off

#
ErrorLog logs/error_log

#
```

```
LogLevel warn

#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

# CustomLog /var/log/httpd/access_log common
CustomLog logs/access_log combined

ServerSignature On

#
# Aliases: Add here as many aliases as you need (with no limit). The format is
# Alias fakename realname
#
<IfModule mod_alias.c>

    Alias /icons/ "/var/www/icons/"

    <Directory "/var/www/icons">
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"

    <Directory "/var/www/cgi-bin">
        AllowOverride None
        Options None
        Order allow,deny
        Allow from all
    </Directory>

</IfModule>
```

```
#
# Controls who can get stuff from this server.
#
    Order allow,deny
    Allow from all
</Directory>

<IfModule mod_userdir.c>
    UserDir public_html
</IfModule>

<IfModule mod_dir.c>
    DirectoryIndex index.html index.htm index.shtml index.php index.php4 index.php3
index.phtml index.cgi
</IfModule>
AccessFileName .htaccess

<Files ~ "^\.ht">
    Order allow,deny
    Deny from all
</Files>

UseCanonicalName On

<IfModule mod_mime.c>
    TypesConfig /etc/mime.types
</IfModule>

DefaultType text/plain

<IfModule mod_mime_magic.c>
#   MIMEMagicFile /usr/share/magic.mime
    MIMEMagicFile conf/magic
</IfModule>

HostnameLookups Off

#
ErrorLog logs/error_log

#
```

```
LogLevel warn

#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

# CustomLog /var/log/httpd/access_log common
CustomLog logs/access_log combined

ServerSignature On

#
# Aliases: Add here as many aliases as you need (with no limit). The format is
# Alias fakename realname
#
<IfModule mod_alias.c>

    Alias /icons/ "/var/www/icons/"

    <Directory "/var/www/icons">
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"

    <Directory "/var/www/cgi-bin">
        AllowOverride None
        Options None
        Order allow,deny
        Allow from all
    </Directory>

</IfModule>
```

```
# End of aliases.

<IfModule mod_autoindex.c>

    #
    # FancyIndexing is whether you want fancy directory indexing or standard
    #
    IndexOptions FancyIndexing NameWidth=*

    #
    # AddIcon* directives tell the server which icon to show for different
    # files or filename extensions.  These are only displayed for
    # FancyIndexed directories.
    #
    AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip

    AddIconByType (TXT,/icons/text.gif) text/*
    AddIconByType (IMG,/icons/image2.gif) image/*
    AddIconByType (SND,/icons/sound2.gif) audio/*
    AddIconByType (VID,/icons/movie.gif) video/*

    AddIcon /icons/binary.gif .bin .exe
    AddIcon /icons/binhex.gif .hqx
    AddIcon /icons/tar.gif .tar
    AddIcon /icons/world2.gif .wrl .wrl.gz .vrml .vrm .iv
    AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
    AddIcon /icons/a.gif .ps .ai .eps
    AddIcon /icons/layout.gif .html .shtml .htm .pdf
    AddIcon /icons/text.gif .txt
    AddIcon /icons/c.gif .c
    AddIcon /icons/p.gif .pl .py
    AddIcon /icons/f.gif .for
    AddIcon /icons/dvi.gif .dvi
    AddIcon /icons/uuencoded.gif .uu
    AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
    AddIcon /icons/tex.gif .tex
    AddIcon /icons/bomb.gif core

    AddIcon /icons/back.gif ..
    AddIcon /icons/hand.right.gif README
    AddIcon /icons/folder.gif ^^DIRECTORY^^
```

```
    AddIcon /icons/blank.gif ^^BLANKICON^^

    #
    # DefaultIcon is which icon to show for files which do not have an icon
    # explicitly set.
    #
    DefaultIcon /icons/unknown.gif

    IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t

</IfModule>
# End of indexing directives.

#
# Document types.
#
<IfModule mod_mime.c>

    AddEncoding x-compress Z
    AddEncoding x-gzip gz tgz

    <IfModule mod_php3.c>
        AddType application/x-httpd-php3 .php3
        AddType application/x-httpd-php3-source .phps
    </IfModule>
    #
    # And for PHP 4.x, use:
    #
    <IfModule mod_php4.c>
        AddType application/x-httpd-php .php .php4 .php3 .phtml
        AddType application/x-httpd-php-source .phps
    </IfModule>

    #
    # For PHP/FI (PHP2), use:
    #
    <IfModule mod_php.c>
        AddType application/x-httpd-php .phtml
    </IfModule>

    AddType application/x-tar .tgz
```

```
    AddType text/html .shtml
    AddHandler server-parsed .shtml

    AddHandler imap-file map

</IfModule>

Alias /doc/ /usr/share/doc/
<Directory /usr/share/doc>
    order deny,allow
    deny from all
    allow from localhost .localdomain
    Options Indexes FollowSymLinks
</Directory>

## SSL Global Context
##
## All SSL configuration in this context applies both to
## the main server and all SSL-enabled virtual hosts.
##

#
#  Some MIME-types for downloading Certificates and CRLs
#
<IfDefine HAVE_SSL>
AddType application/x-x509-ca-cert .crt
AddType application/x-pkcs7-crl    .crl
</IfDefine>

<IfModule mod_ssl.c>

SSLPassPhraseDialog  builtin

#  Inter-Process Session Cache:
#  Configure the SSL Session Cache: First the mechanism
#  to use and second the expiring timeout (in seconds).
#SSLSessionCache        none
#SSLSessionCache        shmht:logs/ssl_scache(512000)
#SSLSessionCache        shmcb:logs/ssl_scache(512000)
SSLSessionCache         shm:logs/ssl_scache(512000)
```

```
SSLSessionCacheTimeout  300

#  Semaphore:
#  Configure the path to the mutual exclusion semaphore the
#  SSL engine uses internally for inter-process synchronization.
SSLMutex  file:logs/ssl_mutex

SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
#SSLRandomSeed startup file:/dev/random  512
#SSLRandomSeed startup file:/dev/urandom 512
#SSLRandomSeed connect file:/dev/random  512
#SSLRandomSeed connect file:/dev/urandom 512

#  Logging:
#  The home of the dedicated SSL protocol logfile. Errors are
#  additionally duplicated in the general error log file.  Put
#  this somewhere where it cannot be used for symlink attacks on
#  a real server (i.e. somewhere where only root can write).
#  Log levels are (ascending order: higher ones include lower ones):
#  none, error, warn, info, trace, debug.
SSLLog      logs/ssl_engine_log
SSLLogLevel error

</IfModule>

<IfDefine HAVE_SSL>

##
## SSL Virtual Host Context
##

<VirtualHost _default_:443>

#  General setup for the virtual host
#DocumentRoot "/etc/httpd/htdocs"
#ServerName new.host.name
#ServerAdmin you@your.address
ErrorLog logs/error_log
TransferLog logs/access_log
```

```
#  SSL Engine Switch:
#  Enable/Disable SSL for this virtual host.
SSLEngine on

SSLCertificateFile /etc/httpd/conf/ssl.crt/server.crt
#SSLCertificateFile /etc/httpd/conf/ssl.crt/server-dsa.crt

#  Server Private Key:
SSLCertificateKeyFile /etc/httpd/conf/ssl.key/server.key
#SSLCertificateKeyFile /etc/httpd/conf/ssl.key/server-dsa.key

<Files ~ "\.(cgi|shtml|phtml|php3?)$">
    SSLOptions +StdEnvVars
</Files>
<Directory "/var/www/cgi-bin">
    SSLOptions +StdEnvVars
</Directory>

SetEnvIf User-Agent ".*MSIE.*" \
         nokeepalive ssl-unclean-shutdown \
         downgrade-1.0 force-response-1.0

CustomLog logs/ssl_request_log \
          "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"

</VirtualHost>

</IfDefine>
```

Trong cấu hình trên giả sử rằng bạn đã có openssl được cài đặt trên hệ thống của bạn.

Các tập tin được cài đặt

/etc/httpd

/etc/httpd/conf

/etc/httpd/conf/access.conf

/etc/httpd/conf/httpd.conf

/etc/httpd/conf/magic

/etc/httpd/conf/srm.conf

/etc/httpd/logs
/etc/httpd/modules
/etc/logrotate.d/apache
/etc/rc.d/init.d/httpd
/usr/bin/dbmmanage
/usr/bin/htdigest
/usr/bin/htpasswd
/usr/lib/apache
/usr/lib/apache/httpd.exp
/usr/lib/apache/libproxy.so
/usr/lib/apache/mod_access.so
/usr/lib/apache/mod_actions.so
/usr/lib/apache/mod_alias.so
/usr/lib/apache/mod_asis.so
/usr/lib/apache/mod_auth.so
/usr/lib/apache/mod_auth_anon.so
/usr/lib/apache/mod_auth_db.so
/usr/lib/apache/mod_auth_dbm.so
/usr/lib/apache/mod_autoindex.so
/usr/lib/apache/mod_cern_meta.so
/usr/lib/apache/mod_cgi.so
/usr/lib/apache/mod_digest.so
/usr/lib/apache/mod_dir.so
/usr/lib/apache/mod_env.so
/usr/lib/apache/mod_example.so
/usr/lib/apache/mod_expires.so
/usr/lib/apache/mod_headers.so
/usr/lib/apache/mod_imap.so
/usr/lib/apache/mod_include.so
/usr/lib/apache/mod_info.so
/usr/lib/apache/mod_log_agent.so
/usr/lib/apache/mod_log_config.so
/usr/lib/apache/mod_log_referer.so
/usr/lib/apache/mod_mime.so

/usr/lib/apache/mod_mime_magic.so
/usr/lib/apache/mod_mmap_static.so
/usr/lib/apache/mod_negotiation.so
/usr/lib/apache/mod_rewrite.so
/usr/lib/apache/mod_setenvif.so
/usr/lib/apache/mod_speling.so
/usr/lib/apache/mod_status.so
/usr/lib/apache/mod_unique_id.so
/usr/lib/apache/mod_userdir.so
/usr/lib/apache/mod_usertrack.so
/usr/lib/apache/mod_vhost_alias.so
/usr/sbin/ab
/usr/sbin/httpd
/usr/sbin/logresolve
/usr/sbin/rotatelogs
/usr/sbin/suexec
/usr/share/doc/apache-1.3.20
/usr/share/doc/apache-1.3.20/ABOUT_APACHE
/usr/share/doc/apache-1.3.20/Announcement
/usr/share/doc/apache-1.3.20/KEYS
/usr/share/doc/apache-1.3.20/LICENSE
/usr/share/doc/apache-1.3.20/README
/usr/share/doc/apache-1.3.20/cgi-bin
/usr/share/doc/apache-1.3.20/cgi-bin/printenv
/usr/share/doc/apache-1.3.20/cgi-bin/test-cgi
/usr/share/man/man1/dbmmanage.1.gz
/usr/share/man/man1/htdigest.1.gz
/usr/share/man/man1/htpasswd.1.gz
/usr/share/man/man8/ab.8.gz
/usr/share/man/man8/httpd.8.gz
/usr/share/man/man8/logresolve.8.gz
/usr/share/man/man8/rotatelogs.8.gz
/usr/share/man/man8/suexec.8.gz
/var/cache/httpd

/var/log/httpd

/var/www

/var/www/cgi-bin

/var/www/html

/var/www/html/index.html

/var/www/html/poweredby.png

/var/www/icons

Chương 24:

CÀI ĐẶT LINUX FreeS/WAN VPN TRÊN Red Hat LINUX 7.2

Trong chương này chúng ta sẽ đề cập đến các vấn đề chính như sau:

- Biên dịch và đưa FreeS/WAN vào Kernel.
- Định lại cấu hình và cài đặt kernel để hỗ trợ FreeS/WAN.
- Biên dịch và cài đặt kernel mới với FreeS/WAN.
- Sự cấu hình cho FreeS/WAN.
- Yêu cầu cài đặt Network cho IPSec.
- Kiểm tra sự cài đặt.

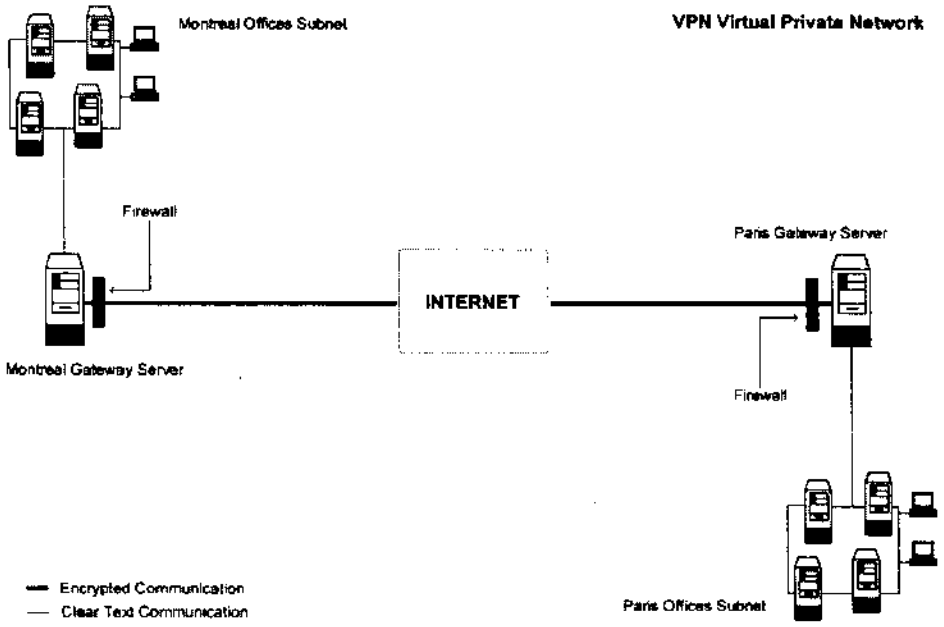
Tổng quan

Như đã trình bày trong phần biên dịch và cài đặt FreeSWAN trong Red Hat Linux 6.2 chúng ta đã thấy được tính ưu việt của sự thực thi IPSec protocol trong FreeS/WAN, trong việc thiết lập kênh trao đổi thông tin bảo mật, đảm bảo đầy đủ sự bí mật, tính xác thực, và tính nguyên vẹn của dữ liệu giữa hai bức tường lửa (firewall) trên Internet.

Sự thực thi IPSec trong phiên bản FreesWan1.94 là một sự cải tiến đáng kể về mặt biên dịch so với phiên bản Freeswan1.91 trên các version kernel khác nhau, các thuật toán được áp dụng trong việc tạo ra các khoá và việc quản lý khoá. Ví dụ như một tên kiến trúc trong KLIPS (kiến trúc net_device) đã được thay đổi để tránh những lỗi phát sinh đáng ngại khi biên dịch và xây dựng VPN trên kernel 2.2.x. Về những điểm cơ bản của một VPN/IPSec, cũng gống như FreesWan1.91, FreesWan1.94 cũng sử dụng mật khẩu phức tạp để cung cấp hai dịch vụ kiểm tra tính xác thực (authentication) và mã hoá (encryption). Dịch vụ Authentication bảo đảm rằng những gói dữ liệu (package) đến từ đúng người gởi và không bị thay đổi trong quá trình truyền tải. Dịch vụ mã hóa (encryption) ngăn cấm những người không có quyền đọc nội dung gói dữ liệu. IPSec có thể bảo vệ bất kỳ giao thức (protocol) nào đang chạy trên IP và bất kỳ sự trung gian sử dụng IP. IPSec cũng có thể cung cấp một vài dịch vụ bảo mật ở phía sau "background" không nhìn thấy được tác động tới người sử dụng.

Những dịch vụ IPSec cho phép bạn xây dựng những đường hầm (tunnel) bảo mật thông qua mạng không an toàn (Internet). Mọi thứ đi xuyên qua mạng không an toàn Internet đều được mã hoá bởi máy gateway IPSEC và được giải mã bởi gateway khác ở đầu cuối. Kết quả là Virtual Private Network hoặc VPN được thiết lập. Đây là một mạng máy tính, một mạng riêng cho dù nó bao gồm các máy tính ở những nơi khác nhau được kết nối bởi mạng không an toàn Internet.

Trong phần này chúng ta cũng giả sử xây dựng một VPN giữa hai mạng LAN cách xa nhau giống như kiến trúc mà bạn đã gặp trong phần trước nhưng trong phần này sẽ trình bày với Freeswan1.94 và trên Red Hat Linux 7.2.



Hướng dẫn cài đặt

- Các lệnh tương thích với Unix.
- Đường dẫn nguồn là "/usr/src".
- Sự cài đặt đã được kiểm tra trên Red Hat Linux 7.1 và 7.2.
- Tất cả các bước trong tiến trình cài đặt được thực thi bởi người sử dụng supper-user root.
- Version của Kernel là 2.4.7.
- Version của freeswan là 1.94.

Các gói dữ liệu

Kernel Homepage: http://www.kernelnotes.org/.

Bạn phải đảm bảo tải về: kernel-source-2.4.7-10.i386.rpm (có trong CD2 cài đặt RH Linux7.2) hoặc linux-2_4_7_tar.gz.

FreeS/WAN VPN Homepage Site: http://www.freeswan.org/.

FreeS/WAN VPN FTP Site: 194.109.6.26.

Bạn phải đảm bảo tải về: freeswan-1.94.tar.gz.

Dạng lưu trữ Tarballs

Để biết các tập tin nào phát sinh khi FreesWan được được cài đặt, trước hết hãy tạo danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt FreeS/WAN, sau đó so sánh chúng dùng lệnh '**diff**' để tìm ra các tập tin nào được đặt ở đâu. Đơn giản chạy lệnh '**find /* >** Freeswan1 ' trước và '**find /* >**

Freeswan2' sau khi bạn cài đặt phần mềm, sử dụng lệnh '**diff** Freeswan1 Freeswan2 > Freeswan-**Installed**' để lấy một danh sách các thay đổi nào đó.

Sự biên dịch của FreeS/WAN

Giải nén tập tin lưu trữ dạng tar (tar.gz)

```
[root@deep /]# cp freeswan-1.94.tar.gz  /usr/src/
[root@deep /]# cd /usr/src
[root@deep src]# tar xzf freeswan-1.94.tar
[root@deep src]# chown -R 0.0 /usr/src/freeswan-version
```

Biên dịch và đưa FreeS/WAN vào Kernel

Như đã nói ban đầu, Freeswan1.94 đã thay đổi mã nguồn và làm cho việc biên dịch và cài đặt FreesWan trở nên đơn giản hơn so với việc chúng ta làm trên RH Linux 6.2, vì thế chúng ta không cần phải thay đổi một số dòng trong các tập tin trong các thư mục con như "utils", "klips/utils", "Pluto" và "lib".

Chuyển vào thư mục gốc của FreeS/WAN và gõ vào những lệnh sau:

Bây giờ chúng ta sẽ biên dịch và cài đặt FreeSWAN trên server:

```
[root@deep freeswan-1.94]# ln –s linux-2.4.7-10 linux
[root@deep freeswan-1.94]# make insert
[root@deep freeswan-1.94]# make programs
[root@deep freeswan-1.94]# make install
```

Lệnh đầu tiên tạo một thư mục linux liên kết tới thư mục linux-2.4.7-10, đây là điểm khác so với linux 6.2, trong Linux7.2 thư mục chứa mã nguồn của kernel là linux-2.4.7-10. Bạn có thể không cần thực hiện lệnh này nếu bạn hiệu chỉnh tập tin Makefile.inc trong thư mục Freeswan1.94, để báo cho Freeswan biết thư mục chứa mã nguồn kernel là linux-2.4.7-10, tuy nhiên cách đơn giản nhất thì chúng ta làm như trên.

Lệnh "make insert" tạo một biểu tượng liên kết "/usr/src/linux/net/ipsec" trỏ tới thư mục nguồn KLIPS, nó ráp một vài tập tin kernel ở những nơi cần thiết, biết về KLIPS và/hoặc sửa lỗi. Nó cũng thêm cấu hình mặc định cho tập tin cấu hình kernel và cuối cùng nó tạo tập tin truyền đạt thông tin KLIPS "/dev/ipsec", nếu nó không sẵn có ở đây.

Lệnh "make programs" xây dựng các thư viện Pluto và những tiện ích mức user. Lệnh "make install" sẽ cài đặt chương trình deamon Pluto và những tiện ích user-level và thiết đặt những công việc cho quá trình startup.

Định lại cấu hình và cài đặt kernel để hỗ trợ FreeS/WAN

Trong phần cấu hình dưới tôi giả sử rằng RH Linux7.2 của bạn có thể làm việc được trong chế độ đồ hoạ (graphic). Bây giờ chúng ta quay trở lại thư mục "/usr/src/linux" và thực thi những lệnh sau để định lại cấu hình kernel với FreeS/WAN hỗ trợ:

```
[root@deep freeswan-1.94]# cd /usr/src/linux
[root@deep linux]# make mrproper
[root@deep linux]# make xconfig
```

Chú ý: *Khác với lệnh "make config" chúng ta đã sử dụng trong phần trước, lệnh "make xconfig" cũng tương tự như "make config" nhưng cho phép cấu hình lại kernel trong môi trường graphic và bây giờ phần mới liên quan tới FreeS/WAN được đưa vào trong cấu hình kernel của chúng ta. Do lý do này, phải định hình lại kernel để tùy chỉnh các tùy chọn IPSec của kernel của bạn.*

Điều đầu tiên cần làm là bảo đảm kernel của bạn được xây dựng với hỗ trợ FreeS/WAN. Trong kernel version 2.4.7, một phần mới liên quan tới freeswan VPN với tên là "IPSec options (FreeS/WAN)" sẽ xuất hiện trong cấu hình kernel của bạn sau khi bạn ráp kernel với chương trình FreeS/WAN như được diễn giải ở trên. Bạn cần bảo đảm rằng bạn đã trả lời Y cho những câu hỏi dưới đây (IPSec options (FreeS/WAN)).

```
IPSec options (FreeS/WAN)
IP Security Protocol (FreeS/WAN IPSEC) (CONFIG_IPSEC) [Y/n/?]
IPSEC: IP-in-IP encapsulation (CONFIG_IPSEC_IPIP) [Y/n/?]
IPSEC: Authentication Header (CONFIG_IPSEC_AH) [Y/n/?]
HMAC-MD5 authentication algorithm (CONFIG_IPSEC_AUTH_HMAC_MD5) [Y/n/?]
HMAC-SHA1 authentication algorithm (CONFIG_IPSEC_AUTH_HMAC_SHA1) [Y/n/?]
IPSEC: Encapsulating Security Payload (CONFIG_IPSEC_ESP) [Y/n/?]
3DES encryption algorithm (CONFIG_IPSEC_ENC_3DES) [Y/n/?]
IPSEC Debugging Option (DEBUG_IPSEC) [Y/n/?]
```

Chú ý: *Tất cả các tùy biến đã làm với kernel lần đầu tiên mà bạn đã chạy các lệnh "make config", "make dep" và "make clean" vẫn duy trì như vậy, không cần phải cấu hình lại mọi thứ trong kernel của bạn, chỉ phần mới được thêm FresS/WAN với tên "IPSec options (FreeS/WAN)" thì được yêu cầu như trình bày ở trên.*

Một vài tùy chọn về Network sẽ được mở một cách tự động ngay cả chúng đã bị tắc ở lần trước đó bởi vì IPSEC cần chúng. Bất cứ cấu hình nào bạn đang sử dụng, nên cẩn thận chú ý đến một vài vấn đề: theo kinh nghiệm cá nhân, bạn không nên làm mất tác dụng tùy chọn "Networking Options" của cấu hình kernel của bạn.

```
Kernel/User netlink socket (CONFIG_NETLINK) [Y/n/?]
Netlink device emulation (CONFIG_NETLINK_DEV) [Y/n/?]
```

Biên dịch và cài đặt kernel mới với FreeS/WAN

Chúng ta đã làm cho kernel của chúng ta hỗ trợ FreeS/WAN, bây giờ có thể biên dịch và cài đặt kernel mới.

- Quay trở lại thư mục "/usr/src/linux" và chạy những lệnh sau:

```
[root@deep linux]# make dep; make clean; make bzImage
```

- Biên dịch các module của kernel mới với lệnh:

```
[root@deep linux]# make modues
```

- Cài đặt các module của kernel mới với lệnh:

```
[root@deep linux]# make modules_install
```

Sau khi thực thi các lệnh ở trên, đi theo phần còn lại của sự chỉ dẫn trong phần nhân Linux (Linux Kernel) của quyển sách này (Chương cấu hình và xây dựng một kernel bảo mật, tối ưu) như để cài đặt kernel. Ở thời điểm này giả sử bạn đã copy và cài đặt tập tin kernel mới (bzImage), system.map, hoặc các module (nếu cần thiết) của bạn tới vị trí thích hợp và hiệu chỉnh tập tin lilo.conf để làm việc với kernel mới, bạn cũng cần chỉnh sửa các tập tin cấu hình có liên quan tới FreeS/WAN như "ipsec.conf" và "ipsec.secrets" trước khi khởi động lại hệ thống của bạn.

Dọn dẹp sau khi cài đặt

```
[root@deep /]# cd /usr/src
[root@deep src]# rm -rf freeswan-1.94/ freeswan-1.94.tar.gz
```

Lệnh "rm" sẽ loại bỏ tất cả các tập tin nguồn chúng ta đã sử dụng để biên dịch và cài đặt FreeS/WAN. Nó cũng sẽ loại bỏ tập tin nén của FreeS/WAN từ thư mục "/usr/src".

Sự cấu hình cho Freeswan

Cấu hình tập tin "/etc/ipsec.conf"

Tập tin cấu hình "/etc/ipsec.conf" của FreeS/WAN cho phép bạn cấu hình IPSEC, điều khiển thông tin và các kiểu kết nối. Hiện tại IPSec hỗ trợ hai kiểu kết nối: Khoá điều khiển bằng tay (Manually keyed) và khoá tự động (Automatically keyed). Phương cách làm việc của hai kiểu kết nối này hoàn toàn khác nhau, sự kết nối Manually keyed sử dụng những khoá được lưu trữ trong tập tin "/etc/ipsec.conf", kiểu của sự kết nối này bảo mật kém hơn Automatically keyed. Sự kết nối Automatically keyed sử dụng những khoá được sinh ra một cách tự động bởi chương trình thoả thuận khoá Pluto. Giao thức thoả thuận khoá được sử dụng mặc định với tên là IKE, kiểm tra tính xác thực với hệ thống khác sử dụng một khoá riêng biệt được lưu trữ trong tập tin "/etc/ipsec.secrets". Với lý do này tôi muốn thông báo rằng, bạn nên sử dụng sự kết nối Automatically keyed sẽ có nhiều tính năng bảo mật hơn Manually keyed. Và đây cũng là phương cách được sử dụng nhiều trong việc thiết lập VPN.

Trong ví dụ về cấu hình dưới đây chúng ta sẽ cấu hình một tunnel xuyên qua hai firewall được thiết lập giữa hai đầu bên trái và bên phải. Tôi chọn trình bày cấu hình này cho các bạn vì tôi cho rằng đây là cấu hình mà nhiều người và các công ty sẽ sử dụng, nó cũng cho phép chúng ta có nhiều tùy chọn trong tập tin cấu hình "ipsec.conf" cho các kết nối automatically keyed. Các ví dụ cấu hình khác nhau cũng có sẵn và bạn cũng có thể tham khảo thư mục "doc/examples" trong thư mục nguồn Frees/WAN.

