内容

[**A.Cloud computing , AWS là gì ?** 2](#_Toc453058149)

[**1.Đám mây là gì ?** 2](#_Toc453058150)

[**2.PaaS,IaaS và SaaS là gì** 8](#_Toc453058151)

[**3.AWS là gì ?** 10](#_Toc453058152)

[**4. Bổ sung** 12](#_Toc453058153)

[**B.Các service của AWS** 18](#_Toc453058154)

[**1.Sơ lược về các dịch vụ cơ bản thường dùng của AWS** 18](#_Toc453058155)

[**2.Một số dịch vụ cần nắm vững** 31](#_Toc453058156)

[**2.1.EC2** 31](#_Toc453058157)

[**2.2.VPC** 34](#_Toc453058158)

[**2.3.Route 53** 39](#_Toc453058159)

[**2.4.S3** 39](#_Toc453058160)

[**2.5.ElasticCache** 44](#_Toc453058161)

[**2.6.Cloud Front** 45](#_Toc453058162)

[**2.7.RDS** 46](#_Toc453058163)

[**2.8.CloudWatch** 46](#_Toc453058164)

[**2.9.SNS** 47](#_Toc453058165)

[**2.10.IAM** 48](#_Toc453058166)

[**C.Ứng dụng thực tế** 48](#_Toc453058167)

[**1. Tạo ứng dụng Rails App trên môi trường Amazon Linux** 48](#_Toc453058168)

[**2.Hướng dẫn cài đặt Redmine trên EC2** 54](#_Toc453058169)

**STEP 5: Amazon Web Service cơ bản**

**(14/9/2015-25/9/2015)**

Các phần cần nghiên cứu:

+ Thao tác Console Amazon Web Service

+ Service cơ bản

+ Luyện tập Network

Tìm hiểu AWS là gì ? Sử dụng AWS để làm gì ? EC2 là gì ? Cấu trúc network trong AWS và thiết lập LB (Load Barancer ) trong AWS.

Trước khi đi vào nghiên cứu phần này, ta sẽ sơ lược qua về 1 số kiến thức cơ bản

# **A.Cloud computing , AWS là gì ?**

## **1.Đám mây là gì ?**

Trước hết để biết AWS là gì, thì ta sẽ nói qua về định nghĩa the cloud computing (điện toán đám mây) trước.

Định nghĩa điện toán đám mây là 1 định nghĩa khá mơ hồ, vì vậy mà ta có thể tham khảo 1 số bài viết cũng như cách tiếp cận để ta hiểu rõ hơn về điện toán đám mây.

* 1. **Cách tiếp cận 1.**

Đây là lời giải thích của tiến sĩ Jony Vũ về điện toán đám mây.

Tham khảo: <http://science-technology.vn/?p=3408>

Question: Một người chủ doanh nghiệp viết cho tôi: “Bạn tôi gợi ý rằng tôi dùng dịch vụ tính toán mây (điện toán đám mây) cho công ty của tôi nhưng tôi không biết nó là gì. Tôi là dân kinh doanh, không phải là dân máy tính. Tôi cần câu trả lời đơn giản nào đó về tại sao tôi cần tính toán mây? Xin thầy lời khuyên.”

Answer (có rút gọn so với bản gốc): Bạn là người chủ của công ty, mục đích của bạn là tăng thu nhập, giảm chi phí, và làm cực đại lợi nhuận. Để đạt tới điều đó, bạn ứng dụng CNTT để tự động hoá các qui trình của bạn để làm tăng tính hiệu quả và hiệu lực của công nhân của bạn. Tuy nhiên, bằng việc có hệ thống CNTT tại công ty của bạn, bạn phải mua phần mềm, phần cứng và thuê nhân viên IT để duy trì nó cho nên có chi phí phụ. Cứ sau vài năm bạn lại phải nâng cấp phần mềm và phần cứng. Nhân viên IT của bạn cũng cần đào tạo thêm để duy trì hệ thống, cho nên bạn phải chi tiền cho đào tạo họ nữa. Thỉnh thoảng họ lại nghỉ việc thì bạn phải thuê người mới và trả tiền đào tạo cho họ lần nữa. Như vậy, hệ thống thông tin đã trở thành một chi phí lớn, làm bạn phải chú tâm vào nó, sao lãng khỏi kinh doanh chính của bạn. Là người chủ công ty, bạn tự hỏi tại sao bạn phải chi nhiều cho cái gì đó KHÔNG liên quan tới kinh doanh của bạn? Tại sao bạn phải lo nghĩ về cái gì đó làm sao lãng bạn khỏi việc vận hành kinh doanh của bạn? Có giải pháp mang tên Tính toán mây (điện toán đám mây) và ngày nay nó đang làm thay đổi toàn thể ngành công nghiệp máy tính: Thay vì có hệ thống tại công ty với phần cứng và phần mềm, bạn có thể để nhân viên của bạn truy nhập vào ứng dụng dựa trên Web mà duy trì mọi chương trình họ cần để làm công việc của họ, từ e-mail tới xử lí văn bản, bảng tính số liệu, phần mềm bán hàng, kế toán... Bạn KHÔNG phải lo nghĩ về nâng cấp phần cứng hay phần mềm; bạn KHÔNG phải lo nghĩ về thuê nhân viên IT hay đào tạo họ. Bạn KHÔNG phải lo nghĩ về gia hạn giấy phép bản quyền phần mềm. Bạn KHÔNG phải lo về mua, bảo trì hệ thống thông tin. Về căn bản thay vì “mua” bạn “thuê” mọi nhu cầu CNTT từ công ty dịch vụ tính toán mây.

