

CONTAINER HÓA: TỔNG HỢP CÁC KHẢO SÁT VỀ HIỆN TRẠNG VÀ XU HƯỚNG ÁP DỤNG NỀN TẢNG CONTAINER TRONG DOANH NGHIỆP

● PHẠM NGỌC DUY - NGUYỄN THÀNH TUẤN ANH

TÓM TẮT:

Trong kỷ nguyên công nghiệp 4.0, các tổ chức và doanh nghiệp ở tất cả các lĩnh vực đều cần phải ứng dụng mạnh mẽ các giải pháp CNTT (công nghệ thông tin) tiên tiến để xây dựng và triển khai các hệ thống một cách nhanh chóng và hiệu quả. Việc ứng dụng nền tảng công nghệ container hóa kết hợp với các dịch vụ điện toán mây là một giải pháp xu hướng hiện nay, đặc biệt trong triển khai ứng dụng cho doanh nghiệp. Nhóm các bài viết về chủ đề "container hóa" sẽ trình bày tổng quan về công nghệ container cùng dịch vụ đám mây, xu hướng áp dụng giải pháp container của các tổ chức hiện nay, đồng thời đưa ra một số lợi thế và thách thức các doanh nghiệp có thể gặp phải khi triển khai, vận hành hệ thống với container. Bài viết này tập trung vào việc tổng hợp và phân tích một số kết quả khảo sát về hiện trạng ứng dụng công nghệ container trong doanh nghiệp, cũng như xu hướng phát triển của container trong thời gian tới.

Từ khóa: Container, container hóa, ứng dụng container trong doanh nghiệp, khảo sát.

1. Giới thiệu

Trong bài báo trước "Container hóa với điện toán đám mây: Giải pháp xu thế trong phát triển và triển khai ứng dụng doanh nghiệp", tác giả đã trình bày tổng quan về container và công nghệ container hóa, các lợi ích của việc áp dụng nền tảng container, cũng như mối quan hệ giữa container và các công nghệ tiên tiến hiện nay như ảo hóa, kiến trúc vi dịch vụ, điện toán đám mây. Container là một gói có khả năng thực thi độc lập, hạng nhẹ (lightweight), chứa mọi thứ mà chúng

cần để hoạt động: mã chương trình (code), runtime, các công cụ hệ thống, thư viện hệ thống (libraries), các gói phụ thuộc (dependencies) - bất cứ thứ gì có thể cài đặt trên một máy chủ. Bằng cách đóng gói và cô lập mọi thứ trong một container, điều này đảm bảo rằng các container sẽ luôn hoạt động giống như nhau bất kể môi trường nào mà nó đang chạy trong đó. Container loại bỏ sự phụ thuộc giữa các ứng dụng và lớp cơ sở hạ tầng căn bản, cung cấp những khả năng mới về tính di động và tính hiệu quả.

2. Docker và Kubernetes

2.1. Docker

2.1.1. Nền tảng container hóa Docker

Docker là một nền tảng mã nguồn mở để xây dựng, triển khai và quản lý các ứng dụng được container hóa. Docker cho phép các nhà phát triển có thể đóng gói các ứng dụng vào trong các container - các thành phần thực thi được tiêu chuẩn hóa, kết hợp mã nguồn ứng dụng với tất cả các thư viện hệ điều hành và gói phụ thuộc cần thiết để thực thi mã nguồn trong bất kỳ môi trường nào. Mặc dù các nhà phát triển có thể tạo các container mà không cần Docker, Docker giúp việc xây dựng, triển khai và quản lý các container dễ dàng hơn, đơn giản hơn và an toàn hơn. Về cơ bản, nó là một bộ công cụ cho phép các nhà phát triển có thể xây dựng, triển khai, chạy, cập nhật và dừng các container bằng cách sử dụng các lệnh đơn giản và sự tự động hóa để tiết kiệm công làm việc. Docker cũng đề cập đến Docker Inc. - công ty bán phiên bản thương mại của Docker và dự án mã nguồn mở Docker - mà Docker Inc. và nhiều tổ chức, cá nhân độc lập khác đang cùng đóng góp phát triển.