```
[root@deep freeswan-1.94]# cd /usr/src/linux
[root@deep linux]# make mrproper
[root@deep linux]# make xconfig
```

Chú ý: *Khác với lệnh "make config" chúng ta đã sử dụng trong phần trước, lệnh "make xconfig" cũng tương tự như "make config" nhưng cho phép cấu hình lại kernel trong môi trường graphic và bây giờ phần mới liên quan tới FreeS/WAN được đưa vào trong cấu hình kernel của chúng ta. Do lý do này, phải định hình lại kernel để tùy chỉnh các tùy chọn IPSec của kernel của bạn.*

Điều đầu tiên cần làm là bảo đảm kernel của bạn được xây dựng với hỗ trợ FreeS/WAN. Trong kernel version 2.4.7, một phần mới liên quan tới freeswan VPN với tên là "IPSec options (FreeS/WAN)" sẽ xuất hiện trong cấu hình kernel của bạn sau khi bạn ráp kernel với chương trình FreeS/WAN như được diễn giải ở trên. Bạn cần bảo đảm rằng bạn đã trả lời Y cho những câu hỏi dưới đây (IPSec options (FreeS/WAN)).

```
IPSec options (FreeS/WAN)
IP Security Protocol (FreeS/WAN IPSEC) (CONFIG_IPSEC) [Y/n/?]
IPSEC: IP-in-IP encapsulation (CONFIG_IPSEC_IPIP) [Y/n/?]
IPSEC: Authentication Header (CONFIG_IPSEC_AH) [Y/n/?]
HMAC-MD5 authentication algorithm (CONFIG_IPSEC_AUTH_HMAC_MD5) [Y/n/?]
HMAC-SHA1 authentication algorithm (CONFIG_IPSEC_AUTH_HMAC_SHA1) [Y/n/?]
IPSEC: Encapsulating Security Payload (CONFIG_IPSEC_ESP) [Y/n/?]
3DES encryption algorithm (CONFIG_IPSEC_ENC_3DES) [Y/n/?]
IPSEC Debugging Option (DEBUG_IPSEC) [Y/n/?]
```

Chú ý: *Tất cả các tùy biến đã làm với kernel lần đầu tiên mà bạn đã chạy các lệnh "make config", "make dep" và "make clean" vẫn duy trì như vậy, không cần phải cấu hình lại mọi thứ trong kernel của bạn, chỉ phần mới được thêm FresS/WAN với tên "IPSec options (FreeS/WAN)" thì được yêu cầu như trình bày ở trên.*

Một vài tùy chọn về Network sẽ được mở một cách tự động ngay cả chúng đã bị tắc ở lần trước đó bởi vì IPSEC cần chúng. Bất cứ cấu hình nào bạn đang sử dụng, nên cẩn thận chú ý đến một vài vấn đề: theo kinh nghiệm cá nhân, bạn không nên làm mất tác dụng tùy chọn "Networking Options" của cấu hình kernel của bạn.

```
Kernel/User netlink socket (CONFIG_NETLINK) [Y/n/?]
Netlink device emulation (CONFIG_NETLINK_DEV) [Y/n/?]
```

Biên dịch và cài đặt kernel mới với FreeS/WAN

Chúng ta đã làm cho kernel của chúng ta hỗ trợ FreeS/WAN, bây giờ có thể biên dịch và cài đặt kernel mới.

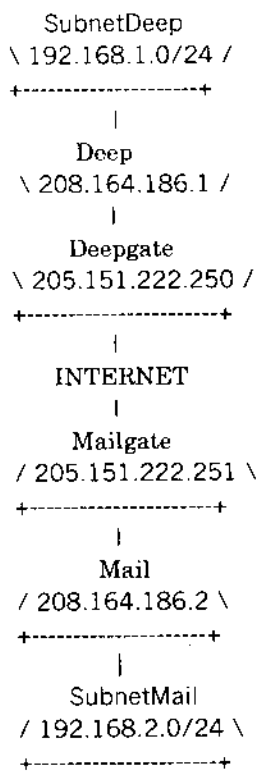
- Quay trở lại thư mục "/usr/src/linux" và chạy những lệnh sau:

```
[root@deep linux]# make dep; make clean; make bzImage
```

SubnetDeep=====Deep-----Deepgate.....Mailgate-----
Mail=====SubnetMail

Mạng không an toàn (Internet)

leftsubnet = **SubnetDeep** (192.168.1.0/24)
left = **Deep** (deep.openna.com)
leftnexthop = **Deepgate** (the first router in the direction or ISP router for deep.openna.com)
Internet = **Untrusted net**
rightnexthop = **Mailgate** (the first router in the direction or ISP router for mail.openna.com)
right = **Mail** (mail.openna.com)
rightsubnet = **SubnetMail** (192.168.2.0/24)



SubnetDeep là địa chỉ mạng IP mạng riêng của bạn trên gateway thứ nhất, eth1 được gắn bên trong network.

Deep là địa chỉ IP của gateway thứ nhất, eth0 được gắn vào Internet.

Deepgate là địa chỉ IP của Router thứ nhất theo hướng của gatewate thứ hai của bạn (mail.openna.com hoặc router ISP của bạn.

Internet là mạng không an toàn.

Mailgate là địa chỉ IP của Router thứ hai theo hướng của gatewate thứ nhất của bạn (deep.openna.com) hoặc router ISP của bạn.

Mail là địa chỉ IP của gateway thứ hai của bạn, eth0 được vào Internet.

SubnetMail là địa chỉ mạng IP mạng riêng của bạn trên gateway thứ hai, eth1 được gắn bên trong mạng.

Chúng ta phải hiệu chỉnh tập tin ipsec.conf (vi /etc/ipsec.conf) và thay đổi những giá trị mặc định để định rõ sự trao đổi thông tin và cấu hình IPSec của chúng ta. Hiện tại chúng ta có các kiểu (type) của section trong tập tin (/etc/ipsec.conf): phần "config" chỉ rõ thông tin cấu hình chung cho IPSec và phần "conn" chỉ rõ sự kết nối IPSec. Nội dung của chúng phải tuyệt đối bảo mật trừ khi

manual keying được dùng (nhắc lại, manual keying không nên dùng cho thông tin cần bảo mật).

Section đầu tiên có tên *config setup* chứa đựng toàn bộ những tham số cài đặt cho IPSEC mà sẽ được áp dụng cho tất cả các connection và thông tin được dùng khi software khởi động.

Section thứ hai có tên conn chứa đựng một sự chỉ định kết nối mạng dùng IPSec. Tên chúng ta đặt cho nó là tùy ý.

Xem ví dụ sau:

```
# /etc/ipsec.conf - FreeS/WAN IPSEC configuration file

# More elaborate and more varied sample configurations can be found
# in doc/examples.

# basic configuration
config setup
        interfaces="ipsec0=eth0"
        klipsdebug=none
        plutodebug=none
        plutoload=%search
        plutostart=%search

# sample connection
conn deep-mail
        left=208.164.186.1
        leftsubnet=192.168.1.0/24
        leftnexthop=205.151.222.250
        right=208.164.186.2
        rightsubnet=192.168.2.0/24
        rightnexthop=205.151.222.251
        keyingtries=0
        auth=ah
        auto=start
```

Trong phần này tôi không có ý định diễn giải các thông số trong tập tin cấu hình trên. Bạn có thể tham khảo các thông tin này trong phần cấu hình Freeswan cho Linux6.2 hoặc có thể với lệnh "man ipsec.conf".

Conn deep-mail

Tùy chọn này chỉ rõ tên đưa ra để nhận biết connection nào dùng IPSec. Bạn nên có một qui ước trong việc đánh tên để tránh sự nhầm lẫn. Ví dụ liên kết

giữa hai gateway server deep.openna.com và mail.openna.com có thể được đặt tên là deep-mail hoặc liên kết giữa các văn phòng ở Saigon và Hanoi có thể đặt tên là saigon-hanoi.

Chú ý rằng tên deep-mail hoặc bất cứ tên nào bạn chọn nên giống nhau trong tập tin ipsec.conf trên cả hai gateway, nếu có thay đổi sẽ làm trong tập tin /etc/ipsec.conf trên gateway thứ hai là thay đổi dòng "interface=" cho phù hợp với interface mà gateway thứ hai sử dụng cho sự kết nối IPSEC, tất nhiên bạn sẽ không cần làm gì hết nếu không có sự khác biệt nào. Ví dụ nếu interface eth0 được sử dụng trên cả hai gateway cho sự trao đổi thông tin IPSEC, bạn không cần thay đổi dòng interface= trên gateway thứ hai. Trong trường hợp nếu gateway thứ nhất dùng eth0 và gateway thứ hai sử dụng eth1, bạn phải thay đổi dòng "interface=" trên gateway thứ hai cho phù hợp với interface eth1.

Cấu hình tập tin "/etc/ipsec.secrets"

Tập tin ipsec.secrets lưu trữ khóa bảo mật được sử dụng bởi chương trình pluto để kiểm tra tính xác thực giữa hai gateway. Có hai loại khóa bảo mật có thể được cấu hình trong tập tin này là khóa bảo mật được chia sẻ (preshared secrets key) và khóa riêng RSA (RSA private keys). Bạn phải kiểm tra chế độ (mode) và quyền (permission) của tập tin này để đảm bảo rằng super-user "root" là sở hữu của tập tin này và permission trên tập tin này bị khóa không cho truy cập bởi những user khác.

Bước 1:

Mặc định có một ví dụ về secret key được cung cấp trong tập tin ipsec.secrets khi bạn triển khai IPSec, bạn nên thay đổi nó bằng cách tạo ra secret key cho chính bạn. Ipsec có thể hỗ trợ (support) cho bạn tạo ra khóa 256bit và dùng khóa này trong suốt quá trình trao đổi khóa để đảm bảo rằng các tin tặc không thể tìm thấy trong thời gian ngắn.

- Để tạo một shared secret key mới dùng những sau:

```
[root@deep /]# ipsec ranbits 256 > temp
```

Khóa mới được tạo với trình tiện ích ranbits(8) và được lưu trong tập tin temp. Khóa mới được tạo sau một vài giây và nó được chọn ngẫu nhiên 256bit.

Chú ý: *Đừng quên xóa tập tin temp sau khi bạn đã làm xong với nó.*

Bước 2:

Bây giờ bạn đã có khóa shared secret mới được lưu trong tập tin temp, bạn phải đặt nó trong tập tin "/etc/ipsec.secrets", khi hiệu chỉnh tập tin ipsec.secrets bạn sẽ thấy một vài điều giống như xuất hiện dưới đây trên màn hình của bạn. Mỗi dòng có địa chỉ IP của cả hai gateway cộng với khóa bảo mật. Nó trông giống như trình bày dưới đây:

```
# This file holds shared secrets which are currently the only inter-Pluto
```

authentication mechanism. See ipsec_pluto(8) manpage. Each secret is
(oversimplifying slightly) for one pair of negotiating hosts.

The shared secrets are arbitrary character strings and should be both
long and hard to guess.

Note that all secrets must now be enclosed in quotes, even if they have
no white space inside them.

10.0.0.1 11.0.0.1 **"jxVS1kVUTTuIkVRRTnTujSm444jRuU1mIkkIku2nkW3nnVu V2WjjRRnuImIkmU1Run5VSnnRT"**

Hiệu chỉnh tập tin ipsec.secrets (vi /etc/ipsec.secrets) và thay đổi khóa .

10.0.0.1 11.0.0.1 " jxVS1kVUTTuIkVRRTnTujSm444jRuU1mIkkIku2nkW3nnVu V2WjjRRnuImIkmU1Run5VSnnRT "

Trở thành:
208.164.186.1 208.164.186.2
"0x9748cc31_2e99194f_d230589b_cd846b57_dc070b01_74b66f34_19c40a1a_804906ed"

Ở đây "208.164.186.1" và "208.164.186.2" là địa chỉ IP của hai gateway và

"0x9748cc31_2e99194f_d230589b_cd846b57_dc070b01_74b66f34_19c40a1a_804906ed" (Chú ý rằng yêu cầu có dấu ngoặc) là khóa bảo mật mà chúng ta đã tạo ra ở trên với lệnh "ipsec ranbits 256 > temp" trong tập tin temp.

Bước 3:

Cả hai tập tin "ipsec.conf" và "ipsec.secrets" phải được copy tới máy gateway thứ hai để đảm bảo tính giống hệt trên cả hai, chỉ ngoại trừ tập tin ipsec.conf ở dòng config setup cần chỉnh sửa chính xác interface cài đặt cho gateway thứ hai nếu chúng khác so với gateway thứ nhất. Khác với RSA private key, secret key có cùng shared key trên cả hai gateway.

Chú ý: *File "/etc/ipsec.secrets" sẽ có permission rw-------(600) và được sở hữu bởi super-user root. tập tin "/etc/ipsec.conf được cài đặt với quyền rw-r—r--(644) và cũng được sở hữu bởi root.*

Cấu hình RSA private keys secrets

Xin nhắc lại rằng hiện tại FreesWAN hỗ trợ hai loại khóa là: preshares secrets key và RSA key. Về preshared secrets là những gì chúng ta đã cấu hình trong các tập tin ipsec.conf và ipsec.secrets ở trên. Một vài người thích dùng RSA private key cho việc kiểm tra tính xác thực bằng chương trình Pluto daemon của những host khác. Nếu bạn ở trong trường hợp này, bạn sẽ phải làm một vài sự thay đổi nhỏ trên cả hai tập tin ipsec.conf và ipsec.secrets, như được diễn giải trong những bước dưới đây:

Bạn cần tạo một RSA riêng lẻ cho mỗi gateway, mỗi gateway đặt khóa riêng (private key) của nó trong chính tập tin ipsec.secrets của gateway đó và khóa công cộng (public key) sẽ được đặt trong các tham số "leftrsasigkey" và "rightrsasigkey" trong phần conn của tập tin "ipsec.conf". Các bước tiến hành như sau:

Bước 1:

Tạo một khóa RSA riêng lẻ cho mỗi gateway:

- Trên gateway thứ nhất (ví dụ deep) sử dụng những lệnh sau:

```
[root@deep /]# cd /
[root@deep /]# ipsec rsasigkey --verbose 1024 > deep-keys
computing primes and modulus...
getting 64 random bytes from /dev/random
looking for a prime starting there
found it after 30 tries
getting 64 random bytes from /dev/random
looking for a prime starting there
found it after 230 tries
swapping primes so p is the larger
computing (p-1)*(q-1)...
computing d...
computing exp1, exp1, coeff...
output...
```

- Trên gateway thứ hai (ví dụ mail) dùng những lệnh sau:

```
[root@mail /]# cd /
[root@mail /]# ipsec rsasigkey --verbose 1024 > mail-keys
computing primes and modulus...
getting 64 random bytes from /dev/random
looking for a prime starting there
found it after 30 tries
getting 64 random bytes from /dev/random
looking for a prime starting there
found it after 230 tries
swapping primes so p is the larger
compu ting (p-1)*(q-1)...
computing d...
computing exp1, exp1, coeff...
output...
```

Trình tiện ích rsasigkey sinh ra một cặp khóa private và public key với một chữ ký 1024 bit và đặt nó trong tập tin deep-keys (mail-keys cho lệnh thứ hai trên gateway thứ hai). Private key có thể được đưa vào đúng nguyên văn trong tập tin "ipsec.secrets" và public key được đưa vào trong tập tin "ipsec.conf".

Chú ý: *Các tập tin deep-keys và mail-keys nên được xóa sau khi đã làm với nó.*

Bước 2:

Thay đổi tập tin /etc/ipsec.conf của bạn để dùng RSA public keys trong mỗi gateway:

Hiệu chỉnh tập tin ipsec.conf nguyên thủy của bạn (vi /etc/ipsec.conf) và thêm vào những tham số liên quan đến RSA trong phần conn của tập tin ipsec.conf trên cả hai gateway:

```
# sample connection
conn deep-mail
left=208.164.186.1
leftsubnet=192.168.1.0/24
leftnexthop=205.151.222.250
right=208.164.186.2
rightsubnet=192.168.2.0/24
rightnexthop=205.151.222.251
keyingtries=0
auth=ah
authby=rsasig
leftrsasigkey=<Public key of deep>
rightrsasigkey=<Public key of mail>
auto=start
```

Tôi cũng không có ý định giải thích những tham số trong cấu hình trên, bạn có thể tham khảo ở phần trước.

RAS keys cho gateway deep (deep-keys)

```
[root@deep /]# cd /
[root@deep /]# vi deep-keys

# 1024 bits, Fri Feb 4 05:05:19 2000
# for signatures only, UNSAFE FOR ENCRYPTION
#pubkey=0x010395daee1be05f3038ae529ef2668afd79f5ff1b16203c9ce
aef801cea9cb74bcfb51a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b
5427e04cf4a38ef98a7f29edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f08
4186a2a54f4b4cec12cca1a5deac3b19d561c16a76bab772888f1fd71aa08f0850
2a141b611f
Modulus:
0x95daee1be05f3038ae529ef2668afd79f5ff1b16203c9ceaef801cea9cb74bcfb51
a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b5427e04cf4a38ef98a7f2
9edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f084186a2a54f4b4cec12cca
1a5deac3b19d561c16a76bab772888f1fd71aa08f08502a141b611f
PublicExponent: 0x03
# everything after this point is secret
```

PrivateExponent:
0x63e74967eaea2025c98c69f6ef0753a6a3ff6764157dbdf1f50013471324dd352
366f48805b0b37f232384b2b52ce2ee85d173468b62eaa052381a9588a317b3a1
324d01a531a41fa7add6c5efbdd88f4718feed2bc0246be924e81bb90f03e49ceedf
7af0dd48f06f265b519600bd082c6e6bd27eaa71cc0288df1ecc3b062b
Prime1:
0xc5b471a88b025dd09d4bd7b61840f20d182d9b75bb7c11eb4bd78312209e3a
ee7ebfe632304db6df5e211d21af7fee79c5d45546bea3ccc7b744254f6f0b847f
Prime2:
0xc20a99feeafe79767122409b693be75f15e1aef76d098ab12579624aec708e85e
2c5dd62080c3a64363f2f45b0e96cb4aef8918ca333a326d3f6dc2c72b75361
Exponent1:
0x83cda11b0756e935be328fcebad5f6b36573bcf927a80bf2328facb6c0697c9eff
2a9976cade79ea3ec0be1674fff4512e8d8e2f29c2888524d818df9f5d02ff
Exponent2:
0x815c66a9f1fefba44b6c2b124627ef94b9411f4f9e065c7618fb96dc9da05f03ec
83e8ec055d7c42ced4ca2e75f0f3231f5061086ccd176f37f9e81da1cf8ceb
Coefficient:
0x10d954c9e2b8d11f4db1b233ef37ff0a3cecfffad89ba5d515449b007803f577e3
bd7f0183ceddfd805466d62f767f3f5a5731a73875d30186520f1753a7e325

RSA keys cho gateway mail (mail-keys):

```
[root@mail /]# cd /
[root@mail /]# vi mail-keys

# 1024 bits, Fri Feb 4 04:46:59 2000
# for signatures only, UNSAFE FOR ENCRYPTION
#pubkey=0x01037631b81f00d5e6f888c542d44dbb784cd3646f084ed96
f942d341c7c4686cbd405b805dc728f8697475f11e8b1dd797550153a3f0d4ff0f2
b274b70a2ebc88f073748d1c1c8821dc6be6a2f0064f3be7f8e4549f8ab9af64944
f829b014788dd202cf7d2e320cab666f5e7a197e64efe0bfee94e92ce4dad82d52
30c57b89edf
Modulus:
0x7631b81f00d5e6f888c542d44dbb784cd3646f084ed96f942d341c7c4686cbd4
05b805dc728f8697475f11e8b1dd797550153a3f0d4ff0f2b274b70a2ebc88f0737
48d1c1c8821dc6be6a2f0064f3be7f8e4549f8ab9af64944f829b014788dd202cf7
d2e320cab666f5e7a197e64efe0bfee94e92ce4dad82d5230c57b89edf
PublicExponent: 0x03
# everything after this point is secret
PrivateExponent:
0x4ecbd014ab3944a5b08381e2de7cfadde242f4b03490f50d737812fd8459dd38
03d003e84c5faf0f84ea0bf07693a64e35637c2a08dff5f721a324b1747db09f62c8
71d5e11711251b845ae76753d4ef967c494b0def4f5d0762f65da603bc04c41b4c
6cab4c413a72c633b608267ae2889c162a3d5bc07ee083b1c6e038400b
Prime1:
0xc7f7cc8feaaac65039c39333b878bffd8f95b0dc22995c553402a5b287f341012
253e9f25b83983c936f6ca512926bebee3d5403bf9f4557206c6bbfd9aac899
Prime2:
```

0x975015cb603ac1d488dc876132d8bc83079435d2d3395c03d5386b5c004ead
d4d7b01b3d86aad0a2275d2d6b791a2abe50d7740b772567981 1a32ca22db976
37

Exponent1:
0x854fddb5471c84357bd7b777d0507ffe5fb92092c1bb92e37801c3cc5aa22b56
16e29bf6e7ad1028624a486e0c619d47f428e2ad2a6a2e3a159d9d2a911c85bb

Exponent2:
0x64e00e87957c81385b3daf9621e5d302050d7937377b92ad38d04792aadf1e8
de52012290471e06c1a3e1e47a61171d435e4f807a4c39a6561177316c9264ecf

Coefficient:
0x6f087591becddc210c2ee0480e30beeb25615a3615203cd3cef65e5a1d476fd9
602ca0ef10d9b858edb22db42c975fb71883a470b43433a7be57df7ace4a0a3f

Định rõ và copy các public key của deep và mail trên cả hai tập tin deep-keys và mail-keys vào tập tin ipsec.conf như được trình bày ở dưới đây. Bạn có thể xác định dòng liên quan tới public key bởi một câu bắt đầu với dòng "#pubkey="

```
# sample connection
conn deep-mail
        left=208.164.186.1
        leftsubnet=192.168.1.0/24
        leftnexthop=205.151.222.250
        right=208.164.186.2
        rightsubnet=192.168.2.0/24
        rightnexthop=205.151.222.251
        keyingtries=0
        auth=ah
        authby=rsasig

        leftrsasigkey=0x010395daee1be05f3038ae529ef2668afd79f5ff1b16203c9ceaef80
        1cea9cb74bcfb51a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b5427e0
        4cf4a38ef98a7f29edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f084186a2a5
        4f4b4cec12cca1a5deac3b19d561c16a76bab772888f1fd71aa08f08502a141b611f

        rightrsasigkey=0x01037631b81f00d5e6f888c542d44dbb784cd3646f084ed96f942
        d341c7c4686cbd405b805dc728f8697475f11e8b1dd797550153a3f0d4ff0f2b274b
        70a2ebc88f073748d1c1c8821dc6be6a2f0064f3be7f8e4549f8ab9af64944f829b01
        4788dd202cf7d2e320cab666f5e7a197e64efe0bfee94e92ce4dad82d5230c57b89
        edf
        auto=start
```

Chú ý: *Xin nhắc lại rằng trong ví dụ trên tham số "leftrsasigkey=" chứa đựng public key của deep và "rightrsasigkey=" chứa đựng public key của mail.*

Bước 3:

Thay đổi tập tin "/etc/ipsec.secrets" để dùng private key trong mỗi gateway:

Hiệu chỉnh tập tin ipsec.secrets ban đầu (/vi /etc/ipsec.secrets) và thêm RSA private key cho việc kiểm tra tính xác thực trên cả hai gateway.

File ipsec.secrets cho gateway deep:

[root@deep /]# vi /etc/ipsec.secrets

208.164.186.1 208.164.186.2
"0x9748cc31_2e99194f_d230589b_cd846b57_dc070b01_74b66f34_19c40a1a_804906ed"

Bạn phải thay đổi tập tin ipsec.secrets của bạn trông giống như được trình bày ở trên cho cả hai gateway. Có một điều quan trọng là private key thì không giống nhau cho hai gateway deep và mail. Private key cho deep được lấy từ tập tin RSA key "deep-keys" trong khi private key cho mail được lấy từ tập tin RSA key "mail-keys".