* 1. **Cách tiếp cận 2**

Đây là nội dung của 1 video giải thích về điện toán đám mây

Tham khảo:

+ <https://www.youtube.com/watch?v=0aBVR3MHsHE>

Sub cho những ai ko tiện xem video: Hiện nay, những gì về điện toán đám mây mà bạn được nghe, những tác dụng kì diệu của nó chỉ là những gì quảng cáo thổi phồng lên thôi. Nhưng thực sự thì nó có ý nghĩa gì ? Liệu có phải bạn ngồi trên 1 đỉnh núi và xung quanh là những đám mây ? Hay phải ngồi trên những chiếc máy bay ? Và liệu nó (đám mây) có hoạt động khi trời nắng, hoặc trời mưa.

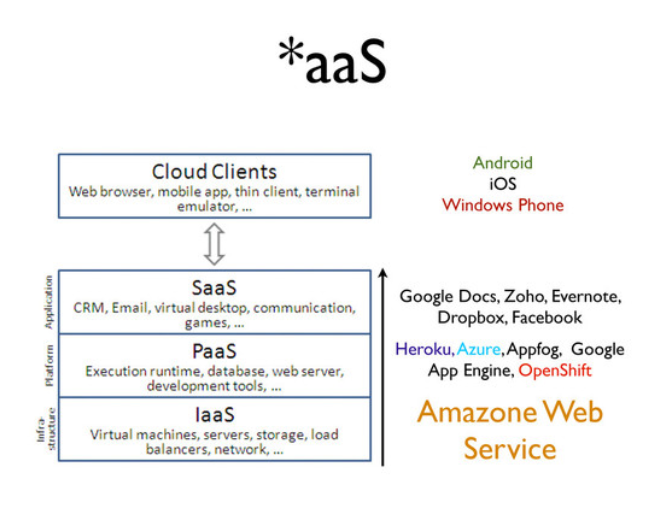
Điện toán đám mây là 1 khái niệm khá rộng lớn và nó ngày càng to lớn.

Về bản chất, điện toán đám mây có 3 tầng

//++++++++++| Tầng ứng dụng(Application)

//++++++| Tầng nền tảng (Platform)

//+++| Tầng cơ sở hạ tầng (Infrastructure)



Các cty có cách sử dụng chúng khác nhau, tùy theo sản phẩm mà họ cung cấp

Nền tảng của nó trước hết vẫn là các trang thiết bị - Infrastructure.

Đây là nơi mọi thứ bắt đầu và cũng là thứ con người cần phải có trước tiên.Đây cũng chính là nhược điểm của các máy chủ đám mây.

Trước tiên phải nói về việc lưu trữ trên mây(Thuê máy chủ trên mây gọi là máy chủ đám mây) - một khái niệm đã cũ , nhưng nhờ có những tiến bộ công nghệ thì nó lại hoạt động trơn tru hơn khá nhiều.

Giả sử bạn có 1 công ty, và bạn có 1 website, và trang web có cung cấp chức năng tạo nhiều cuộc nói chuyện ngắn, lúc đầu chỉ là 1 số thành viên nói chuyện với nhau,nhưng sau đó số lượng thành viên nhiều lên, bạn thành công. Tuy nhiên thành công sẽ đi liền với rắc rối .Khả năng của bạn không đáp ứng kịp với nhu cầu , dịch vụ của bạn bị chậm lại và ngay lập tức, bạn gặp rắc rối .

Có thể vài năm trước, bạn đặt trang web của mình trên 1 máy tính hoặc 1 máy chủ ở đâu đó , và khi thành công kịch ngưỡng, bạn phải chạy khắp nơi lo thuê hoặc mua máy chủ ,tự thiết lập chúng hoặc thuê ai đó thiết lập giúp bạn.Hàng trăm nghìn công ty khác cũng đang làm như bạn vậy. Và bạn sẽ tốn nhiều tiền và thời gian, việc này gọi là thuê máy chủ (hosting)

Bạn mất nhiều tiền bạc và thời gian cho việc thuê máy chủ cũng như duy trì để chúng hoạt động liên tục.Bạn phải trả tiền khi sử dụng chúng, và bạn vẫn phải trả tiền khi không sử dụng chúng .

Dĩ nhiên bạn là người xây dựng dịch vụ, và bạn không ngừng cải tiến sản phẩm của mình. Nhưng thay vì thành công, hệ thống của bạn gặp rắc rối trong việc bảo trì hệ thống, cũng như phần cứng sẽ khiến khách hàng nổi giận và bạn có nguy cơ phá sản.

Và bây giờ bạn có lựa chọn tốt hơn nhiều: Điện toán đám mây.

Với điện toán đám mây, bạn có thể truy xuất với sức mạnh điện toán tức thì ,đưa trang web của bạn lên máy chủ đám mây cũng chẳng khác gì với việc bạn sử dụng 1 máy chủ chuyên dụng khác .Và rồi mọi người lại truy cập vào website của bạn.Nếu bạn cần thêm sức mạnh điện toán chuyên dụng cho trang web của mình, bạn có thể ngay lập tức mở rộng nó ra (scale) bao nhiêu tùy thích , và bạn nhận được sức mạnh điện toán đúng như bạn cần, còn khách hàng sẽ hài lòng với dịch vụ mà bạn cung cấp mà ko nhận ra sự khác biệt nào cả.

Nếu lưu lượng truyền tải giảm xuống, bạn có thể dễ dàng giải phóng máy chủ trở lại các đám mây .