2.1.2. Một số lợi ích khi sử dụng Docker

Docker hiện nay phổ biến đến nỗi từ "Docker" và "container" được sử dụng thay thế cho nhau, nhưng các công nghệ liên quan đến container đã có sẵn từ rất nhiều năm trước, thậm chí nhiều thập kỷ trước khi Docker được phát hành tới công chúng vào năm 2013. Đáng chú ý nhất là vào năm 2008, LXC (Linux Container) được triển khai trong kernel Linux, hoàn toàn cho phép việc ảo hóa một instance duy nhất của Linux. Các phiên bản ban đầu của Docker đã tận dụng độc quyền LXC, nhưng Docker sau đó đã sớm phát triển công nghệ container hóa tùy chỉnh riêng, cho phép: (1) Cải thiện tính di động: Trong khi các container LXC thường tham chiếu các cấu hình dành riêng cho máy, các container Docker hoạt động mà không cần sửa đổi trên bất kỳ máy tính để bàn, trung tâm dữ liệu và môi trường đám mây nào. (2) Nhe hơn và cập nhật chi tiết hơn: Với LXC, nhiều tiến trình có thể được kết hợp vào trong một container duy nhất. Với container Docker, chỉ một tiến trình có thể chạy trong một container duy nhất. Điều này cho

phép nó xây dựng một ứng dụng có thể tiếp tục chạy trong khi một trong những phần của nó được gỡ xuống để cập nhật hoặc sửa chữa. (3) Tạo container tự động: Docker có thể tự động xây dựng một container dựa trên mã nguồn ứng dụng. (4) Đánh phiên bản container: Docker có thể theo dõi các phiên bản của một container image, quay lại lại các phiên bản trước đó và theo dấu ai đã tạo ra phiên bản, bằng cách nào. Nó thậm chí có thể tải lên các bản thử nghiệm giữa một phiên bản cũ sẵn và một phiên bản mới. (5) Tái sử dụng container: Các container hiện có có thể được sử dụng như các image cơ sở (base image) - về cơ bản giống như các bản mẫu cho việc xây dựng các container mới. (6) Thư viện container dùng chung: Các nhà phát triển có thể truy cập vào một registry mã nguồn mở chứa hàng nghìn các container mà người dùng đóng góp. Vì những lý do này, việc áp dụng Docker nhanh chóng bùng nổ và tiếp tục tăng mạnh.

2.1.3. Một số công cụ và thuật ngữ Docker

(1) DockerFile: Mỗi container Docker bắt đầu với một file văn bản đơn giản chứa các chỉ lệnh về cách xây dựng image của container Docker. DockerFile tự động hóa quá trình tạo image Docker. Nó cơ bản là một danh sách các lệnh mà Docker Engine sẽ thực hiện để tạo ra image. (2) Docker image: Docker image chứa mã nguồn ứng dụng có thể thực thi cũng như tất cả các công cụ, thư viện và gói phụ thuộc mà mã nguồn ứng dụng cần để chạy dưới dạng một container. Khi chạy Docker image, nó sẽ trở thành một (hoặc nhiều) instance của container. Có thể xây dựng một Docker image từ đầu, nhưng hầu hết các nhà phát triển đều kéo chúng xuống từ các kho lưu trữ chung. Nhiều Docker image có thể được tạo từ một image cơ sở và chúng chia sẻ những điểm tương đồng trong stack. Docker image được tạo thành từ các lớp (layer) và mỗi lớp tương ứng với một phiên bản của image. Khi một nhà phát triển thực hiện các thay đổi cho image, một lớp mới phía trên được tạo ra và lớp trên cùng đó thay thế lớp trên cùng trước đó dưới dạng phiên bản hiện thời của image. Mỗi khi một container được tạo từ Docker image, một lớp mới khác sẽ gọi lớp container được tạo. Những sự thay đổi được thực

hiện với container. Ví dụ như thêm hoặc xóa các tệp, chỉ được lưu lại vào lớp container và chỉ tồn tại khi container đang chạy. Quá trình tạo image lập lại như vậy cho phép tăng hiệu quả tổng thể, bởi các instance container trực tiếp có thể chạy từ chỉ một image cơ sở duy nhất và khi chúng làm như vậy, chúng tận dụng được một stack chung.

(3) Docker container: Docker container là các instance đang chạy, trực tiếp của Docker image. Trong khi Docker image là các tệp chỉ-đọc, container là các nội dung trực tiếp có thể thực thi. Người dùng có thể tương tác với chúng và các dự án mã nguồn mở có thể điều chỉnh các thiết lập và điều kiện của chúng. (4) Docker Hub: Docker Hub là kho lưu trữ (repository) công cộng của các Docker image, tự gọi là thư viện và công đồng lớn nhất thế giới về container image. Nó chứa hơn 100 nghìn container image cung cấp từ các nhà cung cấp phần mềm thương mại, các dự án mã nguồn mở và các nhà phát triển độc lập. Nó bao gồm các image được sản xuất bởi Docker Inc., các hình ảnh được chứng nhận thuộc về Docker Trusted Registry, cùng hàng nghìn image khác. Tất cả người dùng Docker Hub có thể chia sẻ image của họ tùy ý. Họ cũng có thể tải xuống các image cơ sở được xác định trước để sử dụng như điểm khởi đầu cho bất kỳ một dự án container hóa nào. (5) Triển khai và điều phối Docker: Nếu bạn chỉ chạy một vài container, việc quản lý các ứng dụng ngay trong chính Docker Engine khá đơn giản. Tuy nhiên, nếu việc triển khai bao gồm hàng nghìn container với hàng trăm dịch vụ, gần như không thể quản lý mà không có sự trợ giúp của các công cụ được xây dựng có mục đích này: Docker Compose, Kubernetes.