```
208.164.186.1 208.164.186.2: RSA {
        Modulus:
0x95daee1be05f3038ae529ef2668afd79f5ff1b16203c9ceaef801cea9cb74bcfb51
a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b5427e04cf4a38ef98a7f2
9edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f084186a2a54f4b4cec12cca
1a5deac3b19d561c16a76bab772888f1fd71aa08f08502a141b611f
        PublicExponent: 0x03
        # everything after this point is secret
        PrivateExponent:
0x63e74967eaea2025c98c69f6ef0753a6a3ff6764157dbdf1f50013471324dd352
366f48805b0b37f232384b2b52ce2ee85d173468b62eaa052381a9588a317b3a1
324d01a531a41fa7add6c5efbdd88f4718feed2bc0246be924e81bb90f03e49ceedf
7af0dd48f06f265b519600bd082c6e6bd27eaa71cc0288df1ecc3b062b
        Prime1:
0xc5b471a88b025dd09d4bd7b61840f20d182d9b75bb7c11eb4bd78312209e3a
ee7ebfe632304db6df5e211d21af7fee79c5d45546bea3ccc7b744254f6f0b847f
        Prime2:
0xc20a99feeafe79767122409b693be75f15e1aef76d098ab12579624aec708e85e
2c5dd62080c3a64363f2f45b0e96cb4aef8918ca333a326d3f6dc2c72b75361
        Exponent1:
0x83cda11b0756e935be328fcebad5f6b36573bcf927a80bf2328facb6c0697c9eff
2a9976cade79ea3ec0be1674fff4512e8d8e2f29c2888524d818df9f5d02ff
        Exponent2:
0x815c66a9f1fefba44b6c2b124627ef94b9411f4f9e065c7618fb96dc9da05f03ec
83e8ec055d7c42ced4ca2e75f0f3231f5061086ccd176f37f9e81da1cf8ceb
        Coefficient:
0x10d954c9e2b8d11f4db1b233ef37ff0a3cecfffad89ba5d515449b007803f577e3
bd7f0183ceddfd805466d62f767f3f5a5731a73875d30186520f1753a7e325
        }
```

Tập tin "ipsec.secrets" cho gateway mail

[root@mail /]# vi /etc/ipsec.secrets

```
208.164.186.1 208.164.186.2: RSA {
        Modulus:
```

0x95daee1be05f3038ae529ef2668afd79f5ff1b16203c9ceaef801cea9cb74bcfb51
a6ecc08890d3eb4b5470c0fc35465c8ba2ce9d1145ff07b5427e04cf4a38ef98a7f2
9edcb4d7689f2da7a69199e4318b4c8d0ea25d33e4f084186a2a54f4b4cec12cca
1a5deac3b19d561c16a76bab772888f1fd71aa08f08502a141b611f
PublicExponent: 0x03
everything after this point is secret
PrivateExponent:
0x63e74967eaea2025c98c69f6ef0753a6a3ff6764157dbdf1f50013471324dd352
366f48805b0b37f232384b2b52ce2ee85d173468b62eaa052381a9588a317b3a1
324d01a531a41fa7add6c5efbdd88f4718feed2bc0246be924e81bb90f03e49ceedf
7af0dd48f06f265b519600bd082c6e6bd27eaa71cc0288df1ecc3b062b
Prime1:
0xc5b471a88b025dd09d4bd7b61840f20d182d9b75bb7c11eb4bd78312209e3a
ee7ebfe632304db6df5e211d21af7fee79c5d45546bea3ccc7b744254f6f0b847f
Prime2:
0xc20a99feeafe79767122409b693be75f15e1aef76d098ab12579624aec708e85e
2c5dd62080c3a64363f2f45b0e96cb4aef8918ca333a326d3f6dc2c72b75361
Exponent1:
0x83cda11b0756e935be328fcebad5f6b36573bcf927a80bf2328facb6c0697c9eff
2a9976cade79ea3ec0be1674fff4512e8d8e2f29c2888524d818df9f5d02ff
Exponent2:
0x815c66a9f1fefba44b6c2b124627ef94b9411f4f9e065c7618fb96dc9da05f03ec
83e8ec055d7c42ced4ca2e75f0f3231f5061086ccd176f37f9e81da1cf8ceb
Coefficient:
0x10d954c9e2b8d11f4db1b233ef37ff0a3cecfffad89ba5d515449b007803f577e3
bd7f0183ceddfd805466d62f767f3f5a5731a73875d30186520f1753a7e325
}

Kiểm tra tính xác thực bằng chữ ký RSA yêu cầu mỗi host có private key của chính nó. Phần khóa có thể bắt đầu với một dấu hiệu chỉ rõ loại của khóa. "RAS" biểu hiện cho RSA private key và "PSK" (mặc định) biểu hiện cho PreShared Key. Vì "PSK" được dùng mặc định bạn phải chỉ rõ "RSA", như vậy chúng ta có thể sử dụng RSA private key trong tập tin này (ipsec.secrets). Super-user "root" là người chủ sở hữu tập tin này và permission sẽ bị khóa cho việc truy cập của các user khác.

Yêu cầu cài đặt Network cho IPSec

Có một vài sự nghiên cứu bạn phải bảo đảm là chính xác trước khi chạy phần mềm FreeS/WAN. Những nghiên cứu này rất quan trọng nếu bạn không muốn nhận messages lỗi trong quá trình khởi động (startup) VPN của bạn. Theo sau là những bước cần tiến hành.

Bước 1

Bạn sẽ cần làm cho chức năng TCP/IP forwarding có tác dụng trên cả hai gateway server. Trong Red Hat Linux, việc này thực hiện bằng cách thay đổi hoặc thêm những dòng sau, phụ thuộc trên version Red Hat bạn đang sử dụng:

Dưới Red Hat Linux 7.2

- Hiệu chỉnh tập tin network (vi /proc/sys/net/ipv4/ip_forward) và thay đổi giá trị:

  0
  Trở thành:
  1

- Bạn phải khởi động lại network của bạn để việc thay đổi có hiệu lực:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Bringing up interface lo [ OK ]
Bringing up interface eth0 [ OK ]
Bringing up interface eth1 [ OK ]
```

Bước 2:

Nhắc lại rằng các kết nối automatically keyed sử dụng các khoá được sinh ra một cách tự động bởi chương trình Pluto key negotiation daemon. Chương trình pluto daemon khi khởi động sẽ thử connect tới chương trình Pluto daemon khác ở cuối tunnel và thiết lập một connection. Cho nguyên nhân này một gateway IPSEC nên có các qui tắc lọc package (trong tập tin script firewall) cho phép những protocol sau đi qua gateway khi nói chuyện với IPSEC gateway khác.

- UDP port 500 cho IKE được thực thi bởi Pluto daemon.
- Protocol 50 cho ESP mã hoá và kiểm tra tính xác thực.
- Protocol 51 cho việc kiểm tra tính xác thực mức package.

Hiệu chỉnh tập tin script firewall (vi /etc/rc.d/init.d/firewall) trên cả hai máy gateway và thêm vào những dòng sau cho phép các package IPSEC đi qua gateway từ xa vào gateway của bạn.

```
# ++++++++++++++++++++++++
# + FreeS/WAN IPSec VPN +
# ++++++++++++++++++++++++

# if you are using the FreeSWAN IPSec VPN, you will need to
# fill in the addresses of the gateway in the IPSECSG and the virtual
# interfaces for FreeS/Wan IPSec in the FREESWANVI parametters. Look at
# the beginning of this firewall script rules taäp tin to set the parameters.

# IPSECSG is a space separated list of remote gateways.
```

```
# FREESWANVI is a space seprated list of virtual interfaces
# for FreeS/WAN IPSec
# implementation. Only include those that are actually used.

# Allow IPSec protocol from remote gateway on external
# interface IPSec
# IPSec uses three main types of packet:
# IKE uses the UDP protocol and port 500
# ESP use the protocol number 50, and
# AH use the protocol 51

/sbin/depmod –a
echo "1" > /proc/sys/net/ipv4/ip_forward

# Set default build-in chains, default is DROP packets.
iptables -F INPUT
iptables -P INPUT DROP
iptables -F FORWARD
iptables -P FORWARD DROP
iptables -F OUTPUT
iptables -P OUTPUT DROP

EXTERNAL_INTERFACE="eth0"
INTERNAL_INTERFACE="eth1"
ANYWHERE="any/0"

IPSECSG=208.164.186.2
IPSECLC=208.164.186.1

FREESWANVI="ipsec0"
LOCALNET_1="192.168.1.0/24"
LOCALNET_2="192.168.2.0/24"
```

Dưới Red Hat Linux 7.2

- Hiệu chỉnh tập tin network (vi /proc/sys/net/ipv4/ip_forward) và thay đổi giá trị:

  0
  Trở thành:
  1

- Bạn phải khởi động lại network của bạn để việc thay đổi có hiệu lực:

```
[root@deep /]# /etc/rc.d/init.d/network restart
Bringing up interface lo [ OK ]
Bringing up interface eth0 [ OK ]
Bringing up interface eth1 [ OK ]
```

Bước 2:

Nhắc lại rằng các kết nối automatically keyed sử dụng các khoá được sinh ra một cách tự động bởi chương trình Pluto key negotiation daemon. Chương trình pluto daemon khi khởi động sẽ thử connect tới chương trình Pluto daemon khác ở cuối tunnel và thiết lập một connection. Cho nguyên nhân này một gateway IPSEC nên có các qui tắc lọc package (trong tập tin script firewall) cho phép những protocol sau đi qua gateway khi nói chuyện với IPSEC gateway khác.

- UDP port 500 cho IKE được thực thi bởi Pluto daemon.
- Protocol 50 cho ESP mã hoá và kiểm tra tính xác thực.
- Protocol 51 cho việc kiểm tra tính xác thực mức package.

Hiệu chỉnh tập tin script firewall (vi /etc/rc.d/init.d/firewall) trên cả hai máy gateway và thêm vào những dòng sau cho phép các package IPSEC đi qua gateway từ xa vào gateway của bạn.

```
# ++++++++++++++++++++++++
# + FreeS/WAN IPSec VPN +
# ++++++++++++++++++++++++

# if you are using the FreeSWAN IPSec VPN, you will need to
# fill in the addresses of the gateway in the IPSECSG and the virtual
# interfaces for FreeS/Wan IPSec in the FREESWANVI parametters. Look at
# the beginning of this firewall script rules taäp tin to set the parameters.

# IPSECSG is a space separated list of remote gateways.
```

```
# FREESWANVI is a space seprated list of virtual interfaces
# for FreeS/WAN IPSec
# implementation. Only include those that are actually used.

# Allow IPSec protocol from remote gateway on external
# interface IPSec
# IPSec uses three main types of packet:
# IKE uses the UDP protocol and port 500
# ESP use the protocol number 50, and
# AH use the protocol 51

/sbin/depmod –a
echo "1" > /proc/sys/net/ipv4/ip_forward

# Set default build-in chains, default is DROP packets.
iptables -F INPUT
iptables -P INPUT DROP
iptables -F FORWARD
iptables -P FORWARD DROP
iptables -F OUTPUT
iptables -P OUTPUT DROP

EXTERNAL_INTERFACE="eth0"
INTERNAL_INTERFACE="eth1"
ANYWHERE="any/0"

IPSECSG=208.164.186.2
IPSECLC=208.164.186.1

FREESWANVI="ipsec0"
LOCALNET_1="192.168.1.0/24"
LOCALNET_2="192.168.2.0/24"
```

```
LO_INTERFACE="lo"
LO_IP="127.0.0.1/32"

##################
# FORWARD chains #
##################
iptables -A FORWARD -o $FREESWANVI -p TCP -s $LOCALNET_1 \
-d $LOCALNET_2 -j ACCEPT

iptables -A FORWARD -i $FREESWANVI -p TCP -s $LOCALNET_2 \
 -d $LOCALNET_1 -j ACCEPT

iptables -A FORWARD -o $FREESWANVI -p ICMP -s $LOCALNET_1 \
-d $LOCALNET_2 -j ACCEPT
iptables -A FORWARD -i $FREESWANVI -p ICMP -s $LOCALNET_2 \
-d $LOCALNET_1 -j ACCEPT

#################
# INPUT chains  #
#################
iptables -A INPUT -p TCP -s $LOCALNET_1 -d $IN_IP \
  -m state --state ESTABLISHED,RELATED -j ACCEPT

iptables -A INPUT -i $EXTERNAL_INTERFACE -p UDP \
-s $IPSECSG -d $IPSECLC --dport 500 -j ACCEPT

iptables -A INPUT -i $EXTERNAL_INTERFACE -p 50 \
-s $IPSECSG -d $IPSECLC -j ACCEPT

iptables -A INPUT -i $EXTERNAL_INTERFACE -p 51 \
-s $IPSECSG -d $IPSECLC -j ACCEPT

iptables -A INPUT -i $FREESWANVI -s $LOCALNET_2 \
-d $LOCALNET_1 -j ACCEPT

iptables -A INPUT -p ALL -i $LO_INTERFACE \
-d $LO_IP -j ACCEPT

iptables -A INPUT -p TCP -s $LOCALNET_1 -d $IN_IP \
```

```
--dport 21 -j ACCEPT

iptables -A INPUT -p TCP -s $LOCALNET_1 -d $IN_IP \
--dport 80 -j ACCEPT

iptables -A INPUT -p TCP -s $LOCALNET_1 -d $IN_IP \
--dport 443 -j ACCEPT

iptables -A INPUT -p ICMP -s $LOCALNET_1 -j ACCEPT

################
# OUTPUT chains #
################

iptables -A OUTPUT -p TCP -s $IN_IP -d $LOCALNET_1 \
-m state --state ESTABLISHED,RELATED -j ACCEPT

iptables -A OUTPUT -o $EXTERNAL_INTERFACE -p UDP \
-s $IPSECLC --sport 500 -d $IPSECSG -j ACCEPT

iptables -A OUTPUT -o $EXTERNAL_INTERFACE -p 50 -j ACCEPT
iptables -A OUTPUT -o $EXTERNAL_INTERFACE -p 51 -j ACCEPT

iptables -A OUTPUT -o $FREESWANVI -s $LOCALNET_1 \
-d $LOCALNET_2 -j ACCEPT

iptables -A OUTPUT -p ALL -o $LO_INTERFACE \
-s $LO_IP -j ACCEPT

iptables -A OUTPUT -p ICMP -d $LOCALNET_1 -j ACCEPT

Ở đây EXTERNAL_INTERFACE="eth0"      # Interface kết nối với bên ngoài.
Ở đây ANYWHERE="any/0"               # Ý nghĩa là 0.0.0.0/0.
Ở đây IPSECSG="208.164.186.2"        # Địa chỉ IP của VPN gateway từ xa.
Ở đây FREESWANVI="ipsec0"            # Virtual interface cho FresS/Wan.
```

Bắt đầu từ RH Linux 7.x chúng ta có một công cụ mới để viết các rule trong script firewall. Trong tập tin script firewall chúng ta dùng iptables thay cho ipchains như ở phần trước, trong phần này tôi không có ý định trình bày về iptables mà chỉ dùng chúng để viết các rule trong firewall. IPTABLES thì có nhiều ưu việt hơn so với ipchains, các bạn có thể tham khảo iptables với lệnh "man iptables" hoặc các tài liệu liên quan. Đừng quên thêm hoặc kiểm tra những qui tắc firewall trong những gateway khác.

Chú ý: *Trong trường hợp iptables không được hỗ trợ trên RH Linux 7.x bạn phải cài đặt iptables và sau đó làm cho kernel hỗ trợ kernel.*

Kiểm tra sự cài đặt

- Khởi động lại cả hai gateway.
- Xem xét tập tin /var/log/messages cho nhiều thông tin về trạng thái làm việc của IPSEC.

  Khảm xét tập tin "/var/log/secure" cho nhiều dấu hiệu để sửa lỗi.
- Trên cả hai gateway, toàn bộ những tập tin sau sẽ có trong thư mục "/proc/net/".
- Bây giờ thực thi lệnh để trình bày thông tin debug nhỏ nhất và sẽ thấy thông tin xuất hiện trên màn hình giống như sau:

```
[root@deep /]# ipsec look

deep.openna.com Fri Feb 4 17:25:17 EST 2000


============.============

192.168.1.1/32 -> 192.168.2.2/32 => tun0x106@192.168.1.2
esp0x4450894d@192.168.1.2 ah0x4450894c@192.168.1.2

.....=.....

ah0x3350f551@192.168.1.1 AH_HMAC_MD5: dir=in ooowin=32 seq=115
bit=0xffffffff alen=128 aklen=16
life(c,s,h)=bytes(16140,0,0)add(51656,0,0)use(54068,0,0)packets(115,0,0)
idle=499

ah0x4450894c@192.168.1.2 AH_HMAC_MD5: dir=out ooowin=32 seq=2828
alen=128 aklen=16
life(c,s,h)=bytes(449488,0,0)add(51656,0,0)use(51656,0,0)packets(2828,0,0
) idle=6

esp0x3350f552@192.168.1.1 ESP_3DES: dir=in ooowin=32 seq=115
bit=0xffffffff eklen=24

life(c,s,h)=bytes(13380,0,0)add(51656,0,0)use(54068,0,0)packets(115,0,0)
idle=499

esp0x4450894d@192.168.1.2 ESP_3DES: dir=out ooowin=32 seq=2828
eklen=24

life(c,s,h)=bytes(381616,0,0)add(51656,0,0)use(51656,0,0)packets(2828,0,0
) idle=6

tun0x105@192.168.1.1 IPIP: dir=in 192.168.1.2 -> 192.168.1.1
life(c,s,h)=add(51656,0,0)

tun0x106@192.168.1.2 IPIP: dir=out 192.168.1.1 -> 192.168.1.2

life(c,s,h)=bytes(327581,0,0)add(51656,0,0)use(51656,0,0)packets(2828,0,0
) idle=6
```

192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	ipsec0
192.168.1.1	0.0.0.0	255.255.255.255	UH	0	0	0	eth0
192.168.1.2	192.168.1.2	255.255.255.255	UGH	0	0	0	ipsec0
Destination	Gateway	Genmask	Flags	MSS	Window	irtt	Iface

- Thử ping 192.168.1.2 từ client 192.168.2.2. Nếu lệnh này làm việc thì khi đó bạn đã cài đặt chính xác. Nếu chúng không làm việc, kiểm tra lại network của bạn và bảo đảm rằng 208.164.186.1 có thể gởi đến 208.164.186.2 và đảm bảo rằng TCP/IP forwarding đã được làm cho có hiệu lực, cũng bảo đảm rằng không có những qui tắc khóa package cần cho IPSEC hoặc thử masquerade chúng trước những qui tắc cho phép lưu thông liên quan tới IPSec. Để kiểm tra việc này dùng lệnh ping từ một subnet này đến một subnet khác ví dụ như kiến trúc mạng:

```
208.164.186.1 ···· 205.151.222.250 ···· 205.151.222.251 ···· 208.164.186.2
    |                                                          |
192.168.1.0/24                                                 |
192.168.2.0/24                                                 |
    |                                                          |
192.168.1.2                                                192.168.2.2
```

Một chú ý cuối cùng về việc kiểm tra tiến trình cài đặt FreeSWAN IPSEC, nếu bạn phát hiện một vấn đề mà bạn không thể giải quyết, bạn có thể dùng lệnh sau để ghi lại toàn bộ thông tin (bao gồm nội dung của các file, logfile ...) liên quan tới IPSec gởi tới Linux-IPSec Mailing List (linux-ipsec@clinet.fi) để có sự giúp đỡ.

- Dùng lệnh sau để ghi lại thông tin như nói ở trên.

```
[root@deep /]# ipsec barf > result
```

Lệnh này được cung cấp như một tiện ích cho việc sửa lỗi từ xa, nó ghi lại tất cả các thông tin liên quan để chẩn đoán một vấn đề trong IPSEC.

Các lệnh cần tham khảo thêm

Chúng tôi không có ý định diễn giải tất cả các lệnh dưới đây, bạn có thể tham khảo bằng cách dùng lệnh man với những lệnh bạn cần biết.

$ man ipsec (8)	- invoke IPSEC utilities
$ man ipsec atoaddr, addrtoa (3)	- convert Internet addresses to and from ASCII
$ man ipsec atoasr (3)	- convert ASCII to Internet address, subnet, or range
$ man ipsec atobytes, bytestoa (3)	- convert binary data bytes from and to ASCII formats
$ man ipsec atodata, datatoa (3)	- convert binary data from and to ASCII formats

```
$ man ipsec atosa, satoa (3)          - convert IPSEC Security Association IDs to and from ASCII
$ man ipsec atosubnet, subnettoa (3)  - convert subnet/mask ASCII form to and from addresses
$ man ipsec atoul, ultoa (3)          - convert unsigned-long numbers to and from ASCII
$ man ipsec auto (8)                  - control automatically-keyed IPSEC connections
$ man ipsec barf (8)                  - spew out collected IPSEC debugging information
$ man ipsec bitstomask (3)            - convert bit count to Internet subnet mask
$ man ipsec eroute (8)                - manipulate IPSEC extended routing tables
$ man ipsec goodmask (3)              - is this Internet subnet mask a valid one?
$ man ipsec hostof (3)                - given Internet address and subnet mask, return host part
$ man ipsec klipsdebug (8)            - set Klips (kernel IPSEC support) debug features and level
$ man ipsec look (8)                  - show minimal debugging information
$ man ipsec manual (8)                - take manually-keyed IPSEC connections up and down
$ man ipsec masktobits (3)            - convert Internet subnet mask to bit count
$ man ipsec optionsfrom (3)           - read additional "command-line" options from file
$ man ipsec pluto (8)                 - IPsec IKE keying daemon
$ man ipsec ranbits (8)               - generate random bits in ASCII form
$ man ipsec rangetoa (3)              - convert Internet address range to ASCII
$ man ipsec rsasigkey (8)             - generate RSA signature key
$ man ipsec setup (8)                 - control IPSEC subsystem
$ man ipsec spi (8)                   - manage IPSEC Security Associations
$ man ipsec spigrp (8)                - group/ungroup IPSEC Security Associations
$ man ipsec subnetof (3)              - given Internet address and subnet mask, return subnet number
$ man ipsec tncfg (8)                 - associate IPSEC virtual interface with real interface
$ man ipsec whack (8)                 - control interface for IPSEC keying daemon
$ man ipsec.conf (5)                  - IPSEC configuration and connections
$ man ipsec.secrets (5)               - secrets for IKE/IPsec authentication
$ man ipsec (8)                       - invoke IPSEC utilities
$ man ipsec atoaddr, addrtoa (3)      - convert Internet addresses to and from ASCII
$ man ipsec atoasr (3)                - convert ASCII to Internet address, subnet, or range
$ man ipsec atobytes, bytestoa (3)    - convert binary data bytes from and to ASCII formats
$ man ipsec atodata, datatoa (3)      - convert binary data from and to ASCII formats
$ man ipsec atosa, satoa (3)          - convert IPSEC Security Association IDs to and from ASCII
$ man ipsec atosubnet, subnettoa (3)  - convert subnet/mask ASCII form to and from addresses
```

$ man ipsec atoul, ultoa (3) - convert unsigned-long numbers to and from ASCII
$ man ipsec auto (8) - control automatically-keyed IPSEC connections
$ man ipsec barf (8) - spew out collected IPSEC debugging information
$ man ipsec bitstomask (3) - convert bit count to Internet subnet mask
$ man ipsec eroute (8) - manipulate IPSEC extended routing tables
$ man ipsec goodmask (3) - is this Internet subnet mask a valid one?
$ man ipsec hostof (3) - given Internet address and subnet mask, return host part
$ man ipsec klipsdebug (8) - set Klips (kernel IPSEC support) debug features and level
$ man ipsec look (8) - show minimal debugging information
$ man ipsec manual (8) - take manually-keyed IPSEC connections up and down
$ man ipsec masktobits (3) - convert Internet subnet mask to bit count
$ man ipsec optionsfrom (3) - read additional ``command-line'' options from file
$ man ipsec pluto (8) - IPsec IKE keying daemon
$ man ipsec ranbits (8) - generate random bits in ASCII form
$ man ipsec rangetoa (3) - convert Internet address range to ASCII
$ man ipsec rsasigkey (8) - generate RSA signature key
$ man ipsec setup (8) - control IPSEC subsystem
$ man ipsec spi (8) - manage IPSEC Security Associations
$ man ipsec spigrp (8) - group/ungroup IPSEC Security Associations
$ man ipsec subnetof (3) - given Internet address and subnet mask, return subnet number
$ man ipsec tncfg (8) - associate IPSEC virtual interface with real interface
$ man ipsec whack (8) - control interface for IPSEC keying daemon
$ man ipsec.conf (5) - IPSEC configuration and connections
$ man ipsec.secrets (5) - secrets for IKE/IPsec authentication

Chương 25:

CÀI ĐẶT SQUID PROXY SERVER
Trên Red Hat Linux 7.2

Trong chương này chúng ta sẽ đề cập đến các vấn đề chính như sau:

- Proxy Caching.
- Network Addres Translation (NAT).
- Khác biệt giữa một Packet Filter và một Proxy Server.
- Sự thực thi Squid Web Proxy Cache Server.
- Khởi động và kiểm tra Squid.
- Cấu hình các proxy client.

Tổng quan

Một proxy server đóng vai trò như một máy trung chuyển giữa các máy hoặc các mạng khác nhau chẳng hạn như một mạng nội bộ LAN và Internet. Nó được thực hiện việc lưu trữ cho các dịch vụ nào đó và cho công việc quản trị và đảm bảo an toàn cho mạng. Các proxy server có thể thực thi những chức năng khác nhau như proxy caching và Network Address Translation (NAT). NAT là tính năng kỹ thuật của một gateway mức network (network-level gateway). Nhưng một vài nhà sản xuất đưa luôn tính năng này vào trong sản phẩm của họ. Trước khi cài Squid chúng ta cần phân biệt rõ hai tính năng sau:

Proxy Caching

Các proxy server được thực thi ở tầng ứng dụng (Application layer) và xử lý những protocol Internet cụ thể như Hypertext Transfer Protocol (HTTP) và File Transfer Protocol (FTP). Những qui định được thiết đặt trên proxy server và chỉ định làm thế nào một yêu cầu trạm làm việc được xử lý.