Đây là 1 ưu điểm lớn nhất nên tôi nhắc lại 1 lần nữa : khi bạn cần sức mạnh điện toán, bạn có thể lấy nó ngay lập tức từ các đám mây, và khi dùng xong, bạn có thể trả lại . Điều đó cũng giống như bạn trả tiền ga hay tiền điện vậy, bạn bật công tắc, công tơ quay, và tính mức phí bạn phải trả cho số điện bạn dùng. Khi bạn ko dùng điện,bạn tắt nó đi, công tơ không quay nữa, và bạn sẽ chỉ phải trả số điện bạn đã dùng. Hay cũng như việc bạn đi tắc xi, bạn đi bao xa, công tơ sẽ tính khoảng cách bạn đi và tính ra số tiền bạn phải trả. Vậy tại sao bạn phải trả tiền mua 1 cái xe, trong khi bạn chỉ cần trả tiền cho chuyến đi đó thôi !

Nhưng thực sự làm việc với các đám mây ra sao ? Bạn không thể chạm tay vào phần cứng được !

Đơn giản thôi ! Chỉ cần dùng chiếc máy tính cá nhân của bạn, bạn truy cập vào máy chủ đám mây và điều khiển nó từ xa .Bạn có thể đưa bất cứ thứ gì lên đó, như trang web hay mã phần mềm... và sau khi dùng xong thì ngắt kết nối .Và bạn chẳng bao giờ sờ được vào nó. Thuật ngữ "Điện toán đám mây", không chỉ ám chỉ phần cứng, đó có thể là website, có thể là mã phần mềm bạn đang thực hiện. Mọi người, giống như bạn, đều có thể sở hữu những dịch vụ điện toán đám mây. Bạn có bao giờ để ý rằng những email của bạn được đặt ở đâu hay trên máy chủ vật lý nào không ?

Chắc là không, điều quan trọng hơn là các dịch vụ này hoạt động, và sự ổn định, đáng tin cậy cũng như dễ sử dụng của nó.

Dưới đây là 3 lý do mà điện toán đám mây lại trở nên phổ biến như vậy :

a. Tính co giãn (Scalability)

Dễ dàng mở rộng hoặc thu hẹp để đáp ứng yêu cầu của bạn

Nếu bạn cần 1,2 hay 10 máy chủ, thì với điện toán đám mây, bạn có thể giới hạn số lượng máy chủ theo yêu cầu

b. Tính tức thì (instant)

Bạn có thể thuê máy chủ đám mây tức thì , sức mạnh điện toán luôn có sẵn bất kì khi nào bạn cần .

Chỉ cần bạn bật lên thôi, bất cứ thứ gì bạn cần đều có sẵn trên các đám mây ngay lập tức .

Nếu bạn muốn cắt giảm chi phí, đơn giản là bạn tắt nó đi.

c. Tính tiết kiệm (Save money)

Bạn sẽ chỉ phải trả chi phí cho những gì bạn sử dụng và không cần có trang thiết bị xung quanh.

Như vậy với việc thuê máy chủ đám mây, bạn sẽ không cần lo lắng về phần cứng nữa, thay vào đó bạn chỉ cần tập trung

cho công việc phát triển kinh doanh là đủ. Sử dụng máy chủ đám mây sẽ giúp bạn tiết kiệm nhiều tgian và tiền bạc.

Vậy còn chờ gì nữa mà không vứt bỏ các máy chủ cũ đi và chuyển qua các dịch vụ đám mây :D

Như vậy ta có thể tóm gọn lại về điện toán đám mây chưa ? “**Điện toán đám mây có thể được định nghĩa một cách đơn giản như là sự sử dụng tài nguyên tính toán có khả năng thay đổi theo nhu cầu, được cung cấp như là một dịch vụ từ bên ngoài với chi phí trả cho mỗi lần sử dụng. Bạn có thể truy cập đến bất kỳ tài nguyên nào tồn tại trong "đám mây (cloud)" tại bất kỳ thời điểm nào và từ bất kỳ đâu thông qua hệ thống Internet. Bạn không phải quan tâm xem làm cách nào các thứ đang được duy trì phía trong của đám mây**”

Một số bài viết nên tham khảo:

+ <http://quantrimang.com/tim-hieu-ve-dich-vu-amazon-ec2-83626>

+ <http://tuhocanninhmang.com/dien-toan-dam-may/co-ban-ve-dien-toan-dam-may.htm> (Bài này giải thích rất dễ hiểu)

+ <http://tuhocanninhmang.com/dien-toan-dam-may/tim-hieu-ve-mo-hinh-dien-toan-dam-may.htm>

+ <http://www.ibm.com/developerworks/vn/library/cl-cloudintro/>

+ <http://kimhieu.me/technology/cloud-computing-nhung-dieu-ban-nen-biet.html>

Đây là một số bài viết hay và dễ hiểu về cloud computing

+ <http://anvona.com/tong-quan-dien-toan-dam-may-phan-1/>

+ <http://anvona.com/tong-quan-dien-toan-dam-may-phan-2/>

+ <http://ledduy.blogspot.jp/2012/03/cloud-computing-with-amazon-ec2-free.html>

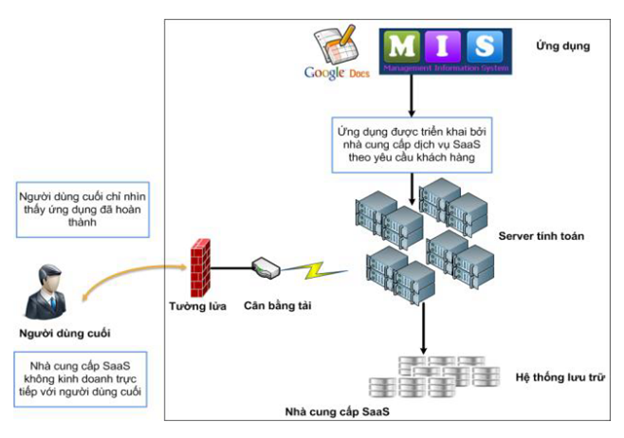
## **2.PaaS,IaaS và SaaS là gì**

Các dịch vụ có thể chia thành 3 lớp chính: Phần mềm dịch vụ (software as a service), nền dịch vụ (platform as a service), và cơ sở hạ tầng dịch vụ (infrastructure as a service). Các lớp này có thể tập hợp thành các tầng kiến trúc khác nhau, có thể chồng chéo, gối nhau.