2.2. Kubernetes

2.2.1. Nền tảng điều phối container Kubernetes

Kubernetes là một nền tảng điều phối container mã nguồn mở để tự động hóa việc triển khai, quản lý và cơ giới các ứng dụng. Cần hiểu rằng, Kubernetes là một nền tảng điều phối container, nó là sự thay thế cho Docker Swarm hay một nền tảng điều phối container nào khác, chứ nó không phải là sự thay thế hoặc đối thủ của chính Docker. Kubernetes, còn được biết đến như là "k8s" hay "k8", là một nền tảng điều phối container cho

việc lập lịch, tự động hóa việc triển khai, quản lý và cơ giới các ứng dụng container hóa. Kubernetes được phát triển lần đầu tiên bởi các kỹ sư tại Google trước khi được chuyển thành mã nguồn mở vào năm 2014. Nó là hậu duệ của "Borg", một nền tảng điều phối container được sử dụng nội bộ ở Google. Ngày nay, Kubernetes và hệ sinh thái container rộng lớn đang phát triển thành một nền tảng và hệ sinh thái điện toán đa năng, cạnh tranh (nếu không muốn bị vượt qua) với máy ảo, như là các khối xây dựng cơ bản của cơ sở hạ tầng đám mây và ứng dụng hiện đại. Hệ sinh thái này cho phép các tổ chức cung cấp PaaS nhằm giải quyết nhiều tác vụ và vấn đề liên quan đến việc vận hành và liên quan đến cơ sở hạ tầng xung quanh việc phát triển cloud-native, để các nhóm phát triển có thể chỉ tập trung vào việc lập trình và đổi mới.

Một số công việc mà Kubernetes có thể thực hiện: (1) Triển khai: Triển khai một số lượng container được ấn định tới một host được chỉ định và giữ chúng chạy ở một trạng thái mong muốn. (2) Rollout: Rollout là một sự thay đổi trong việc triển khai. Kubernetes cho phép bắt đầu, tạm dừng, tiếp tục hoặc quay lui các rollout. (3) Khám phá dịch vụ: Kubernetes có thể hiển thị tự động một container lên trên Internet hoặc với các container sử dụng tên DNS hoặc địa chỉ IP. (4) Cấp phát lưu trữ: Thiết lập Kubernetes để mount lưu trữ cục bộ hoặc đám mây cho các container khi cần thiết. (5) Cân bằng tải và cơ giới: Khi lưu lượng tới một container đạt đỉnh, Kubernetes có thể sử dụng cân bằng tải và cơ giới để phân phối lưu lượng trên toàn mạng để duy trì sự ổn định. (6) Tự hồi phục với tinh sẵn sàng cao: Khi một container bị lỗi, Kubernetes có thể khởi động lại hoặc thay thế nó tự động, nó cũng có thể tháo gỡ các container không đáp ứng các yêu cầu kiểm tra sức khỏe.

2.2.2. Kiến trúc của Kubernetes

Các thành phần chính của kiến trúc Kubernetes bao gồm: (1) Cluster và node (compute): Cluster (cụm) là các khối xây dựng của kiến trúc Kubernetes. Các cụm được tạo thành từ các node (nút), mỗi nút đại diện cho một host điện toán duy nhất (máy ảo hoặc máy vật lý).

Một cụm bao gồm nhiều worker node (nút làm việc) triển khai, chạy và quản lý các ứng dụng container hóa và một master node (nút chủ) điều khiển và giám sát các worker node. Master node chạy một dịch vụ lập lịch tự động vào thời điểm và tại nơi mà các container được triển khai dựa trên các yêu cầu triển khai được thiết lập bởi nhà phát triển và khả năng điện toán có sẵn. Mỗi worker node bao gồm công cụ được sử dụng để quản lý các container (như Docker) và một agent phần mềm được gọi là một Kubelet nhân và thực hiện các yêu cầu từ master node. (2) Pod và deployment (software): Pod là nhóm các container chia sẻ chung các tài nguyên điện toán và cùng mạng. Chúng cũng là đơn vị của khả năng co giãn trong Kubernetes: Nếu một container trong một pod nhận được lưu lượng nhiều hơn khả năng nó có thể xử lý, Kubernetes sẽ sao chép pod tới một node khác trong cụm. Vì lý do này, việc giữ các pod nhỏ gọn để chúng chỉ chứa các container cần phải chia sẻ tài nguyên là một cách hợp lý. Deployment (sự triển khai) kiểm soát việc tạo và trạng thái của ứng dụng được container hóa, và giữ nó hoạt động. Nó chỉ định bao nhiêu bản sao của một pod nên hoạt động trên cụm. Nếu một pod bị lỗi, Deployment sẽ tạo ra một cái mới.