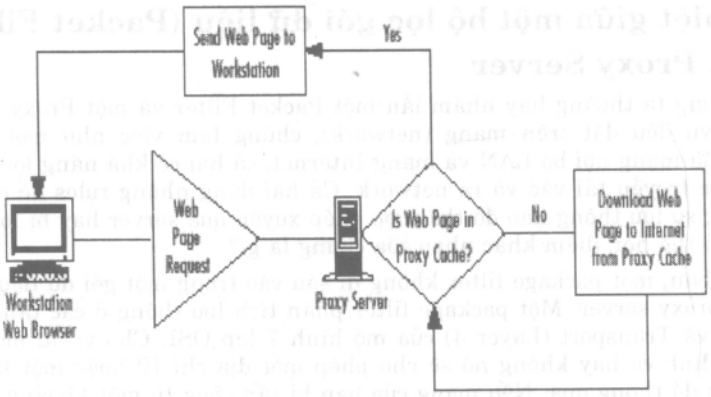
Một trong những công việc chính của của một proxy server là tạo vùng đệm lưu trữ (cache) những trang web và những tập tin FTP cho các proxy client. Những kiểu proxy server này được gọi là proxy cache server. Chức năng caching của proxy server làm tăng khả năng thực hiện của mạng bằng cách giảm lượng dữ liệu được truyền ra bên ngoài của một mạng nội bộ.

Để thực thi proxy caching, mỗi workstation trong network được cấu hình như một proxy client cho một dịch vụ cụ thể. Ví dụ một web proxy client có thể cấu hình browser của họ biết được proxy server. Khi client gởi một yêu cầu tải về một trang web nào đó thì client gởi yêu cầu này đến proxy server. Proxy server có lưu trữ những trang web được viếng thăm gần đây từ các client thông qua proxy server.

Proxy server kiểm tra trong cache của nó xem nếu có sẵn những trang web được yêu cầu thì những trang web cache được gởi tới client, nếu không có sẵn trong cache, proxy server sẽ download những trang web từ yêu cầu của client từ một web site cụ thể nào đó lưu vào trong cache và sau đó gởi những trang này tới các client.

Đảm bảo rằng những trang web này trong cache không quá cũ, những dữ liệu trong cache sẽ hết hạn sau một thời gian được chỉ định trước. Trong chương trình Squid, việc cài đặt này gọi là thời gian làm mới đối tượng (object refreshtime). Thời gian làm mới đối tượng đảm bảo rằng dữ liệu cũ không được truyền tải tới các proxy client.

Tiến trình này tăng khả năng thực hiện mạng bởi vì trang web được gởi ngay lập tức tới client từ proxy server mà không phải download từ internet mỗi lần. Tốc độ này tăng khả năng truy cập Internet và tiết kiệm băng thông (bandwidth). Quá trình này đơn giản chỉ thêm một số lượng nhỏ của dữ liệu tới Web cache có một tác động quan trọng trong tốc độ browsing và sử dụng bandwidth trên mạng.



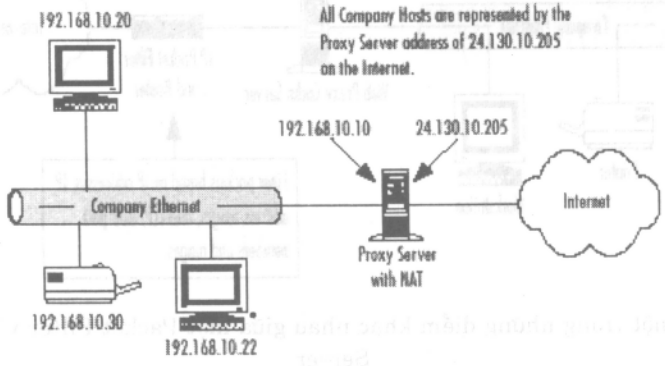
Minh họa chức năng Web Caching của Proxy Server

Chú ý: *Một tiện ích của các proxy server là các người quản trị proxy có thể thêm các qui tắc để lọc (filter) nội dung. Ví dụ bạn có thể filter những trang web từ yêu cầu của client chứa đựng những từ nào đó trong địa chỉ. Nếu một người quản lý của bộ phận marketing phàn nàn rằng nhân viên của anh ấy dùng quá nhiều thời gian làm việc cho những công việc khác và anh ấy có thể yêu cầu administrator tạo các rule để filter các địa chỉ không cần thiết cho công việc mà nhân viên thường hay truy cập.*

Network Address Translation (NAT)

Proxy cache chỉ là một kỹ thuật của một proxy server. Nhiều proxy server được phân phát với khả năng hỗ trợ NAT. NAT cho phép một địa chỉ mạng bên trong của công ty ẩn từ Internet. Công ty được đại diện trên Internet với một địa chỉ IP mà không quan hệ gì đến những địa chỉ IP bên trong của công ty.

Sử dụng một proxy server, tất cả các lưu thông về hướng Internet bên trong công ty được gởi tới proxy server. Proxy gán mỗi package của client địa chỉ IP của proxy server trước khi truyền tải nó đi trong Internet. Khi package trả lời đến, proxy server gởi nó đến máy đã làm yêu cầu ban đầu này. Thủ tục này bảo vệ những địa chỉ bên trong mạng nội bộ từ Internet. Vì vậy sẽ có nhiều khó khăn cho một hacker tấn công hệ thống của chúng ta vì không thể truy cập trực tiếp từ Internet.



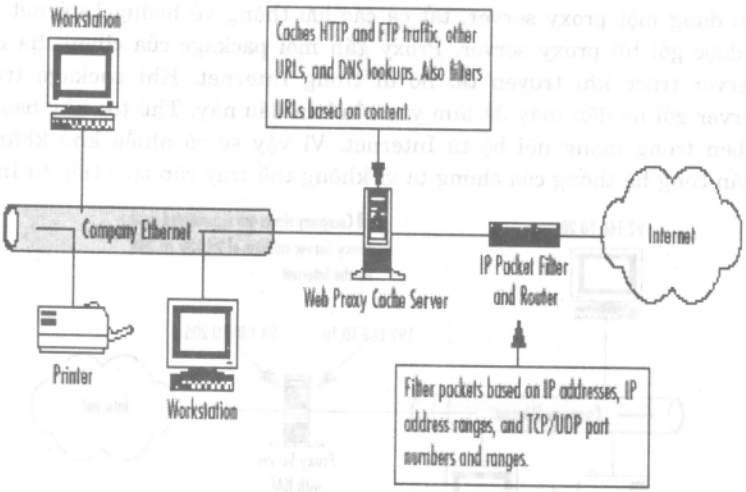
Minh họa chức năng Network Address Translation (NAT)

Khác biệt giữa một bộ lọc gói dữ liệu (Packet Filter) và một Proxy Server

Chúng ta thường hay nhầm lẫn một Packet Filter và một Proxy Server. Cả hai dịch vụ đều đặt trên mạng (network), chúng làm việc như một nơi trung chuyển giữa mạng nội bộ LAN và mạng Internet, cả hai có khả năng lọc những lưu thông được truyền tải vào và ra network. Cả hai dùng những rules để chỉ định có hay không sự lưu thông nào đó thì được phép xuyên qua server hay bị loại bỏ. Như vậy bạn có thể hỏi, điểm khác nhau của chúng là gì?

Cơ bản, một package filter không đi sâu vào trong một gói dữ liệu (package) như một proxy server. Một package filter phân tích lưu thông ở các tầng Network (Layer 3) và Transport (Layer 4) của mô hình 7 lớp OSI. Cho ví dụ một package filter chỉ định có hay không nó sẽ cho phép một địa chỉ IP hoặc một khoảng địa chỉ IP nào đó thông qua. Nếu mạng của bạn bị tấn công từ một khoảng địa chỉ IP nào đó, bạn có thể tạo một rule để loại bỏ tất cả các package ban đầu đến từ khoảng địa chỉ trên. Bạn cũng có thể lọc (filter) khả năng tải (traffic) bằng dịch vụ hoặc port. Ví dụ bạn có thể tạo một rule để loại bỏ tất cả các lưu thông trực tiếp tới những port đang được lắng nghe như FTP, rlogin, telnet hoặc bên trong một khoảng port nào đó. Bạn cũng có thể filter những package ICMP bởi type/code như cho phép bạn chỉ loại bỏ những kiểu nào đó của ICMP. Đây là hữu ích trong sự bảo vệ của chính bạn với kiểu tấn công denial-of-service (DoS).

Một proxy server có thể phân tích những package ở tầng ứng dụng (layer 7) của mô hình OSI. Việc này cho phép nhiều tính mềm dẻo, bởi vì lưu thông bên trong một service chẳng hạn như port 80 (HTTP) có thể được filter. Như được giới thiệu rằng cho phép proxy server phân tích lưu thông HTTP hoặc FTP và chỉ định có hay không traffic được thông qua. Nếu một rule tồn tại ngăn chặn bất kỳ những địa chỉ web với từ "jobs" trong nó thì bất kỳ yêu cầu HTTP URL với "jobs" trong nó sẽ bị loại bỏ. Trong phần tiếp theo bạn sẽ thực thi một proxy server với dịch vụ Web caching. Chúng ta sẽ cài đặt, cấu hình và chạy Squid trên RH Linux7.2.



Minh hoạ một trong những điểm khác nhau giữa một Packet Filter và một Proxy Server

Sự thực thi Squid Web Proxy Cache Server

Server Squid Web Proxy Cache thì miễn phí. Nó cho phép người quản trị hệ thống cài đặt một dịch vụ Web proxy caching, thêm những qui tắc điều khiển (rule) và ngay cả cache DNS lookups. Có một điều quan trọng cần thông báo rằng Squid chỉ là một Web proxy-caching server. Nó không hỗ trợ những tính năng proxy khác chẳng hạn như tính năng NAT hoặc firewall.

Hướng dẫn cài đặt

- Các lệnh tương thích với unix.
- Đường dẫn nguồn là /var/tmp (các đường dẫn khác cũng có thể được).
- Sự cài đặt đã được kiểm tra trên Red Hat Linux 7.1 và 7.2.
- Tất cả các bước cài đặt sẽ được thực hiện với supper-user "root".
- Phiên bản Squid là squid-2.4.STABLE1-5.

Dạng lưu trữ Tarballs

Để biết được các tập tin nào được thêm vào khi bạn cài đặt Squid, trước hết hãy tạo danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt Squid, sau đó so sánh chúng dùng 'diff' để tìm ra các tập tin nào là được dặt ở đâu. Đơn giản chạy lệnh 'find /* > Squid1' trước và 'find /* > Squid2' sau khi bạn cài đặt phần mềm, và sử dụng lệnh 'diff Squid1 Squid2 > Squid-Installed' để lấy một danh sách các thay đổi nào đó.

Cài đặt Squid

Squid được cung cấp như một package RPM trong Red Hat Linux. Nếu bạn hoàn thành một tiến trình cài đặt theo ý muốn và cài đặt mọi thứ trên hệ thống RadHat Linux của bạn thì Squid sẽ được cài đặt sẵn trên hệ thống của bạn. Nếu không bạn cần download nó ở một website đáng tin cậy hoặc cài đặt chúng từ CD Red Hat của bạn, trong CD này có package squid-2.4.STABLE1-5.i386.rpm. Bạn cũng có thể download các tập tin nguồn và biên dịch chúng. Tập tin mã nguồn thì lợi ích bởi vì chúng cho phép bạn hiệu chỉnh một vài tùy chọn cho việc biên dịch mà không được đưa vào trong phiên bản nhị phân. Ví dụ hỗ trợ SNMP không được đưa vào trong phiên bản nhị phân, nếu yêu cầu hỗ trợ SNMP, bạn phải download và biên dịch mã nguồn của squid. Những hệ thống hiện tại Squid có thể chạy được là: Linux, SCO Unix, BSD/OS, NetBSD, MkLinux, Solaris, NeXTStep, OS/2 Warp, AIX.

Hoàn thành những bước sau để đảm bảo rằng Squid được cài đặt trên hệ thống Red Hat Linux của bạn.

- Để bảo đảm rằng squid RPM được cài đặt trên hệ thống của bạn, gõ vào dòng lệnh sau:

```
rpm –qa | grep squid
```

- Bạn sẽ nhận sự trả lời như sau nếu Squid được cài đặt:
  squid-2.4.STABLE1
- Nếu bạn nhận sự trả lời này thì bạn hãy sẵn sàng để cấu hình Squid.
- Nếu bạn không nhận message trả lời như trên, truy cập tới website www.Red Hat.com và nhấp chuột vào clink Download. Thực hiện một sự tìm kiếm RPM bằng cách gõ vào squid trong field Find Latest RPMs By Keyword. Nhấp nút Search phiên bản mới nhất RPM sẽ xuất hiện. Bạn có thể nhận được package squid-2.4.STABLE1.i386.rpm trong CD cài đặt Red Hat Linux7.2.
- Cài đặt Squid từ package dạng RPM bằng cách dùng lệnh sau:
  rpm –i squid-2.4.STABLE1.i386.rpm

Cấu hình Squid

Một khi Squid được cài đặt, bạn phải cấu hình squid từ tập tin /etc/squid/squid.conf. Tập tin này định nghĩa cấu hình squid chẳng hạn như HTTP port number mà squid dùng để lắng nghe những yêu cầu HTTP, những yêu cầu đến và đi, thông tin timeout và dữ liệu truy cập firewall. Tập tin /etc/squid/squid.conf được tạo trong quá trình cài đặt.

Tập tin squid.conf được cấu hình cho tiến trình cài đặt mặc định, việc cấu hình squid có thể được sử dụng sau một vài thay đổi. Bạn phải hiệu chỉnh lại tập tin squid.conf bởi vì từ chối việc truy cập tới tất cả các browser bởi mặc định, việc cấu hình squid hoàn thành đến khi bạn đã thay đổi tập tin squid.conf.

Như được công bố trong tập tin cấu hình squid. Nếu bạn không cần thay đổi một tiến trình cài đặt mặc định thì bạn không cần thiết phải bỏ dấu "#" trước mỗi dòng.

Mỗi tùy chọn cấu hình trong tập tin squid.conf được nhận biết như một tag. Mỗi tag là một sự cấu hình Squid. Ví dụ port yêu cầu HTTP client được nhận biết như tag http_port.

Tập tin squid.conf trình bày phần tùy chọn network. Nó trình bày sự cấu hình cho socket mà squid sẽ lắng nghe cho những yêu cầu HTTP client. Port number mặc định được chỉ định TCP port 3128 và cũng là port mặc định được sử dụng bởi các proxy client để gởi các yêu cầu tới proxy cache server. Nếu bạn thay đổi port này trên proxy cache server bạn cũng sẽ cần thay đổi nó trên proxy client.

"accelerator mode" được đề cập trong phần diễn giải http_port cho phép Squid hành động như một Web server. Về cơ bản Squid dịch những yêu cầu web client bằng cách thay đổi server nơi gởi tới và port và gởi yêu cầu đó tới Web server. Trả lời yêu cầu từ trong cache và squid sẽ gởi nó tới client. Việc này giảm bớt lưu thông tới Web server và có thể bảo vệ web server thông qua việc filter. Chương này sẽ không minh hoạ accelerator mode nhưng bạn nên nhận thấy điều này.

Tag http_port

Tag http_port cấu hình HTTP port trên Squid để lắng nghe các proxy client. Bởi mặc định Squid không lắng nghe các proxy client trên bất kỳ các port nào. Vì thế bạn phải mở tối thiểu một port để cho việc sử dụng tag http_port trong Squid. Port mặc định là 3128 cũng thường được dùng cho dịch vụ này. Bạn có thể cấu hình Squid để lắng nghe trên cả hai port bằng cách đưa vào mỗi số trong tag http_port. Trong ví dụ dưới đây bạn sẽ cấu hình Squid để lắng nghe trên các port 3128 và 8080 cho các proxy client.

- Mở tập tin /etc/squid/squid.conf với lệnh:
  ```
  vi /etc/squid/squid.conf
  ```
- Nhấn phím i để vào mode insert và trỏ tới phần.
  ```
  # http_port 3128
  ```
- Bỏ dấu (#) và thêm port 8080 vào cuối dòng trên:
  ```
  http_port  3128 8080
  ```
- Nhấn Esc để thoát khỏi mode insert và gõ vào như sau để đóng file.
  ```
  :x
  ```

Bạn sẽ cấu hình tập tin squid.conf thêm nữa cho phần kế tiếp.

Tag Cache_dir

Tag cache_dir chỉ rõ ở đâu dữ liệu cache được lưu trữ. Bạn có thể chỉ rõ những thư mục khác nhau cho thư mục cache của bạn bằng cách dùng nhiều hơn một tag cache_dir trong tập tin squid.conf. Trong sự minh họa này bạn sẽ chỉ sử dụng một thư mục cache

```
cache_dir ufs /var/spool/squid 100 16 256
```

Định nghĩa tag mặc định cache_dir trong tập tin squid.conf

Giá trị Tag	Sự diễn giải
cache_dir	Định nghĩa giá trị của thư mục cache được sử dụng cho Squid.
Ufs	Squid giả sử rằng một Unix tập tin system được dùng cho hệ thống lưu trữ của cache.
/var/spool/squid	Thư mục cho tất cả các đối tượng được cache sẽ lưu trong /var/spool/squid.
100	Tổng số dữ liệu cache mà quid sẽ lưu trữ trong thư mục cache. Kích thước cache mặc định là 100MB.
16	Thiết đặt số thư mục con tạo trong cache.
256	Thiết đặt số giây các thư mục con được tạo trong cache. Nếu cache của bạn quá lớn bạn có thể tăng giá trị này.

Bạn có thể thay đổi những giá trị này để phù hợp với những gì bạn cần. Nếu bạn thực hiện công việc này nên nhớ bỏ dấu # ở phía trước tag. Nếu bạn không loại bỏ dấu # thì giá trị mặc định sẽ được sử dụng.

Để minh hoạ, bạn sẽ sử dụng giá trị mặc định tag cache_dir. Thư mục cache /var/spool/squid được tạo một cách tự động khi squid được cài đặt. Kiểm tra thư mục cache mặc định có sẵn hay chưa hoàn thành những bước sau:

- Chuyển vào thư mục spool bằng cách thực hiện lệnh:

  cd /var/spool

- Liệt kê nội dung thư mục để xác nhận thư mục cache của bạn đã sẵn có với lệnh "ls"

- Bạn sẽ nhận được một sự trả lời giống như sau, tùy thuộc vào cấu hình hệ thống của bạn. Thư mục squid thì sẵn có trong sự trả lời này.

  Anacron cron lpd mqueue rwho simpull up2date uucppublic
  voice at fax mail news samba squid ucp vbox

Tag acl

Tag acl cho phép bạn định nghĩa một danh sách truy cập. Danh sách truy cập có thể bao gồm các địa chỉ IP của client, một khoảng của các IP, một địa chỉ URL, một địa chỉ IP nội bộ hoặc một domain. Bất kỳ danh sách truy cập bạn định nghĩa sử dụng tag acl có thể sau đó được sử dụng để cho phép (allow) hoặc từ chối (deny) các yêu cầu tới cache server. Ví dụ như nếu bạn định nghĩa một khoảng của các địa chỉ sử dụng tag acl, bạn có thể cho phép hệ thống sử dụng một địa chỉ từ khoảng của các địa chỉ IP này để sử dụng cache proxy.

Định nghĩa một danh sách rất quan trọng cho bạn. Nếu bạn không làm bạn sẽ không có bất kỳ sự định nghĩa để tạo các qui tắc theo ý muốn. Khi Squid tìm được một rule chẳng hạn như "từ chối truy cập hệ thống từ một khoảng địa chỉ IP từ 192.168.3.1 đến 192.168.3.254" nó sẽ kiểm tra một acl tương ứng được tạo mà định nghĩa khoảng địa chỉ IP này.

Trong phần minh hoạ phía dưới, bạn sẽ tạo một danh sách truy cập bao gồm các client sẽ truy cập server cache proxy của bạn. Bạn sẽ định nghĩa chúng bằng địa chỉ IP.

- Mở tập tin squid.conf (vi /etc/squid/squid.conf)

- Tìm đến vị trí mặc định tag acl, sau đó nhấn phím i để vào mode insert.

- Gõ vào sau tag acl và ngay sau 127.0.0.1/255.255.255.255 những dòng địa chỉ sau (Địa chỉ của bạn sẽ khác hơn tùy thuộc vào các proxy client của bạn).

  acl proxy_client1 src 24.130.8.227/255.255.252.0
  acl proxy_client2 src 24.130.10.205/255.255.252.0

- Nhấn Esc để thoát khỏi mode insert và gõ vào :x để ghi lại và thoát khỏi việc hiệu chỉnh tập tin squid.conf.

Tag http_access

Tag http_access cho phép hoặc từ chối truy cập tới squid. Bạn có thể cho phép hoặc từ chối tất cả các yêu cầu. Bạn cũng có thể cho phép hoặc từ chối các yêu cầu dựa trên một danh sách truy cập đã được định nghĩa. Nếu bạn loại bỏ tất cả các http_access thì mặc định tất cả các yêu cầu sẽ được cho phép.

Các proxy client sẽ không có thể dùng Squid proxy-caching server cho đến khi bạn thay đổi các tags http_access. Nên chú ý rằng một vài mức điều khiển truy cập đã được giới thiệu như vậy đừng loại bỏ các tag http_access.

Để minh hoạ cho việc này bạn sẽ cài đặt một điều khiển truy cập đơn giản. Bạn sẽ chỉ cho phép các máy trong mạng của bạn sử dụng proxy cache. Mỗi máy sẽ được chỉ định bởi địa chỉ IP của chính nó như được định nghĩa trong danh sách điều khiển truy cập (acl tag). Hệ thống này làm việc tốt cho các công ty nhỏ. Để thực thi việc điều khiển truy cập trong một công ty lớn, bạn nên tạo các lớp (class) hoặc các user và sau đó cho phép hoặc từ chối các class này. Bạn có thể thiết đặt việc điều khiển truy cập chặt chẽ hơn trong Squid bằng cách xem các tài liệu ở website http://www.squid-proxy.org

Chú ý: *Squid sẽ không bao giờ được sử dụng mà không có một vài kiểu danh sách điều khiển truy cập hoặc kiểm tra tính xác thực hệ thống. Bạn phải nghiêm khắc với những người dùng Internet từ việc phản hồi những yêu cầu thông qua Web proxy cache.*

Để thực thi việc điều khiển truy cập tới proxy cache của bạn, hoàn thành những bước theo sau. Trong ví dụ này có hai proxy client được định nghĩa trong acl, proxy_client1 và proxy_client2 cần dùng proxy cache. Bởi mặc định localhost có thể sử dụng proxy cache. Để hiệu chỉnh Tag http_access trong tập tin squid.conf thực hiện các bước sau:

- Mở tập tin squid.conf (vi /etc/squid/squid.conf).
- Chuyển vào vị trí tag http_access rồi nhấn phím i để vào mode insert.
- Gõ vào theo sau tag http_access trước các "http_access deny" (Các tên acl của bạn có thể khác).

```
http_access allow proxy_client1
http_access allow proxy_client2
```

- Nhấn Esc để thoát khỏi mode insert và gõ vào :x để ghi lại và thoát khỏi việc hiệu chỉnh tập tin squid.conf.

Bạn cần phải xoá đi các qui tắc điều khiển truy cập mặc định chẳng hạn như "http_access deny all" trong tập tin squid.conf của bạn. Những qui tắc này sẽ không cho phép mọi người khai thác cache của bạn. Ví dụ, mặc dù trạng thái tag sau cùng là "http_access deny all" nó vẫn cho phép các yêu cầu đi vào từ các client bạn nhận biết rõ ràng, tất cả các yêu cầu khác sẽ bị từ chối.

Khởi động và kiểm tra Squid

Bây giờ bạn đã cấu hình Squid cho môi trường mạng của bạn. Để xác định Squid đang làm việc theo những sự định nghĩa của bạn, bạn phải khởi động Squid và sau đó sử dụng những chương trình squid client trên localhost để đảm bảo dữ liệu trang web được ghi trong cache. Hoàn thành những bước sau để khởi động và kiểm tra Squid.

- Khởi động Squid proxy cache bằng cách gõ vào dòng lệnh sau:
  /etc/rc.d/init.d/squid start
- Kiểm tra squid đang làm việc bằng cách sử dụng chương trình squid client. Gõ vào trên trình duyệt web của client:
  `http://www.squid-cache.org`
- Dữ liệu được ghi vào cache được trình bày trên màn hình của bạn. Nếu không có dữ liệu nào được ghi, bạn cần xem lại tập tin cấu hình squid.conf và xác định những cấu hình không chính xác.

Lúc này bạn đã khởi động thành công và đã kiểm tra Squid Web Proxy Cache server. Bây giờ bạn phải cấu hình các proxy client để sử dụng cache. Bạn sẽ biết cách làm thế nào để cấu hình một proxy client trong Netscape Navigator và Internet Explorer.

Cấu hình các proxy client

Một proxy client là một hệ thống (máy) sử dụng các dịch của Web proxy-caching server. Tùy thuộc vào cấu hình mạng của bạn, một client có thể hoặc không có thể phải cấu hình như một proxy client để sử dụng Web proxy cache server. Ví dụ một vài firewall được cấu hình chuyển tiếp (forward) tất cả các lưu thông port 80 tới Web proxy cache server. Trong trường hợp này các proxy client không cần cấu hình bằng tay. Trong trường hợp khác client tự động nhận biết thông tin proxy server trên network và sử dụng nó cho tất cả các truy cập internet. Trong sự minh họa này, bạn sẽ cấu hình một proxy client để sử dụng Squid Web Proxy Cache của bạn.