[](https://dinhnguyenngoc.files.wordpress.com/2014/07/image24.png)

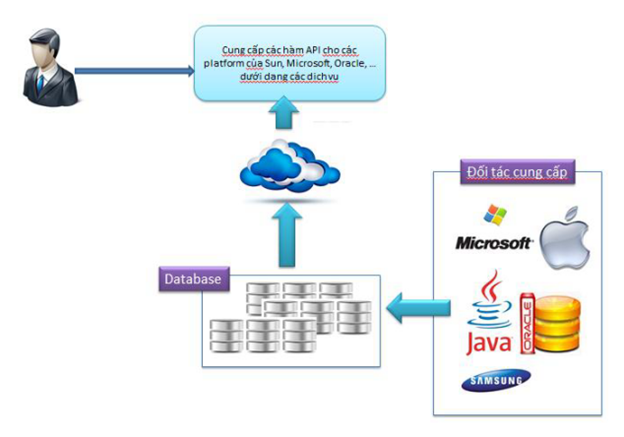
**2.1. Phần mềm hoạt động như dịch vụ (SaaS – Software as a Service)**

*Phần mềm hoạt động hướng dịch vụ hoặc gọi tắt là Phần mềm dịch vụ*, là mô hình triển khai phần mềm, một nhánh của điện toán đám mây, theo đó các nhà cung cấp phần mềm như là các dịch vụ theo yêu cầu cho khách hàng. Theo định nghĩa của hãng nghiên cứu toàn cầu IDC là: "phần mềm hoạt động trên web, được quản lý bởi nhà cung cấp và cho phép người sử dụng truy cập từ xa".

[](https://dinhnguyenngoc.files.wordpress.com/2014/07/image25.png)

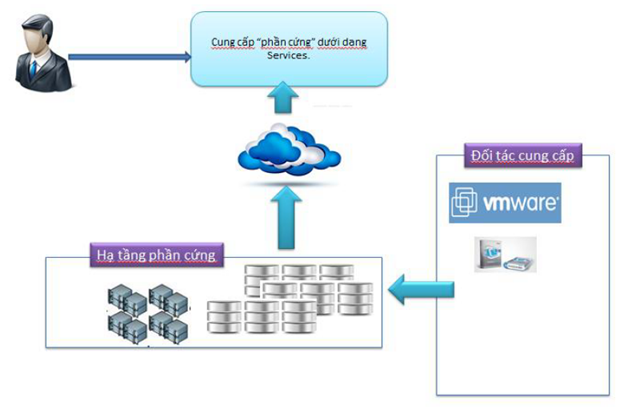
**2.2. Nền tảng hướng một dịch vụ (PaaS – Platform as a Service)**

Đây cũng là một biến thể của SaaS nhưng mô hình này là một nhánh của điên toán đám mây (cloud computing), mang đến môi trường phát triển như một dịch vụ: người sử dụng xây dựng ứng dụng chạy trên cơ sở hạ tầng của nhà cung cấp và phân phối tới người sử dụng thông qua máy chủ của nhà cung cấp đó . Người sử dụng sẽ không hoàn toàn được tự do vì bị ràng buộc về mă ̣t thiết kế và công nghệ . Một số ví dụ điển hình về PaaS là Force.com của Salesforce.com, Google App Engine, Yahoo Pipes …

[](https://dinhnguyenngoc.files.wordpress.com/2014/07/image26.png)

**2.3. Hạ tầng hướng dịch vụ (Iaas – Infrastructure as a Service)**

Infrastructure as a service (IaaS) Là tầng thấp nhất của ĐTĐM, nơi tập hợp các tài sản vật lý như các phần cứng máy chủ, hệ thống lưu trữ và các thiết bị mạng, được chia sẽ và cung cấp dưới dạng dịch vụ IaaS cho các tổ chức hay doanh nghiệp khác nhau. Cũng giống như dịch vụ PaaS, ảo hóa là công nghệ được sử dụng rộng rãi để tạo ra cơ chế chia sẽ và phân phối các nguồn tài nguyên theo yêu cầu. Ví dụ về các dịch vụ IaaS như IBM BlueHouse, Vmware, Amazon EC2, Microsoft Azure Platform, Sun Parascale Cloud Storage…

[](https://dinhnguyenngoc.files.wordpress.com/2014/07/image27.png)

Ngoài ra thì nên tham khảo bài viết sau để hiểu rõ hơn về SaaS

+ SaaS: <http://huykoncode.blogspot.jp/2012/05/phan-2-lam-ro-saas-paas-iaas.html>

+ <https://dinhnn.com/tag/iaas/>

## **3.AWS là gì ?**

Vậy AWS và the cloud computing thì liên quan gì đến nhau ? **AWS chính là dịch vụ điện toán đám mây**. Amazon Web Services là tập hợp các dịch vụ cung cấp cho người lập trình có khả năng truy cập tới hạ tầng kiến trúc tính toán kiểu sẵn sàng-để-sử dụng (ready-to-use) của Amazon. Các máy tính có nền tảng vững chắc đã được xây dựng và tinh chế qua nhiều năm của Amazon bây giờ là có thể cho phép bất cứ ai cũng có quyền cập tới Internet.