3. Các khảo sát hiện trạng và xu hướng áp dụng container trong doanh nghiệp

3.1. Đánh giá hiệu quả khi ứng dụng container trong doanh nghiệp

Theo Docker, việc ứng dụng nền tảng container (như Docker container) mang lại giá trị ngay lập tức cho doanh nghiệp bằng cách giảm chi phí cơ sở hạ tầng và bảo trì để hỗ trợ các ứng dụng hiện tại, trong khi đẩy nhanh thời gian giới thiệu các giải pháp mới. Một số kết quả đem lại bao gồm: (1) Thời gian ra mắt thị trường nhanh hơn: Các dịch vụ và ứng dụng mới là yếu tố giúp mang đến sự cạnh tranh. Tổ chức có thể tăng tốc gấp 3 lần (300%) việc phát hành các dịch vụ mới với sự phát triển và vận hành linh hoạt. (2) Năng suất phát triển: Tăng 130 lần, đơn giản hóa việc thiết lập môi trường phát triển, trao quyền cho các nhà phát triển làm việc hiệu quả ngay ngày đầu tiên, loại bỏ sự phụ thuộc để khởi đầu và vận chuyển các mã nguồn mới nhanh và dễ dàng hơn. (3)

Tăng tốc triển khai: Nhanh hơn 60% khi chuyển từ việc phát triển sang triển khai. Container hóa phá vỡ các rào cản để cho các nhóm DevOps có thể tăng tốc số lần và tần suất triển khai. (4) Giảm bớt cơ sở hạ tầng CNTT: Tối ưu chi phí, giảm 40% chi phí bằng việc tăng cường độ tải công việc ứng dụng, tận dụng tốt hơn cường độ tính toán của máy chủ và giảm các chi phí giấy phép bản quyền phần mềm. (5) Hiệu quả hoạt động CNTT: Tăng 40% hiệu quả hoạt động bằng việc hợp lý hóa và tự động hóa sự quản lý các ứng dụng và cơ sở hạ tầng đa dạng vào trong một mô hình vận hành thống nhất để đơn giản hóa chuỗi cung ứng, đồng thời cải thiện khả năng co giãn và bảo mật trên toàn bộ doanh nghiệp. (6) Giải quyết các vấn đề nhanh hơn: Đảm bảo sự hài lòng của khách hàng và các cấp độ dịch vụ bằng cách tăng tốc thời gian trung bình để giải quyết các vấn đề, hơn 72%. Dễ dàng khôi phục và triển khai các bản sửa lỗi cho các ứng dụng (7) Quyền tự do lựa chọn: Đóng gói, vận chuyển và chạy các ứng dụng trên bất kỳ đám mây riêng hoặc công cộng nào.

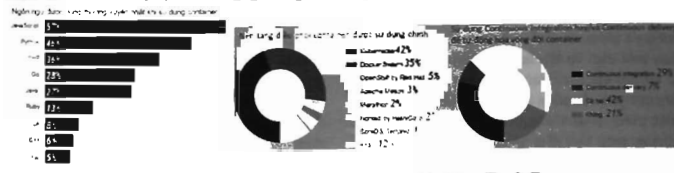
3.2. Một số kết quả khảo sát về việc ứng dụng của container của doanh nghiệp

3.2.1. Báo cáo về xu hướng phát triển đám mây 2018 của DigitalOcean

Theo một báo cáo về các xu hướng phát triển trong đám mây tháng 6/2018 của DigitalOcean (với gần 5.000 người trả lời), các container đang đạt đến điểm bùng phát, với 49% các nhà phát triển hiện đang sử dụng. Với công nghệ và các công cụ quản lý container nâng cấp với tốc độ nhanh chóng trong năm ngoái, container đang trở thành một công cụ đi đầu trong "hành trang" của nhà phát triển. Trong số những người sử dụng, khoảng 40% sử dụng container để phát triển và kiểm thử, trong khi 60% sử dụng cho sản xuất. Đối với những người chưa sử dụng container, 78% nói rằng họ có kế hoạch áp dụng chúng trong tương lai, chứng tỏ rằng các nhà phát triển đang cực kỳ quan tâm đối với container.