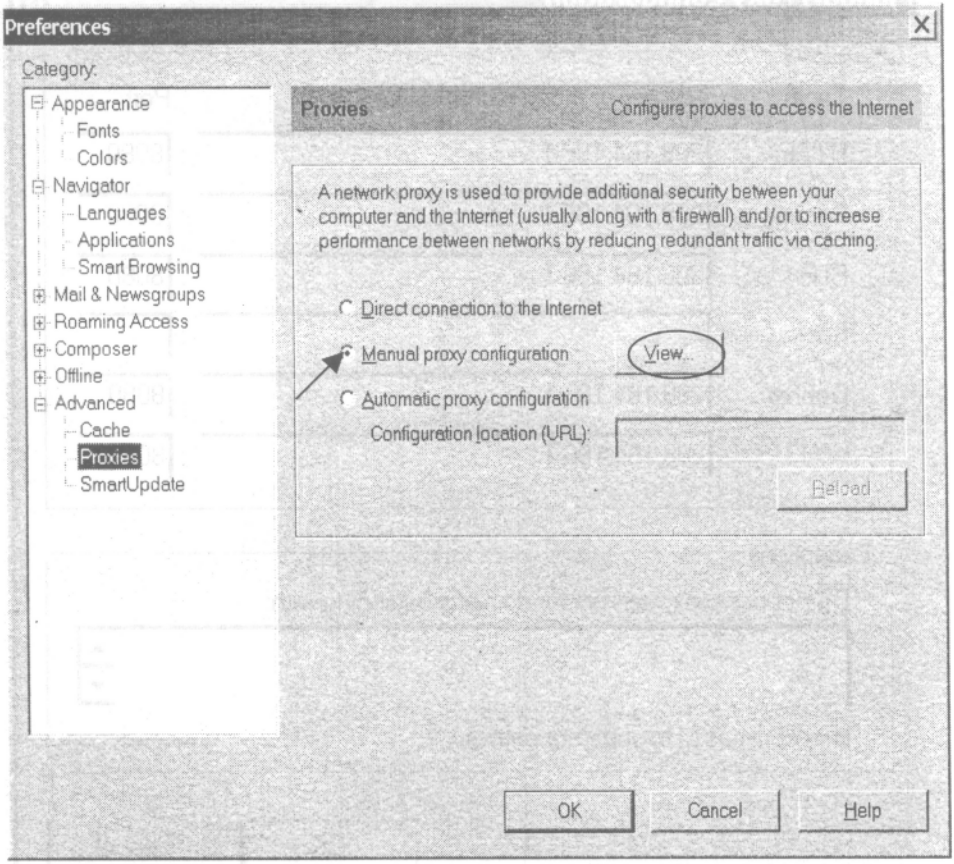
Cấu hình một proxy client thì nhanh hơn việc cấu hình squid. Tất cả các cấu hình proxy client được hoàn thành trong ứng dụng browser. Sự minh hoạ này chỉ dẫn cho bạn làm thế nào để cấu hình hai browser truy cập proxy server. Một máy là Linux đang chạy Netscape Navigator và một máy khác là hệ thống Microsoft Windows Millennium đang chạy Internet Explorer.

Cấu hình proxy cho Netscape Navigator để dùng với Squid

Trong phần này bạn sẽ cấu hình Netscape Navigator browser để nó gởi tất cả các yêu cầu web tới Squid Proxy Cache server. Client được cấu hình trong những bước sau là máy proxy_client2 như trong tập tin squid.conf.

Với Netscape Navigator, theo những bước đơn giản sau:

1. Login vào máy linux proxy client (proxy_client2).
2. Start X Windows bằng lệnh "startx".
3. Mở Netscape Navigator.
4. Vào menu View và chọn tùy chọn Preferences.
5. Trong cột danh mục, nhấp đúp vào nút Advanced ở phía bên trái và nhấp vào tùy chọn Proxies.
6. Chọn trên nút Manual proxy configuration và sau đó click vào nút View. Màn hình của bạn giống như sau:

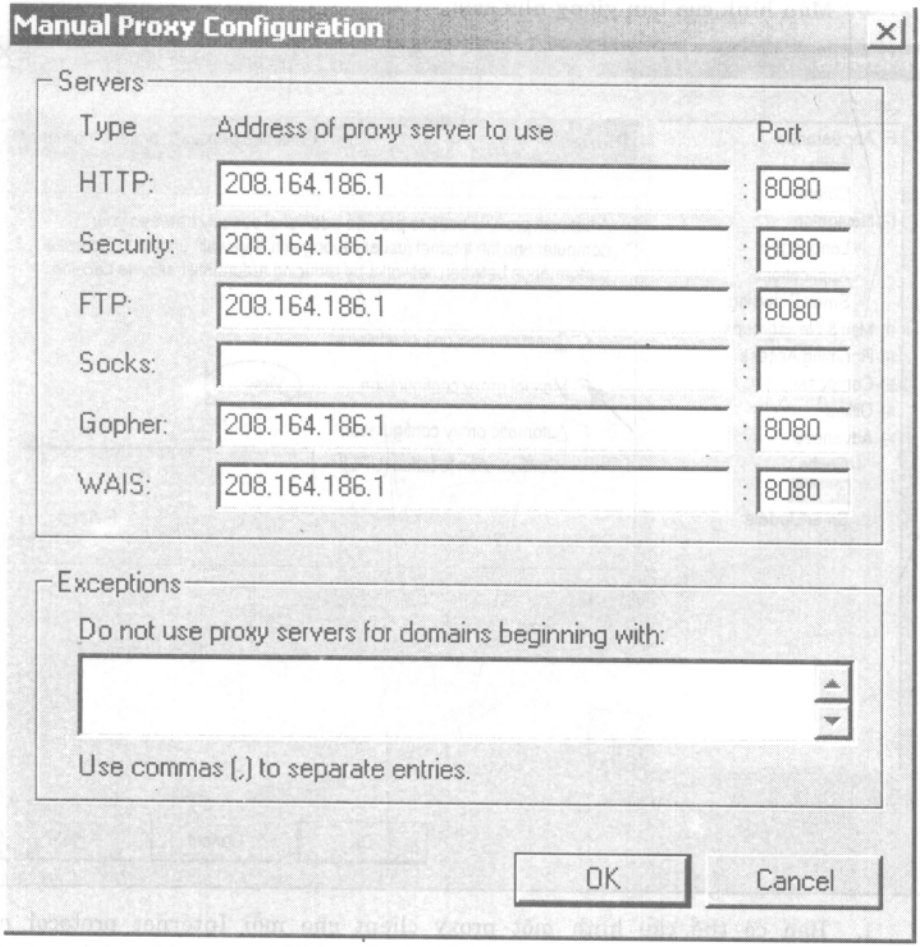


1. Bạn có thể cấu hình một proxy client cho mỗi Internet protocol mà Netscape hỗ trợ. Gõ vào địa chỉ IP của Squid Web proxy Cho ví dụ, nếu hệ thống squid của bạn là 208.164.186.1 và bạn cấu hình squid để lắng nghe các cache request trên port 3128 bạn có thể gõ vào giá trị như sau:

- HTTP: 208.164.186.1 Port: 8080
- Security: 208.164.186.1 Port: 8080

- FTP: 208.164.186.1 Port: 8080
- Gopher: 208.164.186.1 Port: 8080
- WAIS: 208.164.186.1 Port: 8080

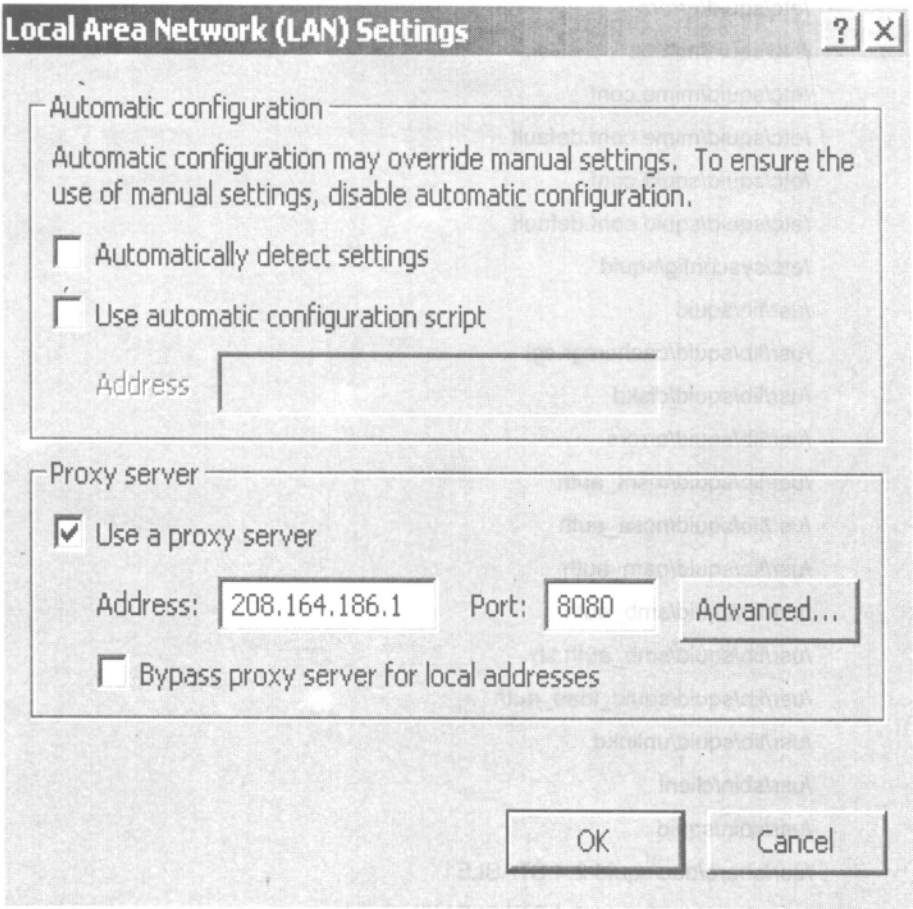
2. Nhấp OK hai lần để quay trở lại browser.
3. Trong Netscape Navigator, gõ vào URL www.squid-cache.org.
4. Trang chủ Squid sẽ xuất hiện, nếu không thì việc cài đặt của bạn không chính xác.



Cấu hình proxy cho Internet Explorer

Trong phần minh họa này bạn sẽ cấu hình Internet Explorer 5.5 trong MS Windows Millennium. Client được cấu hình theo những bước sau từ máy proxy_client1 như được định nghĩa trong tập tin squid.conf.

1. Mở Internet Explorer.
2. Nhấp vào menu Tools và chọnInternet Options.
3. Chọn tab Connections và sau đó click LAN Settings.
4. Đừng chọn Automatically Detect Setting.
5. Trong phần proxy server click vào Use a proxy server.
6. Trong trường (field) địa chỉ gõ vào địa chỉ IP của Squid Web Proxy Cache Server ví dụ như 208.164.186.1.
7. Trong trường Port gõ vào 8080.
8. Cửa sổ LAN Settings của bạn sẽ giống như sau:



9. Nhấp OK hai lần để quay trở lại browser.
10. Trong Internet Explorer, gõ vào URL www.squid-cache.org.

11. Trang chủ Squid sẽ xuất hiện, nếu không thì việc cài đặt của bạn không chính xác.

Nếu không có một thông báo lỗi nào thì việc thực thi Squid trên Linux 7.2 của bạn đã thành công. Bạn có thể làm tương tự cho các máy khác trong mạng nếu bạn thêm vào các client proxy trong mạng.

Các tập tin được cài đặt

/etc/logrotate.d/squid

/etc/rc.d/init.d/squid

/etc/squid

/etc/squid/errors

/etc/squid/mib.txt

/etc/squid/mime.conf

/etc/squid/mime.conf.default

/etc/squid/squid.conf

/etc/squid/squid.conf.default

/etc/sysconfig/squid

/usr/lib/squid

/usr/lib/squid/cachemgr.cgi

/usr/lib/squid/diskd

/usr/lib/squid/errors

/usr/lib/squid/msnt_auth

/usr/lib/squid/ncsa_auth

/usr/lib/squid/pam_auth

/usr/lib/squid/smb_auth

/usr/lib/squid/smb_auth.sh

/usr/lib/squid/squid_ldap_auth

/usr/lib/squid/unlinkd

/usr/sbin/client

/usr/sbin/squid

/usr/share/doc/squid-2.4.STABLE1

/usr/share/doc/squid-2.4.STABLE1/ChangeLog

/usr/share/doc/squid-2.4.STABLE1/FAQ-1.html

/usr/share/doc/squid-2.4.STABLE1/FAQ-10.html

/usr/share/doc/squid-2.4.STABLE1/FAQ-11.html

/usr/share/doc/squid-2.4.STABLE1/FAQ-12.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-13.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-14.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-15.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-16.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-17.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-18.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-19.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-2.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-20.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-21.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-22.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-23.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-24.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-25.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-3.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-4.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-5.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-6.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-7.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-8.html
/usr/share/doc/squid-2.4.STABLE1/FAQ-9.html
/usr/share/doc/squid-2.4.STABLE1/FAQ.html
/usr/share/doc/squid-2.4.STABLE1/FAQ.sgml
/usr/share/doc/squid-2.4.STABLE1/HTTP-codes.txt
/usr/share/doc/squid-2.4.STABLE1/Programming-Guide
/usr/share/doc/squid-2.4.STABLE1/Programming-Guide/Makefile
/usr/share/doc/squid-2.4.STABLE1/Programming-Guide/prog-guide.sgml
/usr/share/doc/squid-2.4.STABLE1/QUICKSTART
/usr/share/doc/squid-2.4.STABLE1/README
/usr/share/doc/squid-2.4.STABLE1/README.cygwin
/usr/share/doc/squid-2.4.STABLE1/Release-Notes-1.0.txt
/usr/share/doc/squid-2.4.STABLE1/Release-Notes-1.1.txt
/usr/share/doc/squid-2.4.STABLE1/debug-sections.txt

/usr/share/doc/squid-2.4.STABLE1/draft-vixie-htcp-proto-04.txt

/usr/share/doc/squid-2.4.STABLE1/mk-debugs.sh

/usr/share/doc/squid-2.4.STABLE1/rredir.c

/usr/share/doc/squid-2.4.STABLE1/rredir.pl

/usr/share/doc/squid-2.4.STABLE1/tree.3

/usr/share/doc/squid-2.4.STABLE1/url-normalizer.pl

/usr/share/doc/squid-2.4.STABLE1/user-agents.pl

/var/log/squid

/var/spool/squid

Chương 26:

CÀI ĐẶT OPENSSH CLIENT/SERVER
Trên Red Hat Linux 7.2

Trong chương này chúng ta sẽ đề cập đến các vấn đề chính như sau:

- Giới thiệu về SSH protocol.
- Các tầng của bảo mật SSH.
- Cài đặt và cấu hình OpenSSH.
- Cấu hình một OpenSSH client.
- Cấu hình ssh-agent với GNOME.
- Cấu hình ssh-agent.

Giới thiệu tổng quan

Để giúp cho các bạn không có điều kiện làm việc trên RH Linux 6.x mà đi thẳng vào nghiên cứu sử dụng OpenSSH trên RH Linux 7.2. Trong phần này tôi sẽ nói kỹ hơn về OpenSSL như các lợi ích của protocol SSH, các sự kiện liên tiếp xuất hiện khi một sự kết nối bảo mật được thiết lập tới một hệ thống từ xa, các tầng khác nhau (mô hình OSI) của SSH, các phương pháp đảm bảo SSH được dùng bởi các user kết nối tới hệ thống của bạn và cuối cùng là hướng dẫn cài đặt OpenSSL server.

Giới thiệu về SSH protocol

SSH (viết tắt của Secure Shell) là một protocol dùng cho việc tạo một sự kết nối bảo mật giữa hai hệ thống. Trong protocol SSH một máy client khởi xướng một sự kết nối với một máy server. Những chức năng bảo vệ sau đây được cung cấp bởi SSH:

- Sau một sự kết nối ban đầu, client có thể xác nhận rằng nó đang kết nối tới cùng server trong lúc các phiên session xảy ra.
- Client có thể truyền tải thông tin kiểm tra tính xác thực của nó tới server chẳng hạn như username và password trong một dạng được mã hoá.
- Tất cả dữ liệu được gởi và nhận trong một sự kết nối được truyền tải sử dụng sự mã hoá phức tạp làm cho nó trở nên khó khăn trong việc giải mã và đọc đối với những người không có quyền đọc nó.
- Client có khả năng sử dụng những ứng dụng X11 thay cho việc khởi tạo kết nối từ dấu nhắc shell. Kỹ thuật này cung cấp một bảo mật, giao tiếp đồ hoạ (được gọi là X11 forwarding)

Server tận dụng SSH, đặc biệt khi nó đang chạy một số dịch vụ. Nếu bạn đang sử dụng port forwarding vì thế những protocol không bảo mật như POP có thể được mã hoá cho sự trao đổi thông tin với máy từ xa. SSH làm nó có liên quan tới những kiểu mã hoá khác nhau của sự trao đổi thông tin thông thường được gởi một cách không an toàn trên các mạng công cộng. Red Hat Linux 7.2 bao gồm các package OpenSSH server (openssh-server) and client (openssh-clients).

Các package OpenSSH yêu cầu package OpenSSL. OpenSSL cài đặt một vài thư viện mật khẩu quan trọng mà OpenSSH cần cho việc cung cấp sự trao đổi thông tin được bảo mật. Bạn phải cài đặt OpenSSL trước khi cài đặt bất kỳ các package OpenSSH khác.

Một số lượng lớn các chương trình client và server có thể dùng protocol SSH bao gồm nhiều ứng dụng có sẵn và miễn phí. Một vài phiên bản SSH khác thì sẵn có cho hầu hết mọi hệ điều hành quan trọng đang sử dụng hiện tại. Ngay cả nếu các user đang kết nối tới hệ thống của bạn đang không chạy RH Linux, họ vẫn có thể tìm và sử dụng SSH client cho hệ thống của họ.

Các sự kiện liên tiếp của một kết nối SSH

Một chuỗi các sự kiện giúp cho việc bảo vệ tính nguyên vẹn của một sự trao đổi thông tin SSH giữa hai máy.

Đầu tiên, một tầng transport bảo mật được tạo, như vậy client biết rằng nó đang trao đổi thông tin với đúng server nó cần. Sau đó thông tin được mã hoá giữa client và server dùng mật khẩu đối xứng.

Kế tiếp, với một sự kết nối bảo mật tới server, client kiểm tra tính xác thực của chính nó với server không lo lắng rằng thông tin kiểm tra tính xác thực bị xâm phạm. Mặc định OpenSSH trên RH Linux sử dụng những khoá DSA hoặc RSA và phiên bản 2.0 của protocol SSH cho việc kiểm tra tính xác thực.

Cuối cùng, với client được kiểm tra tính xác thực tới server, một vài dịch vụ khác nhau có thể được bảo vệ thông qua sự kết nối chẳng hạn như một tương tác shell session, các ứng dụng X11.

Trong ví dụ sau, user1 trên máy client đang khởi tạo một sự kết nối SSH tới một server. Địa chỉ IP của server là 10.0.0.2 nhưng tên miền (domain) của nó có thể được dùng để thay thế. Tên login của user1 trên server là user2. Lệnh ssh được viết như sau:

```
[user1@machine1 user1]$ ssh user2@10.0.0.2
```

Client OpenSSH sẽ yêu cầu mật khẩu (passphrase) khoá riêng của user để giải mã khoá riêng, việc mà cần được sử dụng để thực hiện sự kiểm tra tính xác thực. Tuy nhiên mật khẩu khoá riêng (private key passphrase) thì không được gởi ngang qua sự kết nối bảo mật giữa client và server. Thay cho việc passphrase được sử dụng để mở khoá tập tin id_dsa và sinh ra một chữ ký điện tử sau đó gởi tới server. Nếu server có một bản copy của khoá công cộng (public key) của user, khoá này có thể được sử dụng cho việc xác định chữ ký và như vậy user được kiểm tra tính xác thực.

Trong ví dụ trên user đang dùng một khoá DSA (khoá RSA cũng có thể được sử dụng) và thấy dấu nhắc như sau:

```
Enter passphrase for DSA key '/home/user1/.ssh/id_dsa':.
```

Nếu việc kiểm tra tính xác thực public key sai do bất kỳ lý do nào (bởi vì passphrase gõ vào không chính xác hoặc thông tin kiểm tra tính xác thực không sẵn có trên server) thì kiểu khác của sự kiểm tra tính xác thực sẽ được thử nghiệm. Trong ví dụ của chúng ta OpenSSH server cho phép user1 kiểm tra tính xác thực sử dụng password của user2 bởi vì chữ ký đã gởi không phù hợp một public key được lưu trữ bởi user2

```
user2@machine2's password:
```

Với một password được gõ vào chính xác, uer sẽ được login. Dĩ nhiên user2 phải sẵn có một account trên máy 10.0.0.2.

```
Last login: Mon Apr 15 13:27:43 2001 from machine1
```

```
[user2@machine2 user2]$
```

Lúc này user có thể tương tác với shell trong cùng con đường như họ có thể làm với telnet hoặc rsh, ngoại trừ sự trao đổi thông tin được mã hoá.

Các tầng của bảo mật SSH

Protocol SSH cho phép bất kỳ các chương trình client và server xây dựng những đặc điểm kỹ thuật của protocol để trao đổi thông tin bảo mật. Hai version khác nhau của OpenSSH là version1 và version2, cả hai được kết hợp với OpenSSL cho việc mã hoá.

Cả hai version của SSH protocol sử dụng những tầng giống nhau của bảo mật để củng cố tính toàn vẹn của sự trao đổi thông tin từ nhiều khía cạnh khác nhau.

Tầng Transport (transport)

Vai trò chính của tầng transport là làm cho sự trao đổi thông tin giữa hai máy an toàn và bảo mật ở thời điểm và sau khi kiểm tra tính xác thực. Bình thường khi chạy trên TCP/IP, tầng transport hoàn thành việc này bằng cách thực hiện việc mã hoá và giải mã dữ liệu, xác định chính xác server cho việc kiểm tra tính xác thực và cung cấp tính nguyên vẹn của sự bảo vệ dữ liệu.

Một client liên lạc với một server sử dụng SSH protocol, một vài điểm được dàn xếp để cho hai hệ thống có thể xây dựng chính xác tầng transport.

- Trao đổi khoá.
- Thuật toán public key được sử dụng.
- Thuật toán mã hoá đối xứng được sử dụng.
- Thuật toán kiểm tra tính xác thực thông điệp được sử dụng.
- Kỹ thuật trong tính toán để tạo ra thuật toán băm (hash algorithm) được sử dụng.

Kiểm tra tính xác thực

Một khi tầng transport xây dựng một kênh bảo mật để chuyển thông tin giữa hai hệ thống, server thông báo cho client phương pháp kiểm tra tính xác thực khác được hỗ trợ chẳng hạn như sử dụng một private key hoặc gõ một password. Client sẽ thử kiểm tra tính xác thực chính nó tới server sử dụng bất kỳ các phương pháp được hỗ trợ.

Hầu hết các user yêu cầu một shell bảo mật với việc kiểm tra tính xác thực sử dụng một password. Không giống như các hệ thống kiểm tra tính xác thực khác, password được truyền tải đến server trong dạng không được mã hóa (cleartext). Tuy nhiên bởi vì toàn bộ password được mã hoá khi chuyển dịch trên tầng transport, nó có thể an toàn gởi ngang qua bất kỳ network nào.

Sự kết nối

Sau một sự kiểm tra tính xác thực thành công trên tầng transport SSH, nhiều kênh được mở sự kết nối đa thành phần đơn lẻ giữa hai hệ thống. Mỗi kênh này thực hiện sự trao đổi thông tin cho những phần khác nhau như thông tin X11 forward hoặc bất kỳ dịch vụ riêng lẻ khác nhằm vào sử dụng sự kết nối SSH.

Cả hai client và server có thể tạo một kênh mới, với mỗi kênh được gán một số khác nhau. Khi một phía cố gắng để mở một kênh mới, số kênh của phía đó sẽ được gởi cùng với yêu cầu. Thông tin này được lưu trữ bởi những phía khác và được sử dụng trực tiếp một kiểu quen thuộc của sự trao đổi thông tin của dịch vụ tới kênh đó. Việc này được làm để cho những kiểu khác nhau của session sẽ không tác động tới những cái khác và các kênh có thể đóng không phá vỡ sự kết nối SSH chính giữa hai hệ thống.

Cài đặt và cấu hình OpenSSH

Như chúng ta đã biết OpenSSH thì miễn phí khi sử dụng nó, nó thay thế telnet, ftp, rlogin, rsh và rcp với những công cụ mã hoá và bảo mật. Sử dụng OpenSSH làm cho việc trao đổi thông tin trong mạng trở nên an toàn hơn, một nguyên nhân khác để sử dụng OpenSSH là nó tự động forward biến DISPLAY tới những máy client, ví dụ như bạn đang chạy hệ thống X windows trên máy nội bộ của bạn và bạn login vào một máy từ xa sử dụng lệnh ssh. Khi bạn thực thi một chương trình trên máy từ xa mà chương trình này yêu cầu X, nó sẽ được trình bày trên máy nội bộ của bạn. Đây là một tiện lợi trong việc quản lý hệ thống khi cần có giao diện đồ hoạ.