Tham khảo:

+ <http://codingsoup.com/2014/02/gioi-thieu-aws-amazon-web-services/>

Tham khảo: Đây là chuỗi các bài viết giới thiệu sơ lược ở mức cơ bản về AWS, bao gồm 4 phần

+ Khái quát về AWS: <http://qiita.com/hiroshik1985/items/6433d5de97ac55fedfde>

+ VPC là gì ?

<http://qiita.com/hiroshik1985/items/9de2dd02c9c2f6911f3b>

+ EC2 là gì ?

<http://qiita.com/hiroshik1985/items/f078a6a017d092a541cf>

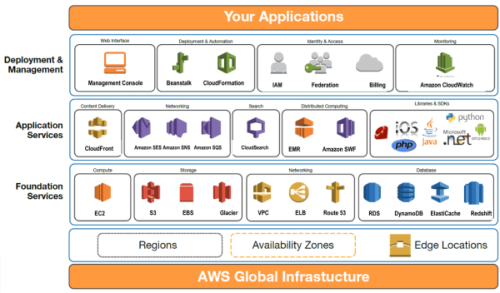
+ ELB là gì ?

<http://qiita.com/hiroshik1985/items/ffda3f2bdb71599783a3>

+ RDS là gì ?

<http://qiita.com/hiroshik1985/items/6643b7323183f82297b2>

Tổng quan kiến trúc của AWS



**Global Infrastructure** (Tạm dịch là cơ sở hạ tầng nằm trên toàn cầu) AWS cung cấp một cơ sở hạ tầng Cloud bảo mật cao cho cả những doanh nghiệp nhỏ cũng như các công ty toàn cầu. Cơ sở hạ tầng bao gồm Regions, Availability Zones, và Edge Locations cho phép các doanh nghiệp lưu trữ dữ liệu của họ một cách mềm dẻo trên nhiều khu vực địa lý và nhiều cơ sở được đặt trên một khu vực địa lý cụ thể.

Tiếp theo ta sẽ đi tìm hiểu các dịch vụ cơ sở và dịch vụ ứng dụng của AWS

## **4. Bổ sung**

Một bài viết hay về Cloud Computing (nên đọc)

Tìm hiểu về Cloud - Điện toán đám mây

**4.1. Điện toán đám mây là gì?**

Điện toán đám mây (tiếng Anh: cloud computing), còn gọi là điện toán máy chủ ảo, là mô hình điện toán sử dụng các công nghệ máy tính và phát triển dựa vào mạng Internet. Thuật ngữ “đám mây” ở đây là lối nói ẩn dụ chỉ mạng Internet (dựa vào cách được bố trí của nó trong sơ đồ mạng máy tính) và như một liên tưởng về độ phức tạp của các cơ sở hạ tầng chứa trong nó.

**4.2. Sự khác biệt giữa điện toán đám mây và điện toán truyền thống**

Trong mô hình điện toán truyền thống, các cá nhân, doanh nghiệp sẽ xây dựng riêng cơ sở hạ tầng kỹ thuật để tự cung cấp các dịch vụ cho các hoạt động thông tin đặc thù của mình. Với mô hình này, mọi thông tin sẽ được lưu trữ, xử lý nội bộ và họ sẽ trả tiền để triển khai, duy trì cơ sở hạ tầng đó (mua thiết bị phần cứng, phần mềm chuyên dụng, trả lương cho bộ phận điều hành...).

Khác với mô hình điện toán truyền thống, điện toán đám mây lưu trữ và xử lý toàn bộ thông tin trong đám mây Internet. Mọi công nghệ, kỹ thuật, cơ sở hạ tầng cũng như chi phí triển khai trong đám mây sẽ do nhà cung cấp đảm bảo xây dựng và duy trì. Do đó, thay vì phải đầu tư từ đầu rất nhiều tiền cho chi phí xây dựng cơ sở hạ tầng riêng, các cá nhân, doanh nghiệp trong quá trình hoạt động sẽ chỉ phải trả số tiền vừa đủ theo nhu cầu sử dụng của mình (pay-for-what-you-use). Như vậy, mô hình này có rất nhiều lợi ích như sử dụng hợp lý nguồn vốn, điều hòa chi tiêu tính toán theo thực tế sử dụng, luôn hưởng năng suất tính toán theo cam kết của nhà cung cấp dịch vụ, tận dụng được sức mạnh của Internet và các siêu máy tính, giảm cơ bản trách nhiệm quản lý hệ thống CNTT nội bộ.

**4.3. Các mô hình triển khai Cloud Computing**

Cho dù sử dụng loại mô hình dịch vụ nào đi nữa thì cũng có ba mô hình triển khai chính là: Public Cloud, Private Cloud và Hybrid Cloud.

- Public Cloud

Các dịch vụ Cloud được nhà cung cấp dịch vụ cung cấp cho mọi người sử dụng rộng rãi. Các dịch vụ được cung cấp và quản lý bởi một nhà cung cấp dịch vụ và các ứng dụng của người dùng đều nằm trên hệ thống Cloud.

Người sử dụng dịch vụ sẽ được lợi là chi phí đầu tư thấp, giảm thiểu rủi ro do nhà cung cấp dịch vụ đã gánh vác nhiệm vụ quản lý hệ thống, cơ sở hạ tầng, bảo mật… Một lợi ích khác của mô hình này là cung cấp khả năng co giãn (mở rộng hoặc thu nhỏ) theo yêu cầu của người sử dụng.

Tuy nhiên Public Cloud có một trở ngại, đó là vấn đề mất kiểm soát về dữ liệu và vấn đề an toàn dữ liệu. Trong mô hình này mọi dữ liệu đều nằm trên dịch vụ Cloud, do nhà cung cấp dịch vụ Cloud đó bảo vệ và quản lý. Chính điều này khiến cho khách hàng, nhất là các công ty lớn cảm thấy không an toàn đối với những dữ liệu quan trọng của mình khi sử dụng dịch vụ Cloud.