Về những khía cạnh lợi ích của container, 39% cho rằng container dễ dàng co giãn, 24% nhận thấy container cho phép họ kiểm tra phần mềm đơn giản hơn và nhanh hơn đối với 23% người trả lời, 10% cho biết container giúp tránh các vấn đề liên quan đến vendor lock-in. (Hình 1)

Hình 1: Kết quả thống kê liên quan đến sử dụng container (theo DigitalOcean 2018)



3.2.2. Khảo sát trực tuyến 2018 của IBM Cloud

Một số đánh giá từ kết quả một cuộc khảo sát trực tuyến vào tháng 8/2017 của IBM (với 206 nhà phát triển, nhà điều hành phát triển và nhà điều hành CNTT):

(1) Theo thứ tự về tầm quan trọng, hào hứng cấp độ doanh nghiệp, các công cụ giúp giảm bớt những thách thức về vận hành, hỗ trợ cho môi trường đám mây và on-premise, các kiến trúc ứng dụng linh hoạt là những yếu tố mấu chốt trong việc áp dụng nền tảng đám mây dựa trên container. (2) Về mặt sử dụng và áp dụng các nền tảng dựa trên container, việc giải quyết các thách thức liên quan đến công nghệ mới hiện đang làm lu mờ nhu cầu về các dịch vụ tiên tiến khác như trí tuệ nhân tạo (AI) hay các công cụ phân tích dữ liệu lớn. (3) Việc sử dụng container có liên quan đến sự cải thiện chất lượng ứng dụng, giảm cả chi phí phát triển ứng dụng và thời gian chết trong ứng dụng sản xuất, tạo điều kiện đổi mới trải nghiệm người dùng. (4) Việc sử dụng container cho các tải công việc của doanh nghiệp sản xuất dự kiến sẽ tăng từ 25% lên 44% trong vòng 3 năm tới. Sự triển khai sẽ chuyển dịch một cách mạnh mẽ sang Đám mây lai (Hybrid cloud) và hỗ trợ cho các môi trường container hóa on-premises, serverless. Việc triển khai chỉ trên đám mây công cộng sẽ giảm. (5) Những người trả lời thấy rằng các ứng dụng doanh nghiệp là phù hợp nhất cho sự phát triển các ứng dụng đang container hóa. Trên 40% số người trả lời cho rằng các ứng dụng liên quan đến phân tích dữ liệu được hưởng lợi nhiều nhất từ việc được container hóa, trong khi khoảng 1/3 cho rằng đó là các dịch vụ web, cơ sở dữ liệu

hay các ứng dụng CRM. (6) Các giải pháp thương mại có thể là chất xúc tác cho việc phát triển dựa trên container để mở rộng vượt ra khỏi các framework mã nguồn mở. Các giải pháp thương mại được nhìn nhận có mối quan hệ khăng khít với DevOps, sự phát triển các vi dịch vụ và các công cụ tự động hóa, làm giảm các thách thức về vận hành và thiếu chắc chắn vốn đang là gánh nặng hiện tại cho thị trường container. (7) Hỗ trợ nền tảng đám mây cho việc phát triển và di trú dễ dàng các ứng dụng chính là nền tảng để thúc đẩy đầu tư vào việc phát triển dựa trên container. 2/3 số người trả lời đánh giá cao khả năng cho phép người dùng thiết kế các môi trường điện toán container. (8) Các dịch vụ tư vấn chỉ được coi là cần thiết cho những nỗ lực phức tạp - ví dụ, khi các công cụ tìm cách tận dụng AI trong các ứng dụng được container hóa và cần hỗ trợ các kiến trúc non-x86. (9) So với các nhà điều hành CNTT, các nhà phát triển ưa thích các công nghệ dựa trên container và chỉ ra lợi ích chiến lược từ việc sử dụng chúng. Các nhà phát triển đặc biệt quan tâm đến các hoàn cảnh sử dụng trong trí tuệ nhân tạo (AI), học máy (Machine learning), dữ liệu lớn (Big data), non-x86 và công nghiệp. (10) Trong phạm vi các vai trò, những người trả lời xem thiếu hụt chuyên môn nội bộ là một trong những thách thức đáng kể trong ứng dụng việc phát triển ứng dụng được container hóa. Các nhà điều hành CNTT quan tâm đến sự thiếu chắc chắn về thời gian và chi phí, trong khi các nhà phát triển gặp vấn đề bởi các thách thức vận hành như tái thiết kế các ứng dụng on-premises và quản lý dữ liệu trên các container. (11) Các nhà phát triển đánh

giá một nền tảng là phù hợp nhất để hỗ trợ các môi trường đa dạng. Đối với các nhà phát triển và nhà điều hành CNTT, khả năng tương thích với môi trường CNTT hiện tại và dễ dàng sử dụng được xếp hạng hàng đầu trong những lý do lựa chọn một nền tảng container. (12) Những người mua chính và có tầm ảnh hưởng nhất trong việc đầu tư cho container là những người đứng đầu về vận hành CNTT và nhà điều hành CNTT cao cấp. Tuy nhiên, những nhà lãnh đạo phát triển, những đội ngũ DevOps và các nhà phát triển ứng dụng cũng có sự ảnh hưởng lớn.