Hướng dẫn cài đặt

- Các lệnh tương thích với unix.
- Đường dẫn nguồn là /var/tmp (các đường dẫn khác cũng có thể được).
- Sự cài đặt đã được kiểm tra trên Red Hat Linux 7.1 và 7.2.
- Tất cả các bước cài đặt sẽ được thực hiện với supper-user "root".
- Phiên bản openssh là **openssh-server-2.9p2-7.**

Trước hết hãy tạo danh sách các tập tin trên hệ thống trước và sau khi bạn cài đặt Openssh, sau đó so sánh chúng dùng 'diff' để tìm ra các tập tin nào là được đặt ở đâu. Đơn giản chạy lệnh 'find /* > Openssh1' trước và 'find /* > Openssh2' sau khi bạn cài đặt phần mềm, và sử dụng lệnh 'diff Openssh1 Openssh2 > Openssh-Installed' để lấy một danh sách các thay đổi nào đó.

Cài đặt OpenSSH

Openssh được cung cấp như một gói dữ liệu (package) RPM trong Red Hat Linux 7.2. Nếu bạn hoàn thành một tiến trình cài đặt theo ý muốn và cài đặt mọi thứ trên hệ thống RadHat Linux của bạn thì OpenSSH sẽ được cài đặt sẵn trên hệ thống của bạn. Nếu không bạn cần download nó ở một website đáng tin cậy hoặc

cài đặt chúng từ CD Red Hat của bạn, trong CD này có package openssh-server-2.9p2-7.i386.rpm. Bạn cũng có thể download các tập tin nguồn và biên dịch chúng. Tập tin mã nguồn thì tiện lợi bởi vì chúng cho phép bạn hiệu chỉnh một vài tùy chọn cho việc biên dịch mà không được đưa vào trong phiên bản nhị phân. Như đã nói ở trên OpenSSH cần có OpenSSL được cài đặt trước, sau đó mới cài đặt OpenSSH. Để kiểm tra việc OpenSSL có cài đặt sẵn trên máy hay không, thực hiện các bước sau:

Hoàn thành những bước sau để đảm bảo rằng OpenSSL được cài đặt trên hệ thống Red Hat Linux của bạn.

- Để bảo đảm rằng OpenSSL RPM được cài đặt trên hệ thống của bạn, gõ vào dòng lệnh sau:

  ```
  rpm –qa | grep openssl
  ```

- Bạn sẽ nhận sự trả lời như sau nếu OpenSSL được cài đặt:

  ```
  openssl-0.9.6b-8
  ```

Nếu bạn nhận sự trả lời này thì bạn tin tưởng rằng openssl-0.9.6b-8 đã được cài đặt sẵn trong hệ thống của bạn.

- Nếu bạn không nhận message trả lời như trên, truy cập tới website www.Red Hat.com và nhấp chuột vào clink Download. Thực hiện một sự tìm kiếm RPM bằng cách gõ vào openssl trong field Find Latest RPMs By Keyword. Nhấp nút Search thì phiên bản mới nhất RPM sẽ xuất hiện. Bạn có thể nhận được package openssl-0.9.6b-8.i386.rpm trong CD cài đặt Red Hat Linux7.2

- Cài đặt OpenSSL từ package dạng RPM bằng cách dùng lệnh sau:

  ```
  rpm –i openssl-0.9.6b-8.i386.rpm
  ```

Bây giờ OpenSSL đã được cài đặt trên hệ thống của bạn và bạn có thể cài đặt OpenSSH từ đây. Tương tự như OpenSSL, để kiểm tra việc OpenSSH có cài đặt sẵn trên máy hay không thực hiện các bước sau:

Hoàn thành những bước sau để đảm bảo rằng OpenSSH được cài đặt trên hệ thống Red Hat Linux của bạn.

- Để bảo đảm rằng OpenSSH RPM được cài đặt trên hệ thống của bạn, gõ vào dòng lệnh sau:

  ```
  rpm –qa | grep openssh
  ```

- Bạn sẽ nhận sự trả lời như sau nếu OpenSSH được cài đặt:

  ```
  openssh-server-2.9p2-7
  ```

Nếu bạn nhận sự trả lời này thì bạn tin tưởng rằng openssh-server-2.9p2-7 đã được cài đặt sẵn trong hệ thống của bạn.

- Nếu bạn không nhận message trả lời như trên, truy cập tới website www.Red Hat.com và nhấp chuột vào clink Download. Thực hiện một sự tìm kiếm RPM bằng cách gõ vào openssh trong field Find Latest RPMs By Keyword. Nhấp

nút Search thì phiên bản mới nhất RPM sẽ xuất hiện. Bạn có thể nhận được package openssh-server-2.9p2-7.i386.rpm trong CD cài đặt Red Hat Linux7.2

- Cài đặt OpenSSH từ package dạng RPM bằng cách dùng lệnh sau:

```
rpm –i openssh-server-2.9p2-7.i386.rpm
```

- Bạn cũng nên cài đặt openssh-client và các package openssl liên quan khác trong CD cài đặt RH Linux7.2 vào trong hệ thống của bạn vì có thể bạn dùng nó như một openssh client để kết nối tới những hệ thống khác. Bạn thực hiện việc cài đặt openssh-client với lệnh như sau:

```
rpm –i openssh-clients-2.9p2-7.i386.rpm
```

Cấu hình OpenSSH server

Một khi OpenSSH được cài đặt, bạn phải cấu hình openssh từ tập tin /etc/ssh/sshd_config, tập tin cấu hình mặc định được cài sẽ thích đáng cho hầu hết các mục đích. Nếu bạn muốn cấu hình openssh theo những con đường khác không được cung cấp bởi mặc định sshd_config, đọc những tài liệu hướng dẫn sshd cho danh sách các từ khoá được định nghĩa trong tập tin cấu hình.

Khởi động dịch vụ OpenSSH sử dụng lệnh "/sbin/service sshd start". Để dừng dịch vụ OpenSSH server, sử dụng lệnh "/sbin/service sshd stop". Nếu bạn muốn khởi động openssh một cách tự động lúc khởi động máy có thể thêm dòng lệnh trên (/sbin/service sshd start) vào cuối tập tin "/etc/rc.d/rc.local". Nếu máy bạn có dùng giao diện đồ họa (graphic) bạn cũng có thể làm công việc này thông qua phần cấu hình dịch vụ trong Linux 7.2 (setup system service).

Chú ý: *Nếu bạn hoàn thành một tiến trình cài đặt theo ý muốn và cài đặt mọi thứ trên hệ thống Red Hat Linux của bạn thì OpenSSH sẽ được cài đặt sẵn trên hệ thống của bạn và nó cũng sẽ tự động start khi máy tính của bạn khởi động.*

Cấu hình một OpenSSH client

Như đã nói ở trên, để kết nối tới một OpenSSH server từ một máy client, bạn phải có gói dữ liệu (package) openssh-client và package openssh được cài đặt trên máy client. Việc kiểm tra và cài đặt openssh-client được thực hiện tương tự như đã làm với server đã được chỉ dẫn ở trên.

Sử dụng lệnh ssh

Lệnh ssh là một sự thay thế bảo mật cho các lệnh rlogin, rsh và telnet. Nó cho phép bạn login và thực thi các lệnh trên một máy ở xa.

Login vào một máy ở xa với ssh thì tương tự như sử dụng telnet. Ví dụ như login vào máy penguin.example.net gõ lệnh sau ở dấu nhắc lệnh shell:

```
ssh penguin.example.net
```

Thời gian đầu bạn chạy lệnh ssh để login tới máy từ xa, bạn sẽ nhận một message tương tự như sau:

```
The authenticity of host 'penguin.example.net' can't be established.
DSA key fingerprint is 94:68:3a:3a:bc:f3:9a:9b:01:5d:b3:07:38:e2:11:0c.
Are you sure you want to continue connecting (yes/no)?
```

Gõ Yes để tiếp tục, việc này sẽ thêm server vào danh sách các host được biết như trong message theo sau:

```
Warning: Permanently added 'penguin.example.net' (DSA) to the list of known hosts.
```

Kế tiếp bạn sẽ nhận một dấu nhắc hỏi mật khẩu của bạn cho máy từ xa. Sau khi gõ vào mật khẩu của bạn, bạn sẽ ở dấu nhắc shell trên máy từ xa. Nếu bạn sử dụng ssh không có bất kỳ những tùy chọn nào thì username mà bạn dùng để login vào máy local cũng được xem như username bạn dùng để login vào máy từ xa. Nếu bạn muốn chỉ rõ một username khác, dùng lệnh sau:

```
ssh -l username penguin.example.net
```

Bạn cũng có thể dùng cú pháp ssh:

```
username@penguin.example.net
```

Lệnh ssh có thể được dùng để thực thi một lệnh trên một máy ở xa không cần login tới để có một dấu nhắc shell. Cú pháp lệnh là "ssh hostname command", ví dụ nếu bạn cần thực thi lệnh "ls /usr/share/doc" trên máy ở xa penguin.example.net gõ lệnh như sau ở dấu nhắc shell.

```
ssh penguin.example.net ls /usr/share/doc
```

Sau khi gõ vào mật khẩu chính xác, nội dung của /usr/share/doc sẽ được trình bày và bạn sẽ quay trở lại dấu nhắc shell của bạn.

Sử dụng lệnh scp

Lệnh scp có thể được dùng để truyền tải những tập tin giữa các máy trên một sự kết nối bảo mật và được mã hoá. Nó tương tự như rcp.

Cú pháp chung để truyền tải một tập tin nội bộ tới một hệ thống từ xa là scp localfile username@to-hostname:/newfilename. Localfile chỉ rõ nguồn và username@to-hostname:/newfilename. Chỉ rõ đích.

Ví dụ truyền tải tập tin shadowman tới tài khoản của bạn trên penguin.example.net, gõ ở dấu nhắc shell lệnh sau (thay thế username bằng username của bạn trên penguin).

```
scp shadowman username@penguin.example.net:/home/username
```

Lệnh trên sẽ truyền tải tập tin nội bộ shadowman vào

/home/username/shadowman trên máy penguin.example.net

Cú pháp chung để truyền tải một tập tin trên máy ở xa vào hệ thống nội bộ của chúng ta là "scp username@tohost-name:/remotefile/*newlocalfile*" Ở đây remotefile chỉ rõ nguồn và newlocalfile chỉ rõ đích.

Nhiều tập tin có thể được chỉ định như các tập tin nguồn. Ví dụ truyền tải nội dung của thư mục /downloads vào một thư mục sẵn có được gọi là uploads trên máy từ xa penguin.example.net gõ lệnh sau ở dấu nhắc shell

```
scp /downloads/* username@penguin.example.net:/uploads/
```

Sử dụng lệnh sftp

Tiện ích sftp có thể được sử dụng để một session FTP tương tác. Nó tương tự như ftp ngoại trừ nó là sự kết nối được mã hoá, bảo mật. Cú pháp chung là sftp username@hostname.com. Một khi đã kiểm tra xong tính xác thực, bạn có thể sử dụng các lệnh giống như sử dụng với FTP. Tham khảo trang hướng dẫn sftp cho danh sách các lệnh này. Để đọc trang này thực thi lệnh man sftp ở dấu nhắc shell trên máy bạn. Tiện ích sftp chỉ sẵn có trong OpenSSH phiên bản 2.5.0p1 hoặc cao hơn.

Sự sinh ra cặp khoá

Nếu không muốn gõ vào mật khẩu của bạn, mỗi lần bạn sử dụng ssh, scp hoặc sftp để truy cập vào hệ thống từ xa, bạn có thể tạo ra một cặp khóa cho việc kiểm tra tính xác thực.

Chú ý: *Các khoá phải được tạo ra cho mỗi user. Để tạo ra khóa cho một user muốn kết nối tới các máy ở xa. Bạn làm theo những bước như được trình bày dưới đây, nếu bạn hoàn thành những bước sau với root thì chỉ root có thể dùng những khoá này.*

Tạo ra một cặp khoá DSA

Sử dụng những bước sau để tạo ra một cặp khoá DSA. DSA được sử dụng bởi giao thức (protocol) SSHv2.

1. Để tạo ra một cặp khoá DSA làm việc với giao thức SSHv2, gõ những lệnh sau ở dấu nhắc shell:
   ssh-keygen –t dsa
   Chấp nhận vị trí tập tin mặc định ~/.ssh/id_dsa. Gõ vào passphrase khác so với password tài khoản của bạn trên máy và xác nhận lại nó bằng cách gõ lại.
   Passphrase là gì? Một passphrase là một chuỗi của những từ và những ký tự để kiểm tra tính xác thực một user. Passphrase khác hơn password là bạn có thể sử dụng những khoảng trắng hoặc dấu tab trong passphrase. Passphrase thì thường dài hơn password vì chúng thường là những cụm từ thay cho chỉ một từ.
2. Thay đổi các permission của thư mục của bạn .ssh sử dụng lệnh chmod 755 ~/.ssh
3. Sao chép nội dung của ~/.ssh/id_dsa.pub vào thư mục ~/.ssh/authorized_keys2 trên máy mà bạn kết nối. Nếu trường hợp ~/.ssh/authorized_keys2 không có sẵn bạn có thể copy ~/.ssh/id_dsa.pub vào tập tin ~/.ssh/authorized_keys2 trên máy khác.

4. Nếu bạn đang chạy GNOME tham khảo phần trình bày về "Cấu hình ssh-agent với GNOME)

Tạo một cặp khoá RSA cho phiên bản 1.3 và 1.5

Sử dụng những bước sau tạo ra một cặp khoá RSA để sử dụng cho phiên bản 1 của protocol SSH. Nếu bạn chỉ kết nối giữa những máy RH Linux7.2 bạn không cần một cặp khoá RSA.

1. Để tạo ra một cặp khoá RSA (cho SSH v1.3 và 1.5) gõ vào những lệnh sau ở dấu nhắc lệnh:
   ssh-keygen
   Chấp nhận vị trí tập tin mặc định (~/.ssh/identity). Gõ vào một passphrase khác so với password tài khoản của bạn trên máy và xác nhận lại nó bằng cách gõ lại.
2. Thay đổi permission của thư mục .ssh và các khoá của bạn với các lệnh chmod 755 ~/.ssh và chmod 644 ~/.ssh/identity.pub.
3. Sao chép nội dung của ~/.ssh/identity.pub vào tập tin ~/.ssh/authorized_keys trên máy mà bạn kết nối. Nếu trường hợp ~/.ssh/authorized_keys không có sẵn bạn có thể copy ~/.ssh/identity.pub vào tập tin ~/.ssh/authorized_keys trên máy ở xa.
4. Nếu bạn đang chạy GNOME tham khảo phần trình bày về "Cấu hình ssh-agent với GNOME").

Cấu hình ssh-agent với GNOME

Tiện ích ssh-agent có thể được sử dụng để lưu passphrase của bạn để bạn không phải gõ nó mỗi lần bạn khởi tạo một sự kết nối ssh hoặc scp. Nếu bạn sử dụng GNOME, tiện ích openssh-askpass-gnome có thể được sử dụng để nhắc nhở bạn gõ passphrase của bạn khi bạn login vào GNOME và lưu nó cho đến khi bạn logout khỏi GNOME. Bạn sẽ không phải gõ vào password hoặc passphrase của bạn cho bất kỳ sự kết nối ssh hoặc scp được làm trong suốt session GNOME đó.

Để lưu passphrase của bạn trong suốt session GNOME của bạn, thực hiện theo những bước sau:

1. Bạn cần phải có package openssh-askpass-gnome được cài đặt. Bạn có thể dùng lệnh rpm -q openssh-askpass-gnome để xác định rõ nó đã được cài đặt hay chưa. Nếu nó chưa được cài đặt thì install nó từ RED HAT CD của bạn hoặc các website có liên quan khác.
2. Nếu bạn không có tập tin ~/.Xclients bạn có thể chạy switchdesk để tạo nó. Trong tập tin ~/.Xclients của bạn hiệu chỉnh những dòng sau:
   Thay đổi:
   ```
   exec $HOME/.Xclients-default
   ```
   Trở thành:
   ```
   exec /usr/bin/ssh-agent $HOME/.Xclients-default
   ```
3. Mở GNOME Control Center bằng cách chạy lệnh "/usr/sbin/gnomecc" từ terminal trong môi trường graphic của bạn. Sau khi chương trình Gnome

Control Certer đã xuất hiện bạn hãy làm theo các bước như sau: Đi tới phần Session=> Startup Programs. Click Add và gõ vào /usr/bin/ssh-add trong vùng Startup Command. Thiết đặt một độ ưu tiên tới một số cao hơn bất kỳ những dòng lệnh sẵn có để đảm bảo rằng nó được thực thi sau cùng. Một số của quyền ưu tiên tốt cho ssh-add là 70 hoặc cao hơn. Click OK để lưu tiến trình cài đặt của bạn và thoát khỏi GNOME Control Center.

Chú ý: *Mặc định trên Red Hat Linux7.2 bạn không thể chạy Gnome Control Center bằng cách click vào nút GNOME Programs => Settings => GNOME Control Center như các hệ thống Linux trước đây, tuy nhiên bạn cũng có thêm nó vào trong Setting như đã từng làm với các chương trình khác.*

4. Logout và sau đó login trở lại vào trong GNOME. Sau khi GNOME được start một dialog sẽ xuất hiện nhắc nhở bạn gõ vào passphrase của bạn. Gõ vào passphrase được yêu cầu. Nếu bạn có cả hai cặp khoá DSA và RSA được cấu hình bạn sẽ được nhắc nhở cho cả hai. Từ thời điểm này bạn sẽ không được nhắc nhở cho một password khi thực hiện các lệnh ssh, scp hoặc sftp.

Cấu hình ssh-agent

Ssh-agent có thể được sử dụng để lưu passphrase của bạn để cho bạn không phải gõ nó mỗi lần bạn làm một sự kết nối ssh và scp. Nếu bạn không chạy hệ thống X Windows theo những bước này từ dấu nhắc shell. Nếu bạn đang chạy GNOME nhưng bạn không muốn cấu hình nó để nhắc nhở bạn gõ vào passphrase khi bạn login. Thủ tục này sẽ chỉ làm việc trong một cửa sổ terminal chẳng hạn như một xterm. Nếu bạn đang chạy X nhưng không GNOME thủ tục này sẽ chỉ làm việc trong một cửa sổ terminal chẳng hạn như một xterm. Tuy nhiên passphrase của bạn sẽ chỉ được nhớ cho cửa sổ terminal đó, nó không là một cài đặt bao trùm.

1. Ở dấu nhắc shell, gõ những lệnh sau:

   ```
   exec /usr/bin/ssh-agent $SHELL
   ```

   Sau đó gõ lệnh:

   ```
   ssh-add
   ```

   Và gõ vào passphrase của bạn. Nếu bạn có cả hai cặp khoá DSA và RSA được cấu hình bạn sẽ được nhắc nhở cho cả hai.

2. Khi bạn logout, passphrase của bạn sẽ bị quên mất. Bạn phải thực thi hai lệnh trên mỗi lần bạn login hoặc mở một cửa sổ terminal.

Các tập tin được cài đặt

Các tập tin được cài đặt với OpenSSL

/lib/libcrypto.so.0.9.6b

/lib/libssl.so.0.9.6b

/usr/bin/openssl

/usr/share/doc/openssl-0.9.6b
/usr/share/doc/openssl-0.9.6b/CHANGES
/usr/share/doc/openssl-0.9.6b/FAQ
/usr/share/doc/openssl-0.9.6b/INSTALL
/usr/share/doc/openssl-0.9.6b/LICENSE
/usr/share/doc/openssl-0.9.6b/NEWS
/usr/share/doc/openssl-0.9.6b/README
/usr/share/doc/openssl-0.9.6b/c-indentation.el
/usr/share/doc/openssl-0.9.6b/openssl.txt
/usr/share/doc/openssl-0.9.6b/openssl_button.gif
/usr/share/doc/openssl-0.9.6b/openssl_button.html
/usr/share/doc/openssl-0.9.6b/ssleay.txt
/usr/share/man/man1/CA.pl.1ssl.gz
/usr/share/man/man1/asn1parse.1ssl.gz
/usr/share/man/man1/ca.1ssl.gz
/usr/share/man/man1/ciphers.1ssl.gz
/usr/share/man/man1/crl.1ssl.gz
/usr/share/man/man1/crl2pkcs7.1ssl.gz
/usr/share/man/man1/dgst.1ssl.gz
/usr/share/man/man1/dhparam.1ssl.gz
/usr/share/man/man1/dsa.1ssl.gz
/usr/share/man/man1/dsaparam.1ssl.gz
/usr/share/man/man1/enc.1ssl.gz
/usr/share/man/man1/gendsa.1ssl.gz
/usr/share/man/man1/genrsa.1ssl.gz
/usr/share/man/man1/nseq.1ssl.gz
/usr/share/man/man1/openssl.1ssl.gz
/usr/share/man/man1/pkcs12.1ssl.gz
/usr/share/man/man1/pkcs7.1ssl.gz
/usr/share/man/man1/pkcs8.1ssl.gz
/usr/share/man/man1/req.1ssl.gz
/usr/share/man/man1/rsa.1ssl.gz
/usr/share/man/man1/rsautl.1ssl.gz
/usr/share/man/man1/s_client.1ssl.gz

/usr/share/man/man1/s_server.1ssl.gz

/usr/share/man/man1/sess_id.1ssl.gz

/usr/share/man/man1/smime.1ssl.gz

/usr/share/man/man1/speed.1ssl.gz

/usr/share/man/man1/spkac.1ssl.gz

/usr/share/man/man1/sslpasswd.1ssl.gz

/usr/share/man/man1/sslrand.1ssl.gz

/usr/share/man/man1/verify.1ssl.gz

/usr/share/man/man1/version.1ssl.gz

/usr/share/man/man1/x509.1ssl.gz

/usr/share/man/man5/config.5ssl.gz

/usr/share/man/man7/des_modes.7ssl.gz

/usr/share/ssl

/usr/share/ssl/cert.pem

/usr/share/ssl/certs

/usr/share/ssl/certs/Makefile

/usr/share/ssl/certs/ca-bundle.crt

/usr/share/ssl/certs/make-dummy-cert

/usr/share/ssl/lib

/usr/share/ssl/misc/CA

/usr/share/ssl/misc/c_hash

/usr/share/ssl/misc/c_info

/usr/share/ssl/misc/c_issuer

/usr/share/ssl/misc/c_name

/usr/share/ssl/openssl.cnf

/usr/share/ssl/private

Các tập tin được cài đặt với Openssh-server

/etc/pam.d/sshd

/etc/rc.d/init.d/sshd

/etc/ssh

/etc/ssh/sshd_config

/usr/libexec/openssh/sftp-server

/usr/sbin/sshd

/usr/share/man/man8/sftp-server.8.gz

/usr/share/man/man8/sshd.8.gz

Các tập tin được cài đặt với Openssh-client

/etc/ssh/ssh_config

/usr/bin/sftp

/usr/bin/slogin

/usr/bin/ssh

/usr/bin/ssh-add

/usr/bin/ssh-agent

/usr/bin/ssh-keyscan

/usr/share/man/man1/sftp.1.gz

/usr/share/man/man1/slogin.1.gz

/usr/share/man/man1/ssh-add.1.gz

/usr/share/man/man1/ssh-agent.1.gz

/usr/share/man/man1/ssh-keyscan.1.gz

/usr/share/man/man1/ssh.1.gz

Các tập tin được cài đặt với Openssh-askpass-gnome

/etc/profile.d/gnome-ssh-askpass.csh

/etc/profile.d/gnome-ssh-askpass.sh

/usr/libexec/openssh/gnome-ssh-askpass

Chương 27:

CÀI ĐẶT VÀ SỬ DỤNG SAMBA

Trong Red Hat Linux7.2

Trong chương này chúng ta sẽ đề cập đến các vấn đề chính như sau:

- Cài đặt Samba-2.x.
- Security = domain trong samba2.x.
- Làm thế nào để cấu hình samba-2.2 như một Primary Domain Controller (PDC).

Tổng quan

Các phiên bản của samba trước phiên bản 2.2 rất khó hoặc đôi khi không được khi làm việc như một Windows NT 4.0 Primary Domain Controller (PDC). Bắt đầu với Samba 2.2.0 chúng ta có thể yên tâm là Samba chính thức hỗ trợ kiểu WinNT 4.0 domain logon từ các client Window NT4.0 (SP6) và Windows 2000 (SP1). Phần này sẽ cung cấp cho bạn những bước cần thiết cho việc cấu hình Samba như một PDC, những bước này là cần thiết để Samba có tính năng PDC. Những tính năng sau sẽ làm việc trong samba2.2.

- Domain logon cho các client Windows NT 4.0/2000.
- Đặt một client Windows 9x trong mức bảo mật (level security).
- Thu được một danh sách của các user và các group từ một PDC tới các client Windows 9x/NT/2000.
- Cách đối xử hệ thống (system policies) kiểu Windows NT 4.0.

Cài đặt Samba

Có hai cách cài đặt samba trên Red Hat Linux7.2. Bạn có thể cài đặt samba từ tập tin samba-2.2.1a-4.i386.rpm có sẵn trong CD cài đặt Linux7.2 hoặc bạn cũng có thể cài đặt từ tập tin samba-2.2.1a-4.tar.gz, mỗi cách cài đặt sẽ có những ưu nhược điểm khác nhau như nếu bạn cài đặt từ tập tin samba-2.2.1a-4.i386.rpm thì bạn sẽ không cần thay đổi bất kỳ giá trị nào tập tin này và khi cài đặt thành công bạn sẽ có tập tin cấu hình smb.conf tại thư mục "/etc/samba/smb.conf". Nếu bạn cài đặt samba từ tập tin samba-2.2.1a-4.tar.gz thì bạn có thể tiến hành tương tự như đã làm trên RH Linux 6.2 mà tôi đã trình bày ở phần trước, với cách cài đặt này bạn có thể thay đổi vị trí cài đặt mặc định theo ý muốn của bạn tuy nhiên bạn sẽ phải chỉnh sửa một vài thông số trong tập tin Makefile và configure của samba. Phần này chỉ tập trung cho việc cài đặt và cấu hình samba từ tập tin samba-2.2.1a-4.i386.rpm.