- Private Cloud

Trong mô hình Private Cloud, cơ sở hạ tầng và các dịch vụ được xây dựng để phục vụ cho một tổ chức (doanh nghiệp) duy nhất. Điều này giúp cho doanh nghiệp có thể kiểm soát tối đa đối với dữ liệu, bảo mật và chất lượng dịch vụ. Doanh nghiệp sở hữu cơ sở hạ tầng và quản lý các ứng dụng được triển khai trên đó. Private Cloud có thể được xây dựng và quản lý bởi chính đội ngũ IT của doanh nghiệp hoặc có thể thuê một nhà cung cấp dịch vụ đảm nhiệm công việc này.

Như vậy, mặc dù tốn chi phí đầu tư nhưng Private Cloud lại cung cấp cho doanh nghiệp khả năng kiểm soát và quản lý chặt chẽ những dữ liệu quan trọng.

- Hybrid Cloud

Như chúng ta đã phân tích ở trên, Public Cloud dễ áp dụng, chi phí thấp nhưng không an toàn. Ngược lại, Private Cloud an toàn hơn nhưng tốn chi phí và khó áp dụng. Do đó nếu kết hợp được hai mô hình này lại với nhau thì sẽ khai thác ưu điểm của từng mô hình. Đó là ý tưởng hình thành mô hình Hybrid Cloud.

Hybrid Cloud là sự kết hợp của Public Cloud và Private Cloud. Trong đó doanh nghiệp sẽ “out-source” các chức năng nghiệp vụ và dữ liệu không quan trọng, sử dụng các dịch vụ Public Cloud để giải quyết và xử lý các dữ liệu này. Đồng thời, doanh nghiệp sẽ giữ lại các chức năng nghiệp vụ và dữ liệu tối quan trọng trong tầm kiểm soát (Private Cloud).

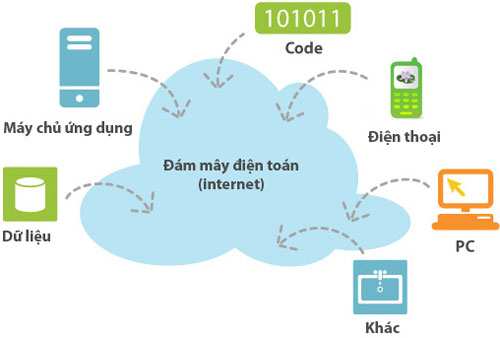
Một khó khăn khi áp dụng mô hình Hybrid Cloud là làm sao triển khai cùng một ứng dụng trên cả hai phía Public và Private Cloud sao cho ứng dụng đó có thể kết nối, trao đổi dữ liệu để hoạt động một cách hiệu quả.

Doanh nghiệp có thể chọn để triển khai các ứng dụng trên Public, Private hay Hybrid Cloud tùy theo nhu cầu cụ thể. Mỗi mô hình đều có điểm mạnh và yếu của nó. Các doanh nghiệp phải cân nhắc đối với các mô hình Cloud Computing mà họ chọn. Và họ có thể sử dụng nhiều mô hình để giải quyết các vấn đề khác nhau. Nhu cầu về một ứng dụng có tính tạm thời có thể triển khai trên Public Cloud bởi vì nó giúp tránh việc phải mua thêm thiết bị để giải quyết một nhu cầu tạm thời. Tương tự, nhu cầu về một ứng dụng thường trú hoặc một ứng dụng có những yêu cầu cụ thể về chất lượng dịch vụ hay vị trí của dữ liệu thì nên triển khai trên Private hoặc Hybrid Cloud.

**4.4. Điện toán đám mây và 10 câu hỏi bảo mật thường gặp**

Mô hình điện toán đám mây dường như ngày càng được ưa chuộng. Tuy nhiên, nghiên cứu gần đây cho thấy vấn đề về bảo mật là rào cản lớn nhất quyết định liệu điện toán đám mây có được sử dụng rộng rãi nữa hay không. Điện toán đám mây hay điện toán máy chủ ảo là mô hình điện toán sử dụng các công nghệ máy tính và phát triển dựa vào mạng Internet. Thuật ngữ "đám mây" ở đây chính là mạng Internet và các kết cấu hạ tầng bên trong.

Trên thực tế, điện toán đám mây đơn giản chỉ là một bước tiến khác trong cách mạng công nghệ thông tin. Mô hình đám mây được phát triển dựa trên 3 yếu tố cơ bản gồm máy tinh trung ương, máy chủ/khách và ứng dụng Web. Nhưng bản chất của 3 thành phần này đều tồn tại các vấn đề về bảo mật.



Mô hình đơn giản của điện toán đám mây

Các vấn đề bảo mật vẫn không ngăn được sự bùng nổ công nghệ cũng như sự ưa chuông điện toán đám mây bởi khả năng giải quyết và đáp ứng các nhu cầu bức thiết trong kinh doanh. Để đảm bảo an toàn cho đám mây điện toán, chúng ta cần nắm được vai trò của nó trong sự phát triển công nghệ. Rất nhiều câu hỏi tồn tại xung quanh những ưu và khuyết điểm khi sử dụng điện toán đám mây trong đó tính bảo mật, hữu dụng và quản lí luôn được chú ý xem xét kĩ lưỡng. Bảo mật là đề tài được giới người dùng thắc mắc nhiều nhất và sau đây là 10 câu hỏi hàng đầu được đặt ra để quyết định liệu việc triển khai điện toán đám mây có phù hợp hay không và nếu không thì nên chọn mô hình nào cho phù hợp: cá nhân, công cộng hay cả 2.