3.2.3. Khảo sát các nhà phát triển năm 2019 của Stack Overflow

Theo “Kết quả khảo sát các nhà phát triển năm 2019” của StackOverflow:

- Về mức độ phổ biến của các nền tảng, với hơn 67.000 người trả lời là các nhà phát triển chuyên nghiệp, Docker đạt tỉ lệ 34,9%, trong khi một số nền tảng khác như AWS (29,5%), Microsoft Azure (13,0%), Google Cloud Platform (12,8%), Heroku (10,6%), Kubernetes (9,6%). Về mức độ yêu thích và muốn tiếp tục sử dụng của các nền tảng, Docker đạt 77,8%, Kubernetes đạt 76,8%, trong khi với các nền tảng đám mây AWS (71,6%), Google Cloud Platform (66,8%), Microsoft Azure (65,4%).

- Về việc sử dụng Container, với hơn 85.000 người trả lời, có gần 54% người trả lời đang sử dụng nền tảng container, trong đó, cho việc phát triển chiếm 38,4%, kiểm thử chiếm 30,2%, sản xuất chiếm 25,9%.

3.2.4. Khảo sát của RightScale 2019 (Flexera)

Tỷ lệ sử dụng Docker và Kubernetes đứng ở hai vị trí đầu và có xu hướng tăng, đặc biệt là tỷ lệ sử dụng Kubernetes tăng rất mạnh (Hình 2).

Hình 2: Tỷ lệ sử dụng các công cụ container năm 2018 và 2019 (theo RightScale 2019)



3.2.5. Khảo sát của Portworx - Aqua Security

Theo khảo sát thường niên “Khảo sát sự ứng dụng container 2019” của Portworx và Aqua Security (với 501 chuyên gia CNTT):

- Có 87% số người trả lời cho biết tổ chức sử dụng các công nghệ container, 10% không sử dụng. Đối với tỷ lệ sử dụng các công nghệ container trong sản xuất, lên tới 89,7% có sử dụng và 9% không sử dụng.

- Tỷ lệ về số ứng dụng đang chạy trên container, công cụ điều phối container phổ biến và kết hợp các nhà cung cấp dịch vụ đám mây để sử dụng container được thể hiện trong các biểu đồ thuộc Hình 3A.

- Thống kê các lý do cơ bản mà tổ chức sử dụng container, xếp hạng các thách thức phải đối mặt khi sử dụng container, được trình bày trong các bảng trên Hình 3B.

3.2.6. Khảo sát của Diamanti

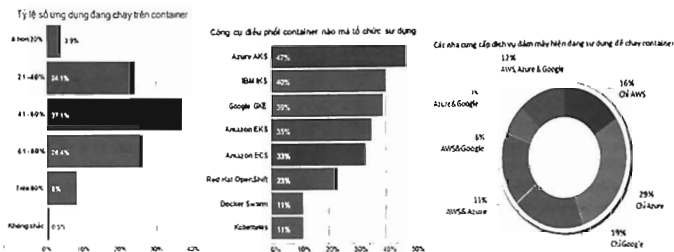
Theo “Khảo sát đánh giá sự ứng dụng container 2019” của Diamanti, so sánh kết quả khảo sát của năm 2018 và năm 2019 (với trên 500 nhà lãnh đạo CNTT):

Khảo sát chỉ ra các ứng dụng cloud-native đứng đầu trong các hoàn cảnh sử dụng container (33%), các đơn vị đang vận hành container trên nhiều môi trường (on-premise đến đám mây). Một số kết quả khảo sát được thể hiện trong các biểu đồ ở Hình 4, gồm các trường hợp triển khai container đến thách thức lớn nhất khi sử dụng container.