Trong Linux 7.2 trước khi cài đặt bạn nên kiểm tra xem samba có cài đặt trên hệ thống của bạn hay chưa với lệnh như sau:

```
[root@deep /]# rpm –q samba
```

Nếu samba chưa được cài đặt trên hệ thống thì bạn tiến hành cài đặt theo những lệnh sau:

```
[root@deep /]# mount /mnt/cdrom
[root@deep /]# cd /mnt/cdrom/Red Hat/RPMS
[root@deep /]# rpm –ivh samba-2.2.1a-4.i386.rpm
```

Lệnh đầu tiên kết gắn ổ đĩa CD vào hệ thống, lệnh thứ hai chuyển vào thư mục chứa chương trình nguồn samba-2.2.1a-4.i386.rpm và lệnh thứ ba sẽ cài đặt package samba-2.2.1a-4.i386.rpm vào hệ thống của bạn. Sau khi cài đặt thành công samba vào hệ thống của bạn, tập tin cấu hình smb.conf sẽ nằm trong thư

mục /etc/samba/smb.conf và tất cả những gì bạn cần cấu hình về samba sẽ được thực hiện từ tập tin này.

Để hiểu rõ và thực thi việc cấu hình Samba làm việc như một PDC trên Red Hat Linux 7.2, tôi giả sử rằng bạn đã có kinh nghiệm trong việc cấu hình Samba trên các hệ thống Linux trước đây hoặc bạn đã đọc và thực hành phần cài đặt và cấu hình Samba trên Red Hat Linux 6.2 trong cùng cuốn sách này.

Thiết đặt security = domain trong Samba 2.x

Kết nối một NT domain với Samba 2.2

Trong thứ tự để kết nối một server Samba-2 tới một NT domain, đầu tiên bạn phải thêm tên NetBIOS của samba server vào NT domain trên PDC bằng cách dùng Server Manager for Domains. Việc này tạo một tài khoản máy trong domain(PDC) SAM. Lưu ý rằng bạn sẽ chỉ thêm Samba server như một "Windows NT Workstation hoặc server" không như một Primary hoặc Backup domain controller.

Giả sử rằng bạn có một server Samba với trên NetBIOS là SERV1 và kết nối vào một NT domain có tên DOM với PDC có tên NetBIOS là DOMPDC và hai backup domain controllers với tên lần lượt là DOMBDC1 và DOMBDC2.

Trong thứ tự để gia nhập domain, đầu tiên dừng tất cả các Samba daemons và chạy lệnh sau:

```
root # smbpasswd -j DOM -r DOMPDC
```

Khi chúng ta gia nhập domain DOM và PDC cho domain (chỉ máy có thể truy cập tới cơ sở dữ liệu SAM domain) là DOMPDC. Nếu việc này thành công bạn sẽ thấy message.

"smbpasswd: Joined domain DOM" trên màn hình của bạn.

Lệnh trên thông qua password tài khoản máy thay đổi giao thức (protocol) sau đó viết password tài khoản máy mới cho samba server này vào trong một tập tin trong cùng thư mục mà tập tin smbpasswd được lưu trữ. Vị trí của nó như sau:

```
/etc/samba/private
```

Trong samba 2.0.x tên tập tin trông giống như sau:

```
<NT DOMAIN NAME>.<Samba Server Name> .mac
```

Phần mở rộng .mac có nghĩa là tập tin mật khẩu tài khoản máy (machine account password file). Như vậy trong ví dụ trên tập tin này có thể có tên như sau:

```
DOM.SERV1.mac
```

Trong Samba 2.2 tập tin này được thay thế với một tập tin TDB (Trivial Database) được đặt tên secrets.tdb.

Tập tin này được tạo và được sở hữu bởi root và không thể đọc bởi các user khác. Nó là khóa bảo mật mức domain cho hệ thống của bạn và nên được xử lý cẩn thận như một tập tin shadow password.

Bây giờ trước khi khởi động lại Samba daemon, bạn phải hiệu chỉnh tập tin smb.conf để báo cho samba biết rằng lúc này nó đang sử dụng domain security.

Hiệu chỉnh tập tin smb.conf với lệnh (vi /etc/samba/smb.conf ", thay đổi hoặc thêm dòng "security = " trong phần [global] của tập tin smb.conf trở thành:

```
security = domain
```

Kế đến thay đổi dòng "workgroup" cũng trong phần [global] trở thành:

```
workgroup = DOM
```

(DOM là tên của domain chúng ta tham gia vào)

Bạn cũng phải có tham số "encrypt passwords" thiết đặt bằng Yes trong thứ tự cho những user của bạn kiểm tra tính xác thực tới NT PDC.

Cuối cùng thêm hoặc thay đổi dòng "password server =" trong phần [global] trở thành:

```
password server = DOMPDC DOMBDC1 DOMBDC2
```

Đây là những Backup và Primary domain controller samba sẽ cố gắng liên lạc theo thứ tự để kiểm tra tính xác thực các user. Samba sẽ thử liên lạc với mỗi server này theo thứ tự và bạn có thể sắp xếp lại danh sách này theo thứ tự thích hợp để chia sẻ công việc kiểm tra tính xác thực giữa các domain controller.

Một cách nữa, nếu bạn muốn smbd tự động chỉ rõ danh sách của các domain controller dùng cho việc kiểm tra tính xác thực bạn có thể thiết đặt tham số trên như sau:

```
password server = *
```

Phương pháp này được giới thiệu trong samba 2.0.6, cho phép samba dùng chính xác cùng kỹ thuật như NT đang dùng. Phương pháp này hoặc broadcast hoặc sử dụng một WINS database theo thứ tự để tìm các domain controller cho việc kiểm tra tính xác thực.

Samba và các domain Windows 2000

Nhiều người cũng đã hỏi về việc khả năng của samba tham gia trong một domain windows 2000. Samba 2.2 có thể làm việc như một server thành viên của của một domain windows 2000 trong các mode mixed và native.

Có nhiều sự nhầm lẫn giữa các trường hợp là yêu cầu một mode "mixed" Win2K DC và một khi host này có thể chuyển sang mode "native". Một mode "mixed" Win2k domain controller chỉ cần nếu Windows NT BDC cần phải sẵn có trong cùng domain. Bởi mặc định một Win2K DC trong mode "native" sẽ vẫn hỗ trợ NetBIOS và NTLMv1 cho việc kiểm tra tính xác thực của các client chẳng hạn như Windows 9x và NT4.0. Samba có những yêu cầu tương tự như một server WinNT 4.0.

Các bước thêm một host samba 2.2 vào domain Win2K thì tương tự như việc thêm một samba server vào Windows NT 4.0 domain chỉ ngoại trừ là dùng "Server Manager" từ NT4 đã được thay thế bởi "Active Directory Users and Computers".

Tại sao security = domain thì tốt hơn security=server?

Hiện tại bảo mật domain trong samba tự do, bạn phải tạo một user unix nội bộ để đại diện cho user gia nhập vào server của bạn. Điều này có ý nghĩa rằng nếu domain user DOM\fred gia nhập vào server samba với domain security của bạn thì sẽ cần một user unix nội bộ fred đại diện cho user đó trong hệ thống unix. Việc này thì tương tự mode Samba security "security = server" ở đó Samba có thể thông qua yêu cầu kiểm tra tính xác thực tới một WinNT trong cùng con đường như một Windows 95 hoặc 98.

Tiện lợi của bảo mật mức domain là sự kiểm tra tính xác thực trong mode này được đi xuống kiểm tra RPC channel, việc kiểm tra này tương tự như cách thức mà một NT server có thể làm. Điều này có ý nghĩa rằng các samba server bây giờ gia nhập trong domain trust relationship chính xác là cùng đường như các NT server làm. (ví dụ bạn có thể thêm các Samba server vào trong một domain resource và có sự kiểm tra tính xác thực được thông qua từ một resource domain PDC tới một account domain PDC.

Thêm nữa với "security = server" mọi Samba daemon trên một server phải giữ kết nối để kiểm tra tính xác thực tới server. Việc này có thể làm tiêu hao tài nguyên trên một NT server và là nguyên nhân nó chạy vượt quá những kết nối sẵn có. Với security = domain các tiến trình Samba kết nối tới PDC/BDC chỉ khi cần thiết để kiểm tra tính xác thực của user và sau đó cắt sự kết nối (comnection) vì thế bảo toàn được số lượng kết nối tới PDC (PDC connection resources).

Cuối cùng Samba làm việc theo cách đối xử như NT server kiểm tra tính xác thực từ một PDC có ý nghĩa như sự trả lời của việc kiểm tra tính xác thực. Samba server nhận thông tin nhận diện chẳng hạn như user SID (Security IDentifier), danh sách các group mà user này thuộc trong group đó..... Tất cả các thông tin này sẽ cho phép samba mở rộng đặc trưng này vào trong một kiểu (mode) đang được tiến hành bởi các phát triển phần mềm hiện tại và được gọi là appliance mode. Theo mode này không cần thiết phải có user unix nội bộ và samba sẽ sinh ra các unix uids và gids từ thông tin được kiểm trả phản hồi lại từ PDC khi một user được kiểm tra tính xác thực và khi đó chúng ta sẽ có một samba có khả năng plug and play trong môi trường NT domain.

Cấu hình Samba 2.2 như một Primary Domain Controller

Khả năng làm việc như một PDC của Samba là một bước cải tiến lớn của các nhà phát triển phần mềm Samba. Trước khi cấu hình Samba làm việc như một PDC, chúng ta cần chú ý những chức năng sau vì nó không được hỗ trợ trong phiên bản samba hiện tại:

- Việc trusts giữa các domain WinNT 4.0.
- Sự tái tạo (replication) SAM với các NT 4.0 domain controller ví dụ như Samba PDC và các Windows NT BDC.
- Tạo các user mới thông qua User Manager for Domains.

- Làm việc như một Windows 2000 Domain Controller.

Sự thực thi một Samba PDC cơ bản có thể chia thành hai bước chính như sau:

- Cấu hình một Samba PDC.
- Tạo các tài khoản tin tưởng máy (machine trust account) và kết nối các client vào domain.

Ngoài ra cũng có những chi tiết không quan trọng khác như user profile, system policies, ...Tuy nhiên chúng không đặc biệt cần thiết cho một Samba PDC, chúng thường liên quan tới những khái niệm trong liên kết các Windows NT.

Cấu hình Samba domain controller

Bước đầu tiên để tạo một Samba làm việc như một PDC là hiểu các tham số cần thiết trong tập tin smb.conf. Phần này sẽ không diễn giải lại những tham số trong tập tin smb.conf, bạn có thể tham khảo ở phần trước (cài đặt samba trên linux 6.2) hoặc chạy lệnh "man smb.conf" trong cửa sổ linux của bạn.

Sau đây là một ví dụ về tập tin smb.conf để samba làm việc như một PDC. Tất nhiên những tham số khác bạn có thể thêm vào tập tin này cho phù hợp với yêu cầu của bạn.

```
[global]
    ; Basic server settings
    netbios name = POGO
    workgroup = NARNIA

    ; we should act as the domain and local master browser
    os level = 64
    preferred master = yes
    domain master = yes
    local master = yes

    ; security settings (must user security = user)
    security = domain

    ; encrypted passwords are a requirement for a PDC
    encrypt passwords = yes

    ; support domain logons
    domain logons = yes

    ; where to store user profiles?
    logon path = \\%N\profiles\%u
```

```
    ; where is a user's home directory and where should it
    ; be mounted at?
    logon drive = H:
    logon home = \\homeserver\%u

    ; specify a generic logon script for all users
    ; this is a relative **DOS** path to the [netlogon] share
    logon script = logon.cmd

; necessary share for domain controller
[netlogon]
    path = /etc/samba/netlogon
    writeable = no
    write list = ntadmin


; share for storing user profiles
[profiles]
    path = /export/smb/ntprofile
    writeable = yes
    create mask = 0600
    directory mask = 0700
```

Những điểm cần lưu ý cho cấu hình trên là:

- Sự mã hóa password phải có hiệu lực. Phần sau cùng trong chương này sẽ nói về việc này.
- Server phải hỗ trợ domain logon và một [netlogon] share
- Server phải là domain master browser trong thứ tự cho các client đang chạy windows chỉ rõ server như một DC.

Tạo các Machine Trust Account và kết nối các client vào domain.

Một machine trust account là một tài khoản sử dụng samba được nhìn nhận bằng một máy tính. Mật khẩu tài khoản (account password) có tác dụng như một khóa bảo mật cho việc bảo mật sự trao đổi thông tin với Domain Controller. Đây là một đặc trưng bảo mật nhằm tránh những máy có cùng tên NetBIOS không được phép gia nhập vào domain và dành được quyền truy cập tới những tài nguyên bên trong domain.

Trên một Windows NT PDC, các machine trust account này được lưu trữ trong registry. Trong khi một Samba PDC lưu giữ những tài khoản này trong tập tin smbpasswd. Tuy nhiên chỉ những trust account hiện hữu là sử dụng thuật toán băm mật khẩu NT (NT password hash).

Bởi vì Samba yêu cầu những tài khoản máy (machine account) chiếm hữu một UNIX uid từ một Windows NT SID (Security IDentifier) có thể được sinh ra, tất cả các account này phải có tên trong các tập tin /etc/passwd và /etc/samba/smbpasswd. Phiên bản samba tương lai có thể sẽ chỉ cần tạo trong /etc/passwd.

Có hai phương pháp để tạo các machine trust accounts.

- Tạo bằng tay trước khi gia nhập client vào domain. Trong trường hợp này mật khẩu (password) được thiết đặt với một giá trị chúng ta biết được. Tên NetBIOS của các máy là chữ thường.
- Tạo account ở thời điểm gia nhập domain. Trong trường hợp này phần session key của việc quản trị account được dùng để gia nhập client vào domain có tác dụng như một khóa mã hóa cho việc thiết đặt password với một giá trị bất kỳ (random value). Đây là phương pháp được đề nghị.

Tạo machine trust account bằng tay

Bước đầu tiên trong việc tạo một machine trust account bằng tay là tạo một mục (entry) cho machine trong tập tin /etc/passwd. Việc này có thể làm bằng cách dùng lệnh useradd mà chúng ta hay dùng để tạo các account unix. Dưới đây là một ví dụ cho trường hợp này.

```
root# /usr/sbin/useradd -g 100 -d /dev/null -c machine_nickname \
 -m -s /bin/false machine_name$
```

Tập tin /etc/passwd sẽ liệt kê tên máy với một dấu $ được gắn vào, sẽ không có một password, sẽ có một null shell và không home directory. Ví dụ một máy được gọi là "doppy" có thể có một dòng trong tập tin /etc/passwd giống như sau:

```
doppy$:x:505:100:machine_nickname:/dev/null:/bin/false
```

Ở đây machine_nickname có thể là bất kỳ tên diễn giải nào cho PC. Ví dụ BasementComputer. Machine_name tuyệt đối phải là tên NetBIOS của PC được thêm vào domain. Dấu $ phải được gắn vào tên NetBIOS của PC hoặc samba sẽ không nhận ra đây như là một machine account.

Bây giờ Unix account đã được tạo, bước kế tiếp là tạo smbpasswd entry cho máy chứa đựng trust account password ban đầu đã biết. Việc này có thể làm với lệnh smbpasswd như sau:

```
root# smbpasswd -a -m machine_name
```

Ở đây machine_name là tên NetBIOS của máy.

Gia nhập client vào domain ngay lập tức

Bình thường việc tạo một machine trust account dùng phương pháp này thì tương đương với việc tạo một machine account trên một Windows NT PDC sử dụng "Server Manager". Thời gian từ lúc account được tạo đến thời gian client được gia nhập domain và thay đổi password thì domain của bạn dễ bị tấn công bởi một ai

đó dùng một máy với cùng tên NetBIOS vì PDC đã tin tưởng vào tên của máy vừa tạo, vì đây là nguyên nhân cho việc bảo mật nên bạn cần gia nhập client ngay lập tức khi vừa tạo xong machine account trên PDC.

Tạo machine trust account lúc gia nhập domain

Phương pháp thứ hai này dùng để tạo machine trust account khi client cần ở thời gian gia nhập domain. Bạn sẽ cần đưa vào một giá trị cho tham số toàn cục "add user script" trong phần cài đặt [global]. Dưới đây là giá trị bạn phải gõ vào cho tham số này. Tham số này chỉ sử dụng khi bạn thiết đặt security = server hoặc security = domain, để biết thêm chi tiết bạn hãy chạy lệnh (man smb.conf) trong terminal của bạn.

```
add user script = /usr/sbin/useradd -d /dev/null -g 100 -s /bin/false -M %u
```

Trong samba2.2 chỉ tài khoản (account) root có thể được dùng để tạo machine account giống như trên. Vì thế yêu cầu tạo một entry cho smbpasswd cho root. Password nên được thiết đặt khác so với password trong /etc/passwd cũng vì nguyên nhân bảo mật.

Các tập tin được cài đặt

/etc/logrotate.d/samba

/etc/pam.d/samba

/etc/rc.d/init.d/smb

/etc/samba/smbusers

/etc/sysconfig/samba

/usr/bin/mksmbpasswd.sh

/usr/bin/smbadduser

/usr/bin/smbstatus

/usr/sbin/nmbd

/usr/sbin/smbd

/usr/share/doc/samba-2.2.1a

/usr/share/doc/samba-2.2.1a/COPYING

/usr/share/doc/samba-2.2.1a/Manifest

/usr/share/doc/samba-2.2.1a/README

/usr/share/doc/samba-2.2.1a/Read-Manifest-Now

/usr/share/doc/samba-2.2.1a/Roadmap

/usr/share/doc/samba-2.2.1a/WHATSNEW.txt

/usr/share/doc/samba-2.2.1a/docs

/usr/share/doc/samba-2.2.1a/docs/NT4-Locking.reg

/usr/share/doc/samba-2.2.1a/docs/NT4_PlainPassword.reg

/usr/share/doc/samba-2.2.1a/docs/OID

/usr/share/doc/samba-2.2.1a/docs/OID/allocated-arcs.txt
/usr/share/doc/samba-2.2.1a/docs/OID/samba-oid.mail
/usr/share/doc/samba-2.2.1a/docs/README.pam_smbpass
/usr/share/doc/samba-2.2.1a/docs/Samba-HOWTO-Collection.pdf
/usr/share/doc/samba-2.2.1a/docs/THANKS
/usr/share/doc/samba-2.2.1a/docs/Win2000_PlainPassword.reg
/usr/share/doc/samba-2.2.1a/docs/Win95_PlainPassword.reg
/usr/share/doc/samba-2.2.1a/docs/Win98_PlainPassword.reg
/usr/share/doc/samba-2.2.1a/docs/Win9X-CacheHandling.reg
/usr/share/doc/samba-2.2.1a/docs/WinME_PlainPassword.reg
/usr/share/doc/samba-2.2.1a/docs/WindowsTerminalServer.reg
/usr/share/doc/samba-2.2.1a/docs/announce
/usr/share/doc/samba-2.2.1a/docs/faq
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-Server-FAQ-1.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-Server-FAQ-2.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-Server-FAQ.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-meta-FAQ-1.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-meta-FAQ-2.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-meta-FAQ-3.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-meta-FAQ-4.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-meta-FAQ-5.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-meta-FAQ-6.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-meta-FAQ.html
/usr/share/doc/samba-2.2.1a/docs/faq/Samba-meta-FAQ.txt
/usr/share/doc/samba-2.2.1a/docs/faq/sambafaq-1.html
/usr/share/doc/samba-2.2.1a/docs/faq/sambafaq-2.html
/usr/share/doc/samba-2.2.1a/docs/faq/sambafaq-3.html
/usr/share/doc/samba-2.2.1a/docs/faq/sambafaq-4.html
/usr/share/doc/samba-2.2.1a/docs/faq/sambafaq-5.html
/usr/share/doc/samba-2.2.1a/docs/faq/sambafaq.html
/usr/share/doc/samba-2.2.1a/docs/faq/sambafaq.txt
/usr/share/doc/samba-2.2.1a/docs/history
/usr/share/doc/samba-2.2.1a/docs/htmldocs
/usr/share/doc/samba-2.2.1a/docs/htmldocs/CVS-Access.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/DOMAIN_MEMBER.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/ENCRYPTION.html

/usr/share/doc/samba-2.2.1a/docs/htmldocs/Integrating-with-Windows.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/NT_Security.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/OS2-Client-HOWTO.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/PAM-Authentication-And-Samba.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/Samba-HOWTO-Collection.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/Samba-PDC-HOWTO.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/UNIX_INSTALL.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/msdfs_setup.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/printer_driver2.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/samba-pdc-faq.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/samba-pdc-howto.html
/usr/share/doc/samba-2.2.1a/docs/htmldocs/wfw_slip.htm
/usr/share/doc/samba-2.2.1a/docs/htmldocs/winbind.html
/usr/share/doc/samba-2.2.1a/docs/samba.lsm
/usr/share/doc/samba-2.2.1a/docs/textdocs
/usr/share/doc/samba-2.2.1a/docs/textdocs/Application_Serving.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/BROWSING-Config.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/BROWSING.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/BUGS.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/DHCP-Server-Configuration.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/DIAGNOSIS.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/DNIX.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/Faxing.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/GOTCHAS.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/HINTS.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/INSTALL.sambatar
/usr/share/doc/samba-2.2.1a/docs/textdocs/Imprints.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/Macintosh_Clients.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/NetBIOS.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/PROFILES.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/Passwords.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/Printing.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/README.DCEDFS
/usr/share/doc/samba-2.2.1a/docs/textdocs/README.jis
/usr/share/doc/samba-2.2.1a/docs/textdocs/README.sambatar
/usr/share/doc/samba-2.2.1a/docs/textdocs/Recent-FAQs.txt

/usr/share/doc/samba-2.2.1a/docs/textdocs/RoutedNetworks.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/SCO.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/SMBTAR.notes
/usr/share/doc/samba-2.2.1a/docs/textdocs/SSLeay.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/Speed.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/Speed2.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/Tracing.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/UNIX-SMB.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/UNIX_SECURITY.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/Win95.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/WinNT.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/cifsntdomain.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/kurs.pdf
/usr/share/doc/samba-2.2.1a/docs/textdocs/kurs.tex
/usr/share/doc/samba-2.2.1a/docs/textdocs/logo.ps
/usr/share/doc/samba-2.2.1a/docs/textdocs/outdated
/usr/share/doc/samba-2.2.1a/docs/textdocs/outdated/NTDOMAIN.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/outdated/PRINTER_DRIVER.txt
/usr/share/doc/samba-2.2.1a/docs/textdocs/outdated/PROJECTS
/usr/share/doc/samba-2.2.1a/docs/textdocs/security_level.txt
/usr/share/doc/samba-2.2.1a/examples
/usr/share/doc/samba-2.2.1a/examples/LDAP
/usr/share/doc/samba-2.2.1a/examples/LDAP/README
/usr/share/doc/samba-2.2.1a/examples/LDAP/export2_smbpasswd.pl
/usr/share/doc/samba-2.2.1a/examples/LDAP/export_smbpasswd.pl
/usr/share/doc/samba-2.2.1a/examples/LDAP/import2_smbpasswd.pl
/usr/share/doc/samba-2.2.1a/examples/LDAP/import_smbpasswd.pl
/usr/share/doc/samba-2.2.1a/examples/LDAP/ldapchpasswd
/usr/share/doc/samba-2.2.1a/examples/LDAP/ldapsync.pl
/usr/share/doc/samba-2.2.1a/examples/LDAP/samba.schema
/usr/share/doc/samba-2.2.1a/examples/README
/usr/share/doc/samba-2.2.1a/examples/VFS
/usr/share/doc/samba-2.2.1a/examples/VFS/.cvsignore
/usr/share/doc/samba-2.2.1a/examples/VFS/Makefile
/usr/share/doc/samba-2.2.1a/examples/VFS/README
/usr/share/doc/samba-2.2.1a/examples/VFS/audit.c

/usr/share/doc/samba-2.2.1a/examples/VFS/block
/usr/share/doc/samba-2.2.1a/examples/VFS/block/Makefile
/usr/share/doc/samba-2.2.1a/examples/VFS/block/block.c
/usr/share/doc/samba-2.2.1a/examples/VFS/block/samba-block.conf
/usr/share/doc/samba-2.2.1a/examples/VFS/block/smb.conf
/usr/share/doc/samba-2.2.1a/examples/VFS/skel.c
/usr/share/doc/samba-2.2.1a/examples/appliance
/usr/share/doc/samba-2.2.1a/examples/appliance/Makefile
/usr/share/doc/samba-2.2.1a/examples/appliance/README
/usr/share/doc/samba-2.2.1a/examples/appliance/appliance.spec
/usr/share/doc/samba-2.2.1a/examples/appliance/build.sh
/usr/share/doc/samba-2.2.1a/examples/appliance/smb.conf-appliance
/usr/share/doc/samba-2.2.1a/examples/autofs
/usr/share/doc/samba-2.2.1a/examples/autofs/auto.a
/usr/share/doc/samba-2.2.1a/examples/dce-dfs
/usr/share/doc/samba-2.2.1a/examples/dce-dfs/README
/usr/share/doc/samba-2.2.1a/examples/dce-dfs/smb.conf
/usr/share/doc/samba-2.2.1a/examples/libsmbclient
/usr/share/doc/samba-2.2.1a/examples/libsmbclient/testsmbc.c
/usr/share/doc/samba-2.2.1a/examples/libsmbclient/tree.c
/usr/share/doc/samba-2.2.1a/examples/misc
/usr/share/doc/samba-2.2.1a/examples/misc/extra_smbstatus
/usr/share/doc/samba-2.2.1a/examples/misc/modify_samba_config.pl
/usr/share/doc/samba-2.2.1a/examples/misc/swat.pl
/usr/share/doc/samba-2.2.1a/examples/misc/wall.perl
/usr/share/doc/samba-2.2.1a/examples/printer-accounting
/usr/share/doc/samba-2.2.1a/examples/printer-accounting/README
/usr/share/doc/samba-2.2.1a/examples/printer-accounting/acct-all
/usr/share/doc/samba-2.2.1a/examples/printer-accounting/acct-sum
/usr/share/doc/samba-2.2.1a/examples/printer-accounting/hp5-redir
/usr/share/doc/samba-2.2.1a/examples/printer-accounting/lp-acct
/usr/share/doc/samba-2.2.1a/examples/printer-accounting/printcap
/usr/share/doc/samba-2.2.1a/examples/printing
/usr/share/doc/samba-2.2.1a/examples/printing/smbprint
/usr/share/doc/samba-2.2.1a/examples/printing/smbprint.newer
/usr/share/doc/samba-2.2.1a/examples/printing/smbprint.sysv

/usr/share/doc/samba-2.2.1a/examples/simple
/usr/share/doc/samba-2.2.1a/examples/simple/README
/usr/share/doc/samba-2.2.1a/examples/simple/smb.conf
/usr/share/doc/samba-2.2.1a/examples/smb.conf.default
/usr/share/doc/samba-2.2.1a/examples/svr4-startup
/usr/share/doc/samba-2.2.1a/examples/svr4-startup/README
/usr/share/doc/samba-2.2.1a/examples/svr4-startup/samba.server
/usr/share/doc/samba-2.2.1a/examples/thoralf
/usr/share/doc/samba-2.2.1a/examples/thoralf/smb.conf
/usr/share/doc/samba-2.2.1a/examples/tridge
/usr/share/doc/samba-2.2.1a/examples/tridge/README
/usr/share/doc/samba-2.2.1a/examples/tridge/smb.conf
/usr/share/doc/samba-2.2.1a/examples/tridge/smb.conf.WinNT
/usr/share/doc/samba-2.2.1a/examples/tridge/smb.conf.fjall
/usr/share/doc/samba-2.2.1a/examples/tridge/smb.conf.lapland
/usr/share/doc/samba-2.2.1a/examples/tridge/smb.conf.vittjokk
/usr/share/doc/samba-2.2.1a/examples/validchars
/usr/share/doc/samba-2.2.1a/examples/validchars/msdos70.out
/usr/share/doc/samba-2.2.1a/examples/validchars/nwdos70.out
/usr/share/doc/samba-2.2.1a/examples/validchars/readme
/usr/share/doc/samba-2.2.1a/examples/validchars/validchr.c
/usr/share/doc/samba-2.2.1a/examples/validchars/validchr.com
/usr/share/man/man1/smbstatus.1.gz
/usr/share/man/man5/smbpasswd.5.gz
/usr/share/man/man7/samba.7.gz
/usr/share/man/man8/nmbd.8.gz
/usr/share/man/man8/smbd.8.gz
/var/cache/samba
/var/log/samba
/var/spool/samba

PHẦN 7
CÁC PHỤ LỤC

Phụ lục A: Những lắt léo (Tweaks), mẹo (Tips) và công việc của người quản trị.