4.4.1. Khả năng rủi ro khi triển khai mô hình điện toán đám mây?

Dù mang tính chất cá nhân hay công cộng thì chúng ta vẫn không thể hoàn toàn quản lí được môi trường, dữ liệu và kể cả con người. Những thay đổi trong mô hình có thể làm tăng hoặc giảm rủi ro. Những ứng dụng đám mây cung cấp thông tin rõ ràng, các công cụ thông báo tiên tiến và tích hợp với hệ thống sẵn có sẽ làm giảm rủi ro. Tuy nhiên, một vài ứng dụng khác lại không thể điều chỉnh các trạng thái bảo mật, không phù hợp với hệ thống sẽ làm gia tăng rủi ro.

4.4.2. Cần phải làm gì để chắc chắn chính sách bảo mật hiện tại tương thích với mô hình đám mây?

Mỗi thay đổi trong mô hình là mỗi dịp để ta cải thiện tình trạng và chính sách bảo mật. Vì người sử dụng sẽ tác động và điểu khiến mô hình đám mây nên chúng ta không nên tạo ra chính sách bảo mật mới. Thay vào đó là mở rộng chính sách hiện thời để tương thích với các nền tảng kèm theo. Để tay đổi chính sách bảo mật thì ta cần xem xét các yếu tố tương quan như: dữ liệu sẽ được lưu ở đâu, bảo vệ như thế nào, ai được phép truy cập, và cần tuân theo những quy tắc và thỏa hiệp gì.

4.4.3. Việc triển khai mô hình đám mây có đáp ứng được yêu cầu ủy thác?

Triển khai mô hình đám mây tác động đến tỉnh rủi ro và ảnh hưởng đến khả năng đáp ứng các quy tắc khác nhau. Một vài ứng dụng đám mây có khả năng thông báo hay báo cáo tình trạng hoạt động mạnh mẽ đồng thời được thiết lập để đáp ứng những yêu cầu thích ứng riêng biệt. Trong khi một số lại quá chung chung và không thể đáp ứng được những yêu cầu chi tiết. Ví dụ như khi chúng ta truy xuất dữ liệu, một thông báo hiện ra cho biết dữ liệu chỉ được lưu trữ trong phạm vi lãnh thổ (server trong nước) thì chúng ta không thể truy xuất được bởi các nhà cung cấp dịch vụ không thể thực hiện yêu cầu này.

4.4.4. Liệu các nhà cung cấp dịch vụ sử dụng các chuẩn bảo mật hay theo thực tế kinh nghiệm (SAML, WSTrust, ISO, v.v..)?

Các tiêu chuẩn có vai trò quan trọng trong điện toán đám mây như một sự tương kết giữa các dịch vụ và ngăn tình trạng độc quyền dịch vụ bảo mật. Rất nhiều tổ chức được thành lập nhằm khởi tạo và mở rộng để hổ trợ trong bước khởi đầu triển khai mô hình. Danh sách các tổ chức hổ trợ được liệt kê tại: Cloud-standards.org.

4.4.5. Điều gì sẽ xảy ra nếu vi phạm và xử lý như thế nào?

Khi lên chương trình bảo mật cho mô hình, chúng ta cũng cần lên kế hoạch giải quyết các lỗi vi phạm và tình trạng mất dữ liệu. Đây là yếu tố quan trọng trong các điều khoảng của nhà cung cấp và được thực hiện bởi cá nhân. Chúng ta buộc phải đáp ứng những chính sách và điều lệ do nhà cung cấp đề ra để đảm bảo được hổ trợ kịp thời nếu gặp sự cố.

4.4.6. Ai sẽ quan sát và chịu trách nhiệm bảo đảm an toàn cho dữ liệu?

Trên thực tế thì trách nhiệm bảo mật được chia sẻ. Tuy nhiên, ngày nay vai trò này lại thuộc về hệ thống thu thập dữ liệu mà không phải nhà cung cấp. Chúng ta có thể đàm phán để giới hạn trách nhiệm đối với viêc mất mát dữ liệu cụ thể là chia sẻ vai trò này với nhà cung cấp. Nhưng cuối cùng, chúng ta vẫn là người chịu trách nhiệm.

4.4.7. Làm thế nào để chắc chắn rằng những dữ liệu phù hợp đã được chuyển vào mô hình?

Để biết được dữ liệu nào đã được chuyển vào đám mây, chúng ta phải hiểu dữ liệu là gì và xây dựng một hệ thống bảo mật phù hợp dựa trên dữ liệu và các ứng dụng. Quy trình này tốn nhiều thời gian để bắt đầu và rất nhiều công ty sử dụng công nghệ chống rò rỉ dữ liệu để phân loại và theo dõi dữ liệu.

4.4.8. Làm thế nào để chắc chắn những nhân viên, đối tác và khác hàng được ủy quyền có thể truy xuất dữ liệu và ứng dụng?

Vấn đề về quản lí thông tin truy cập và truy xuất dữ liệu là một thách thức trong bảo mật. Các công nghệ như truy cập chéo miền (federation), hệ thống ảo an toàn, và dự phòng đóng vai trò quan trọng trong bảo mật điện toán đám mây. Hổ trợ đám mây bằng cách mở rộng và bổ sung môi trường có thể giúp giải quyết thách thức này.

4.4.9. Dữ liệu và ứng dụng được đăng tải như thế nào, công nghệ bảo mật nào thưc hiện công việc này?

Các nhà cung cấp đám mây sẽ cung cấp thông tin này cũng như trực tiếp tác động đến khả năng đáp ứng các yêu cầu của một tổ chức hay cá nhân. Do đó, yếu tố rõ ràng là rất cần thiết đối với chúng ta trước khi đưa ra quyết định.