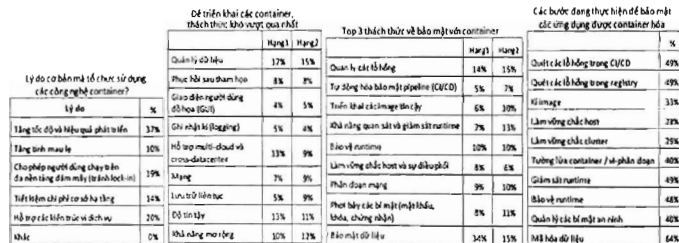
3.2.7. Một số khảo sát và dự đoán của IDC (International Data Corporation) về sự phát triển của container

Theo khảo sát “Container Infrastructure Software Survey 01/2018” của IDC (với 301 người trả lời):

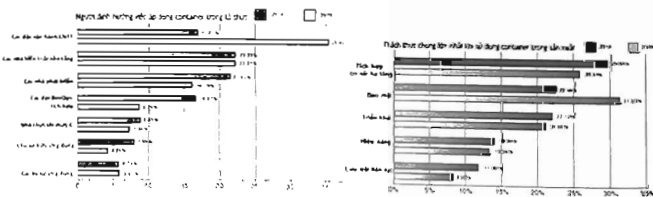
Hình 3A: Kết quả thống kê khảo sát về việc ứng dụng container (theo Portworx 2019)



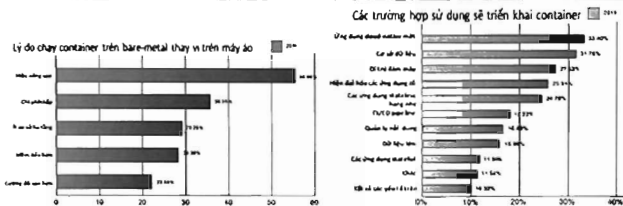
Hình 3B: Kết quả thống kê khảo sát về việc ứng dụng container (theo Portworx 2019)



Hình 4A: Kết quả khảo sát đánh giá việc ứng dụng container (theo Diamanti 2018-2019)



Hình 4B. Kết quả khảo sát đánh giá việc ứng dụng container (theo Diamanti 2018-2019)



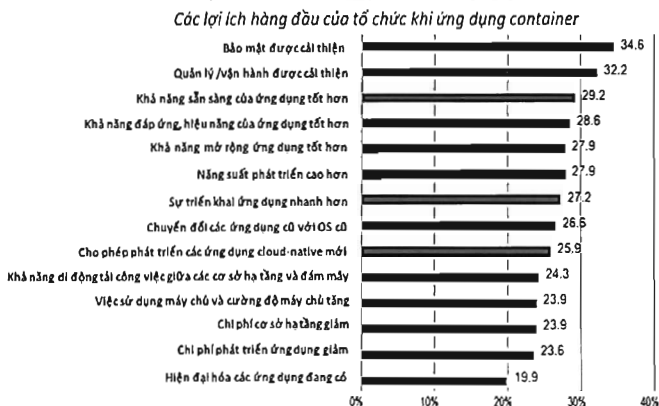
- Các container trong sản xuất và đang được sử dụng cho các ứng dụng quan trọng (mission-critical app). Có 85% số người trả lời đang áp dụng container cho các ứng dụng sản xuất, với 76.4% sử dụng container cho các ứng dụng quan trọng.

- Khoảng 88% sử dụng các container có nhiều hệ điều hành và 51% các nhà triển khai container có cả container Windows và Linux.

83% các nhà triển khai container sử dụng các container trên nhiều đám mây công cộng. Các nhà triển khai container cho biết 55% các container triển khai tại chỗ, trong khi 45% chạy trên đám mây công cộng.

- Khảo sát cũng cho thấy 53.6% ứng dụng container hóa là các ứng dụng có sẵn được di trú tới một container, so với 46% là các ứng dụng mới bắt đầu với container từ ngày đầu. (Hình 5)

Hình 5: Khảo sát top các lợi ích khi ứng dụng container (theo IDC 2018)



Cuộc cách mạng container đang diễn ra ngay lúc này, khi các công ty tìm cách chuyển dịch nhanh hơn và xây dựng các khối tài sản kỹ thuật số để hỗ trợ tăng trưởng. IDC dự báo sẽ có hơn 1,8 tỷ instance container doanh nghiệp được triển khai trong năm 2021. Các nền tảng container sẽ là chìa khóa để quản lý các môi trường CNTT ngày càng đa dạng và mở rộng (đại hệ điều hành, hypervisor, đám mây riêng, đám mây công cộng...). Trong khi các container là phương tiện hoàn hảo để củng cố các ứng dụng cloud-native, kiến trúc vi dịch vụ, chúng cũng phù hợp với nhiều ứng dụng hiện có. Những ứng dụng này có thể gặt hái những lợi ích ngay lập tức từ việc được hiện đại hóa và chuẩn bị cho khả năng tái cấu trúc qua thời gian. Để tận dụng thành công container trong những trường hợp đó và thực hiện ở quy mô lớn, các doanh nghiệp cần một nền tảng container mạnh mẽ, tích hợp và toàn diện.