Phụ lục B: Các RFC (Obtaining Requests for Comments) đang tồn tại.

Phụ lục A:
NHỮNG LẮT LÉO (TWEAKS), MẸO (TIPS) VÀ CÔNG VIỆC CỦA NGƯỜI QUẢN TRỊ

Một số mẹo kỹ thuật (tip) trong phần này là được xác định rõ ra cho các hệ thống Linux. Nói chung hầu hết là có thể áp dụng được cho hệ điều hành UNIX.

1.0. Lệnh tiện ích "du"

Bạn có thể dùng tiện ích "du" để ước lượng việc sử dụng không gian của đĩa. Lấy ví dụ, để xác định các kích thước của các cây thư mục "/var/log" và "/home/" tính bằng megabyte, gõ vào lệnh sau:

```
[root@deep /]# du -sh /var/log /home
3.5M /var/log
350M /home
```

Chú ý: *Lệnh ở trên sẽ báo cáo kích thước dữ liệu thật sự của bạn. Và bây giờ bạn biết thêm ví dụ là "/home" thì đang dùng 350M bạn có thể chuyển vào bên trong nó và "du –sh *" để định vị nơi các tập tin lớn nhất ở đó.*

```
[root@deep /]# cd /home/
[root@deep /home]# du -sh *
343M admin
11k ftp
6.8M httpd
12k lost+found
6.0k named
6.0k smbclient
6.0k test
8.0k www
```

Ghi chú: *Bạn có thể thêm lệnh này vào crontab của bạn và như vậy mỗi ngày bạn lấy được danh sách các email có không gian đĩa như mong muốn, và bạn sẽ có thể theo dõi nó mà không phải log in.*

1.1. Tìm đường đi mà các packet được gửi từ máy của bạn đến máy host ở xa

Nếu bạn muốn tìm ra lộ trình mà các packet được gửi từ máy của bạn tới một máy host ở xa , đơn giản dùng lệnh sau:

```
[root@deep /]# traceroute www.Red Hat.com
traceroute to www.portal.Red Hat.com (206.132.41.202), 30 hops max, 38 byte packets
```

1 ppp005.108-253-207.mtl.mt.videotron.net (207.253.108.5) 98.584 ms 1519.806 ms 109.911 ms

2 fa5-1-0.rb02-piex.videotron.net (207.96.135.1) 149.888 ms 89.830 ms 109.914 ms

3 ia-tlpt-bb01-fec1.videotron.net (207.253.253.53) 149.896 ms 99.873 ms 139.930 ms

4 ia-cduc-bb02-ge2-0.videotron.net (207.253.253.61) 99.897 ms 169.863 ms 329.926 ms

5 if-4-1.core1.Montreal.Teleglobe.net (207.45.204.5) 409.895 ms 1469.882 ms 109.902 ms

6 if-1-1.core1.NewYork.Teleglobe.net (207.45.223.109) 189.920 ms 139.852 ms 109.939 ms

7 206.132.150.133 (206.132.150.133) 99.902 ms 99.724 ms 119.914 ms

8 pos1-0-2488M.wr2.CLE1.gblx.net (206.132.111.89) 189.899 ms 129.873 ms 129.934 ms

9 pos8-0-2488m.wr2.kcy1.globalcenter.net (206.132.111.82) 169.890 ms 179.884 ms 169.933 ms

10 206.132.114.77 (206.132.114.77) 199.890 ms 179.771 ms 169.928 ms

11 pos8-0-2488M.wr2.SFO1.gblx.net (206.132.110.110) 159.909 ms 199.959 ms 179.837 ms

12 pos1-0-2488M.cr1.SNV2.gblx.net (208.48.118.118) 179.885 ms 309.855 ms 299.937 ms

13 pos0-0-0-155M.hr2.SNV2.gblx.net (206.132.151.46) 329.905 ms 179.843 ms 169.936 ms

14 206.132.41.202 (206.132.41.202) 2229 906 ms 199.752 ms 309.927 ms

Trong đó www.RedHat.com là tên hay địa chỉ ip của host mà bạn muốn thử kiểm tra (trace).

1.2. Hiển thị số lần mà các trang web của bạn đã được truy cập:

Để hiển thị nhanh số lần mà trang web của bạn đã được truy cập dùng lệnh sau:

```
[root@deep /]# grep "GET / HTTP" /var/log/httpd/access_log | wc -l
467
```

1.3 Tắt bỏ hoàn toàn nhiều dịch vụ cùng lúc

Như là root, bạn có thể tắt bỏ hầu hết các dịch vụ (service) với lệnh sau:

```
[root@deep /]# killall httpd smbd nmbd slapd named
```

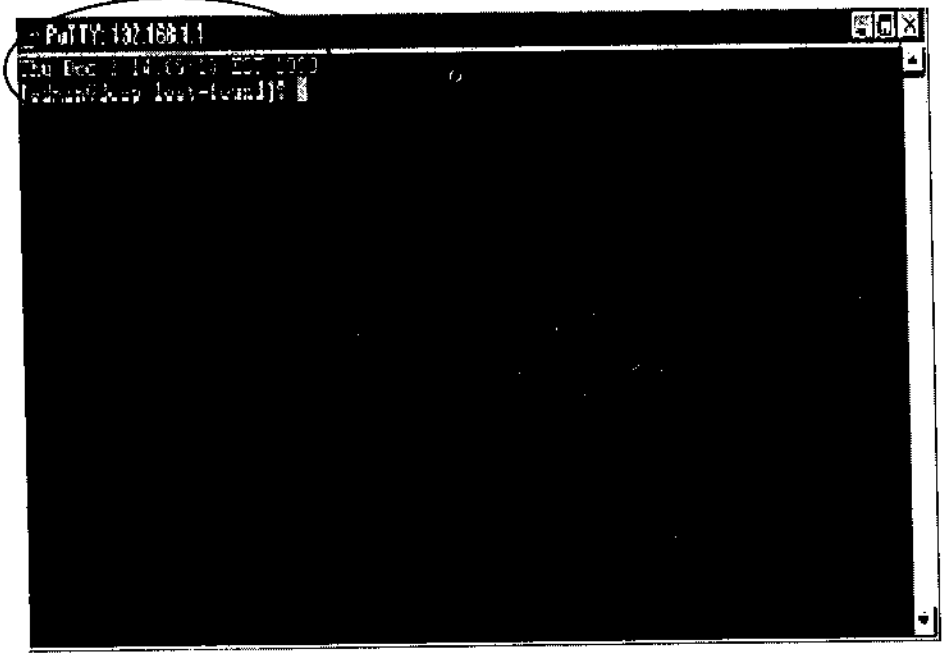
Lệnh trên sẽ tắt bỏ (shut down) Apache server, Samba services, LDAP server, và DNS server từng dịch vụ một.

1.4 Cần một đồng hồ ở phía trên trong cửa sổ terminal của bạn cho tất cả các người dùng?

Soạn thảo tập tin profile (vi /etc/profile) và thêm vào dòng sau:

```
PROMPT_COMMAND='echo -ne
"\0337\033[2;999r\033[1;1H\033[00;44m\033[K"`date`"\033[00m\0338"'
```

Kết quả sẽ như sau:



1.5. Bạn có muốn cài đặt "lsof" trên máy server?

Nếu không, cài đặt nó và thi hành "lsof -i". Việc này sẽ liệt kê ra các cổng nào mà bạn đã mở trên máy của bạn. Chương trình lsof là một công cụ rất hay, nó báo cho bạn biết các process nào đang lắng nghe trên một cổng cho sẵn nào đó.

```
[root@deep /]# lsof -i
COMMAND  PID  USER  FD  TYPE  DEVICE  SIZE  NODE  NAME
inetd    344  root  4u  IPv4  327           TCP   *:ssh (LISTEN)
sendmail 389  root  4u  IPv4  387           TCP   *:smtp (LISTEN)
smbd     450  root  5u  IPv4  452           TCP   deep.openna.com:
                                                  netbios-ssn (LISTEN)
nmbd     461  root  5u  IPv4  463           UDP   *:netbios-ns
nmbd     461  root  6u  IPv4  465           UDP   *:netbios-dgm
nmbd     461  root  8u  IPv4  468           UDP   deep.openna.com:
                                                  netbios-ns
nmbd     461  root  9u  IPv4  470           UDP   deep.openna.com:
                                                  netbios-dgm
named    2599 root  4u  IPv4  3095          UDP   *:32771
named    2599 root  20u IPv4  3091          UDP   localhost.localdomain:
                                                  domain
named    2599 root  21u IPv4  3092          TCP   localhost.localdomain:
                                                  domain (LISTEN)
named    2599 root  22u IPv4  3093          UDP   deep.openna.com:
                                                  domain
named    2599 root  23u IPv4  3094          TCP   deep.openna.com:
                                                  domain (LISTEN)
```

1.6. Thi hành các lệnh trên các máy server ở xa thông qua giao thức ssh không cần phải logging in

Lệnh ssh cũng có thể được dùng để chạy các lệnh trên các hệ thống ở xa không phải log in. Kết quả xuất ra của lệnh được hiển thị, và điều khiển trả về hệ thống cục bộ. Đây là một ví dụ sẽ hiển thị tất cả các user đang log in vào hệ thống máy ở xa.

```
[admin@deep /]$ ssh mail.openna.com who
admin@mail.openna.com's password:
root tty1 Dec 2 14:45
admin tty2 Dec 2 14:45
wahib pts/0 Dec 2 11:38
```

1.7. Làm đầy đủ tên tập tin

Tab Filename completion cho phép bạn gõ vào các phần riêng lẻ tên của một tập tin hay chương trình, và khi nhấn [TAB] nó sẽ làm đầy đủ tên tập tin cho bạn. Nếu có nhiều hơn một tập tin hay chương trình mà bắt đầu với những gì mà bạn gõ vào, nó sẽ kêu bip, và khi đó bạn nhấn [TAB] lần nữa nó sẽ liệt kê tất cả các tập tin mà bắt đầu với những gì mà bạn gõ vào lúc đầu.

1.8. Các ký tự đặc biệt

Bạn có thể thực hiện xong các công việc nhanh chóng mà bạn thực hiện thường xuyên bằng cách dùng các phím nóng – một hoặc nhiều phím mà bạn nhấn trên bàn phím để hoàn thành một công việc. Lấy ví dụ, các ký tự đặc biệt có thể được dùng trên Linux shell giống như sau:

Control-d: Nếu bạn đang trong shell và đánh control-d thì bạn log off.

Control-I: Nếu bạn đang trong shell và đánh control-I thì bạn xoá màn hình.

?: Đây là một ký tự đại diện cho một ký tự (wildchar). Nếu bạn chỉ rõ một cái gì đó tại dòng lệnh như "m?b" thì Linux sẽ tìm kiếm mob, mib, và mỗi ký tự khác/ hay số giữa a-z, 0-9.

: Đây là một wildchar có thể đại diện cho một số các ký tự. Nếu bạn chỉ rõ một "mi" nó sẽ dùng "mit", mim, miiii, miya và BẤT KỲ từ nào mà nó bắt đầu với "mi". Còn "m*l" có thể là mill, mull, và bất kỳ cái gì mà nó bắt đầu là một ký tự "m" và kết thúc với "l".

[] – Chỉ rõ một miền giá trị. Nếu như là m[o,u,i]m thì Linux sẽ nghĩ là: mim, mum, mom, còn nếu như là : m[a-d]m thì Linux sẽ nghĩ là: mam, mbm, mcm, mdm. Các ký tự đặc biệt này [],?, và * thường dùng với sao chép, xóa, và liệt kê thư mục.

Ghi chú: *Mọi thứ trong Linux đều được phân biệt chữ thường và chữ in. Điều này có nghĩa là "Bill" và "bill" là không giống nhau. Điều này cho phép rất nhiều tập tin có thể được lưu trữ. Bởi vì "Bill", "bill", "billL" , v.v..có thể là các tập tin khác nhau. Vì vậy, khi sử dụng [], bạn phải chỉ ra các ký tự in nếu có bất kỳ tập tin nào gắn với các ký tự in, dù có rất nhiều thứ là chữ thường trong Unix.*

Phụ lục B
Các RFC (Obtaining Requests for Comments) đang tồn tại

Các yêu cầu cho các chú thích (RFCs) là một tập hợp các tài liệu đang phát triển và được phát hành bởi Internet Engineering Task Force (IETF) ở trung tâm thông tin mạng (Network Informatio Center - NIC), nơi công bố các giao thức mới và thiết lập các chuẩn cho bộ giao thức Internet (Internet protocol suite). Mỗi tài liệu định nghĩa một khía cạnh của giao thức đối với Internet. Chúng tôi đã liệt kê bên dưới tất cả RFCs gắn liền với quyển sách này, và các phần mềm khác nhau được mô tả trong quyển sách này. RFCs là có sẵn ở web site sau: http://www.cis.ohio-state.edu/rfc/ hoặc một vài website có liên quan khác. Để biết chi tiết về từng RFC bạn nên đọc trên các tập tin RFC cụ thể

RFC706
On the Junk Mail Problem.

RFC733
Standard for the Format of ARPA Network Text Messages.

RFC768
User Datagram Protocol (UDP).

RFC791
Internet Protocol (IP).

RFC792
Internet Control Message Protocol (ICMP).

RFC793
Transmission Control Protocol (TCP).

RFC805
Computer Mail Meting Notes.

RFC821
Simple Mail Transfert Protocol (SMTP).

RFC822
Standard for the Format of ARPA Internet Text Massages.

RFC934

Proposed Standard for Message Encapsulation.

RFC950
IP Subnet Extention.

RFC959
File Transfer Protocol (FTP).

RFC976
UUCP Mail Interchange Format Standard.

RFC1034
Domain Names: Concepts and Facilities.

RFC1036
Standard for Interchange of USENET Message.

RFC1058
Routing Information Protocol (RIP).

RFC1112
Internet Group Multicast Protocol (IGMP).

RFC1122
Requirement for Internet Host—Communication Layers.

RFC1123
Requirements for Internet Host—Application and Support.

RFC1137
Mapping Between Full RFC 822 and RFC 822 with Restricted Encoding.

RFC1153
Digest Message Format.

RFC1155
Structure of Management Information (SMI).

RFC1157
Simple Network Management Protocol (SNMP).

RFC1176
Interactive Mail Access Protocol Version 2

RFC1274
The COSINE and Internet X.500 Schema.

RFC1275
Replication Requirements to provide an Internet Directory using X.500.

RFC1279
X.500 and Domains.

RFC1308
Executive Introduction to Directory Services Using the X.500 Protocol.

RFC1309
Technical Overview of Directory Services Using the X.500 Protocol.

RFC1310
The Internet Standards Process.

RFC1319
MD2 Message-Digest Algorithm.

RFC1320
MD4 Message-Digest Algorithm.

RFC1321
MD5 Message-Digest Algorithm.

RFC1343
User Agent Configuration Mechanism for Multimedia Mail Format Information.

RFC1344
Implications of MIME for Internet Mail Gateways.

RFC1345
Character Mnemonics and Character Sets.

RFC1421
Privacy Enhancement for Internet Electronic Mail: Part I—Message Encipherment and
authentication Procedures.

RFC1422
Privacy Enhancement for Internet Electronic Mail: Part II—Certificate-based key Management.

RFC1423
Privacy Enhancement for Internet Electronic Mail: Part III—Algorithms, modes, and identifiers
[Draft].

RFC1428
Transmition of Internet Mail from Just-Send-8 to 8bit-SMTP/MIME.

RFC1430
A Strategic Plan for Deploying an Internet X.500 Directory Service.

RFC1492
An Access Control Protocol, Sometimes Called TACACS.

RFC1495
Mapping Between X.400(1988)/ISO 10021 and RFC 822.

RFC1496
X.400 1988 to 1984 Downgrading.

RFC1505
Encoding Header Field for Internet Messages.

RFC1510
The Kerberos Network Authentication Service (V5).

RFC1519
Classless Inter-Domain Routing (CIDR) Assignment and Aggregation Strategy.
RFC1521
MIME (Multipurpose Internet Mail Extensions): Mechanisms for Specifying and Describing the
Format of Internet Message Bodies (MIME).

RFC1522
Representation of Non-ASCII Text in Internet Message Headers.

RFC1558
A String Representation of LDAP Search Filters.

RFC1566
Mail Monitoring MIB.

RFC1579
Firewall-Friendly FTP.

RFC1583
Open Shortest Path First Routing V2 (OSPF2).

RFC1617
Naming and Structuring Guidelines for X.500 Directory Pilots.

RFC1625
WAIS over Z39.50-1988.

RFC1631
The IP Network Address Translator (NAT).

RFC1652
SMTP Service Extentions for 8bit-MIMEtransport.

RFC1661
Point-to-Point Protocol (PPP).

RFC1711
Classifications in E-mail Routing.

RFC1725
Post Office Protocol, Version 3 (POP)3.

RFC1738
Uniform Resource Locators (URL).

RFC1739
A Primer on Internet and TCP/IP Tools.

RFC1777
Lightweight Directory Access Protocol.

RFC1778
The String Representation of Standard Attribute Syntaxes.

RFC1779

A String Representation of Distinguished Names.

RFC1781
Using the OSI Directory to Achieve User Friendly Naming.

RFC1796
Not All RFCs are Standards.

RFC1798
Connection-less Lightweight Directory Access Protocol.

RFC1823
The LDAP Application Program Interface.

RFC1830
SMTP Services Extentions for Transmission of Large and Binary MIME Messages.

RFC1844
Multimedia E-mail (MIME) User Agent checklist.

RFC1845
SMTP Service Extension for Checkpoint/Restart.

RFC1846
SMTP 521 Reply Code.

RFC1854
SMTP Service Extention for command pipelining.

RFC1855
Netiquette Guidelines.
RFC1864
The content-MD5 Header.

RFC1866
Hypertext Markup Language - 2.0.

RFC1869
SMTP Service Extensions.

RFC1870
SMTP Service Extension for Message Size Declaration.

RFC1872
The MIME Multipart/Related Content-type.

RFC1873
Message/External-Body Content-ID Access-type.

RFC1883
Internet Protocol, Version 6 (Ipv6) Specification.

RFC1884
IP Version 6 Addressing Atchitecture.

RFC1886
DNS Extentions to support IP version 6.

RFC1891
SMTP Service Extension for Delivery Status Notifications.

RFC1892
The Multipart/Report Content Type for the Reporting of Mail System Administrative Messages.

RFC1893
Enhanced Mail System Status Codes.

RFC1894
An Extensible Message Format for Delivery Status Notifications.

RFC1918
Address Allocation for Private Internets.

RFC1928
SOCKS Protocol Version 5.

RFC1929
Username/Password Authentication for SOCKS V5.

RFC1959
An LDAP URL Format.

RFC1960
A String Representation of LDAP Search Filters.

RFC1961
GSS-API Authentication Method for SOCKS Version 5.

RFC2003
IP Encapsulation within IP.

RFC2028
The Organizations Involved in the IETF Standards Process.

RFC2044
UTF-8, a transformation format of Unicode and ISO 10646.

RFC2060
Internet Message Access Protocol – Version 4rev1 (IMAP4).

RFC2104
HMAC: Keyed-Hashing for Message Authentication.

RFC2138
Remote Authentication Dial In User Service (RADIUS).

RFC2164
Use of an X.500/LDAP directory to support MIXER address mapping.

RFC2200
Internet Official Protocol Standards.

RFC2218
A Common Schema for the Internet White Pages Service.

RFC2247
Using Domains in LDAP/X.500 Distinguished Names.

RFC2251
Lightweight Directory Access Protocol (v3).

RFC2252
Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions.

RFC2253

Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names

RFC2254
The String Representation of LDAP Search Filters.

RFC2255
The LDAP URL Format.

RFC2256
A Summary of the X.500(96) User Schema for use with LDAPv3.

RFC2279
UTF-8, a transformation format of ISO 10646.

RFC2293
Representing Tables and Subtrees in the X.500 Directory.

RFC2294
Representing the O/R Address hierarchy in the X.500 Directory Information Tree.

RFC2305
A Simple Mode of Facsimile Using Internet Mail.

RFC2307
An Approach for Using LDAP as a Network Information Service.

RFC2313
PKCS 1: RSA Encryption Version 1-5.

RFC2314
PKCS 10: Certification Request Syntax Version 1-5.

RFC2315
PKCS 7: Cryptographic Message Syntax Version 1-5.

RFC2377
Naming Plan for Internet Directory-Enabled Applications.

BẢO MẬT VÀ TỐI ƯU TRONG RED HAT LINUX

Chịu trách nhiệm xuất bản:
NGUYỄN ĐÌNH THIÊM

Chịu trách nhiệm nội dung:
NGUYỄN BÁ NGỌC

Biên tập:
LÊ THANH DUY

Sửa bản in:
NGỌC AN

Thiết kế bìa:
NHỮ ĐÌNH NGOẠN

Tổng phát hành tại **NHÀ SÁCH MINH KHAI**
249 Nguyễn Thị Minh Khai – Q.1 – TP. HCM
ĐT: (08) 9250590 – 9250591 FAX: (08) 8331124
E-mail: mk.pub @ cinet.vnnews.com
Website: www.minhkhai.com.vn

In 1.000 cuốn khổ 15,5 x 23,5 tại XN In Machinco - 21 Bùi Thị Xuân, Q.I. Đóng xén tại DNTN Ngôi Sao – 57 Võ Thành Trang, Q.Tân Bình. Giấy phép xuất bản số: 19/624 XB-QLXB ngày 07/06/2002, mã số 28-5/2002.
In xong và nộp lưu chiểu tháng 10/2002.

Bảo mật và Tối ưu trong Red Hat Linux

- Hướng dẫn chi tiết làm gì, làm như thế nào, tại sao làm như thế cho ai muốn trở thành Nhà quản trị Linux và muốn cài đặt một máy Linux an toàn, tối ưu.
  - Cấu hình và xây dựng một Linux Kernel bảo mật và tối ưu.
  - Xây dựng web server hỗ trợ SSL
  - Xây dựng mạng riêng ảo VPN/IP Sec
  - Cấu hình một Samba server bảo mật
  - Xây dựng một Linux server với các công cụ giám sát, quản lý truy cập ...
- Hướng dẫn trên nền Linux 6.x, 7.x
- Đĩa mềm chứa tất cả các cấu hình chuẩn của các ứng dụng trên Linux

(Đĩa mềm kèm theo sách)



NHÀ SÁCH MINH KHAI
MK. PUB

Giá : 89.000đ

NHU DINH NGOAN DESIGN