4. Kết luận

Trong bài báo này, tác giả đã trình bày về Docker và Kubernetes - hai nền tảng container và điều phối container hàng đầu hiện nay, đồng thời cung cấp một số kết quả khảo sát và dự báo về việc ứng dụng nền tảng container trong các tổ chức doanh nghiệp. Bài viết tập trung vào việc tổng hợp và phân tích các kết quả của một số khảo sát về hiện trạng và xu hướng của việc ứng dụng công nghệ container trong doanh nghiệp. Trong bài nghiên cứu ở những số báo sau, tác giả sẽ phân tích về một số nghiên cứu tình huống (case study) điển hình về việc áp dụng container kết hợp các dịch vụ điện toán đám mây của doanh nghiệp, đồng thời đặt ra các thách thức mà doanh nghiệp có thể gặp phải tại quá trình áp dụng nền tảng container cho tổ chức của mình, và những best practice có thể áp dụng trong việc triển khai và vận hành container ■

TÀI LIỆU THAM KHẢO:

1. IBM Cloud Education, "IBM Cloud Learn Hub", IBM, truy cập ngày 01/01/2020, <https://www.ibm.com/cloud/learn/>.
2. Container Journal, "Container Ecosystems Archives", MediaOps Inc., truy cập ngày 01/01/2020, <https://containerjournal.com/topics/container-ecosystems/>.
3. Gartner, "Survey Analysis: Container Adoption and Deployment, 2018", Gartner Inc., truy cập ngày 01/01/2020, <https://www.gartner.com/en/documents/3865408/survey-analysis-container-adoption-and-deployment-2018>.
4. IBM Cloud, "The state of container-based application development", IBM, truy cập ngày 01/01/2020, <https://www.ibm.com/downloads/cas/BBKLLK1L>.
5. Stack Overflow, "Developer Survey Results 2019", Stack Exchange Inc., truy cập ngày 01/01/2020, <https://survey.stackoverflow.com/survey/2019>.
6. RightScale, "RightScale 2019 State of the Cloud Report", Flexera, truy cập ngày 01/01/2020, <https://www.flexera.com/blog/cloud/2019/02/the-cloud-computing-trends-2019-state-of-the-cloud-survey/>.
7. Portworx & Aqua Security, "2019 Container Adoption Survey", Portworx Inc., truy cập ngày 01/01/2020, <https://portworx.com/wp-content/uploads/2019/05/2019-container-adoption-survey.pdf>.
8. Damián, "2019 Container adoption benchmark survey", Damián, truy cập ngày 01/01/2020, <https://www.redhat.com/en/topics/cloud>.
9. Gary Chen, "The Rise of the Enterprise Container Platform", IDC, truy cập ngày 01/01/2020, <https://www.idc.com/getdoc.jsp?docId=44092913L1780image/IDC-containerplatform-wp.pdf>.

10. DigitalOcean, "Currents - A quarterly report on developer trends in the cloud". DigitalOcean, truy cập ngày 01/01/2020, <https://www.digitalocean.com/currents/june-2018/>

11. Phạm Ngọc Duy, Nguyễn Trung Quân (2020). Container hóa với Điện toán đám mây: Giải pháp xu thế trong phát triển và triển khai ứng dụng doanh nghiệp. Tạp chí Công thương, Số 01, tháng 01/2020, tr. 191-200.

Ngày nhận bài: 3/1/2020

Ngày phản biện đánh giá và sửa chữa: 13/1/2020

Ngày chấp nhận đăng bài: 23/1/2020

Thông tin tác giả:

ThS. PHẠM NGỌC DUY

ThS. NGUYỄN THÀNH TUẤN ANH

Khoa Công nghệ thông tin, Trường Đại học Hàng hải Việt Nam

CONTAINERIZATION: A SUMMARY OF SURVEYS ON CONTAINER ADOPTION IN ENTERPRISES

● Master. PHAM NGOC DUY

● Master. NGUYEN THANH TUAN ANH

Faculty of Information Technology,

Vietnam Maritime University

ABSTRACT:

In the Industry 4.0 era, advanced information technology solutions should be applied intensively by organizations and enterprises in all areas in order to achieve effective system building and deployment. Nowadays, the combination of containerization and cloud computing is one of the trendy solutions, especially in enterprise application deployment. The series of articles on "containerization" topic generally presents the overall of container technology and related cloud computing services, recent trends of container-based application solution in organizations, advantages as well as challenges that enterprises have to face in the process of system deployment and operation by using container platform. This article focuses on summarizing and analyzing results of some surveys about current states and future trends in container adoption in enterprises.

Keywords: Container, containerization, container adoption, survey.