4.4.10. Yếu tố nào khiến chúng ta có thể tin tưởng vào nhà cung cấp?

Rất nhiều yếu tố đề ra để đánh gía độ tin cậy của một nhà cung cấp như: kỳ hạn dịch vụ, hình thức hợp đồng, thủ tục SLAs (Service Level Agreements) thỏa hiệp hợp đồng dịch vụ, chính sách bảo mật, tiểu sử hoạt động, chiến lược, và danh tiếng. Tuy nhiên, vẫn chưa có câu trả lời chính xác cho câu hỏi trên.

Sưu tầm

# **B.Các service của AWS**

## **1.Sơ lược về các dịch vụ cơ bản thường dùng của AWS**

|  |  |  |
| --- | --- | --- |
| **Amazon Web Service**  **Các dịch vụ cơ sở do Amazon cung cấp** | | |
| Computing  Tính toán  AWS cung cấp cho các khách hàng những dịch vụ tính toán có thể mở rộng | EC2 | Amazon Elastic Compute Cloud (AWS EC2) là 1 dịch vụ web cung cấp khả năng quản lý các tài nguyên điện toán có khả năng resizeable (tùy chỉnh kích thước – mở rộng hoặc thu hẹp các tài nguyên tính toán) theo đúng nghĩa đen,quản lý các máy chủ trung tâm dữ liệu, là nơi mà bạn sử dụng để xây dựng ứng dụng cũng như tổ chức các hệ thống phần mềm.  Dùng Amazon EC2 giúp loại bỏ nhu cầu đầu tư vào phần cứng trước tiên, vì vậy bạn có thể tập trung phát triển và triển khai ứng dụng nhanh hơn. Bạn có thể dùng Amazon EC2 để nhiều hoặc vài máy chủ ảo như bạn cần, cấu hình bảo mật và kết nối mạng, và quản lý lưu trữ. Amazon EC2 cho phép tăng/giảm nhu cầu tính toán, xử lý tùy theo nhu cầu bạn cần. |
| Lambda | AWS Lambda là 1 dịch vụ điện toán,nơi mà bạn có thể tải mã của bạn để Lambda và các dịch vụ khác có thể chạy mã của bạn bằng cách sử dụng cơ sở hạ tầng của AWS.Tất cả việc bạn cần làm đó là cung cấp mã được viết bởi 1 ngôn ngữ mà AWS Lambda hỗ trợ(nodejs , java và python), còn lại về cơ sở hạ tầng và quản lý các tài nguyên tính toán bao gồm máy chủ, hệ điều hành , khả năng cung cấp và mở rộng quy mô , giám sát và khai thác tài nguyên đều được Lambda quản lý.Điều này có nghĩa bạn ko thể đăng nhập để tùy chỉnh các hệ thống điều hành, cũng như thay đổi thời gian chạy ngôn ngữ mà phải đảm bảo về mặt mã nguồn của mình. |
| EC2 Container Service | ECS là 1 dịch vụ có khả năng mở rộng, nhanh, là dịch vụ quản lý container(thùng chứa) để nó có thể dễ dàng để chạy, dừng, và quản lý các container Docker trên một cụm các instance của Amazon EC2.  Amazon ECS tiến hành 1 cách đơn giản application trên cluster phía quản lý instance EC2, support container docker bằng dịch vụ quản lý container tính năng cao và khả năng mở rộng thông thường.  Khi sử dụng ECS thì việc tiến hành mở rộng , vận dụng , cài đặt infrastructure quản lý cluster tại công ty là không cần thiết  Khi sử dụng bằng cách gọi các API đơn giản, ứng dụng tương ứng với Docker sẽ khởi động và kết thúc, query trạng thái kết thúc của cluster và cũng có thể thực thi access đến các tính năng trong dòng  Chảy như security group , Elastic Load Balancing , EBS volume , IAM rool |
| Elastic Beanstalk | Với AWS Elastic Beanstalk, bạn có thể nhanh chóng triển khai và quản lý các ứng dụng trong các đám mây AWS mà không đáng lo ngại về cơ sở hạ tầng mà chạy các ứng dụng. AWS Elastic Beanstalk giảm độ phức tạp quản lý cũng như deploy sản phẩm bên trong AWS Cloud. Bạn chỉ đơn giản là tải lên ứng dụng của bạn, và AWS Elastic Beanstalk tự động xử lý các chi tiết của dự dung lượng dự phòng, cân bằng tải, scaling- tự động mở rộng , và quản lý việc deploy như là giám sát tình trạng ứng dụng. |
| Amazon ELB | Amazon ELB (Elastic Load Balancing) tự động phân phối lưu lượng đầu vào của ứng dụng thông qua nhiều Amazon EC2 instances (instance có thể hiểu là virtual server).  Tham khảo: <https://namvuhn.wordpress.com/2015/08/12/aws-tim-hieu-ve-ec2-elastic-load-balancing/> |
| Auto Scaling | Auto Scaling tự động mở rộng hoặc thu hẹp hiệu năng của Amazon EC2 theo những điều kiện được định nghĩa trước của khách hàng |
| Networking  Mạng - Các dịch vụ của AWS Networking giúp mở rộng các cơ sở hạ tầng của doanh nghiệp trên AWS Cloud | VPC | Amazon Virtual Private Cloud (Amazon VPC) cho phép bạn khởi động Amazon Web Services (AWS) resource vào một mạng ảo mà bạn đã định nghĩa .  Mạng ảo này gần giống như một mạng truyền thống mà bạn muốn hoạt động tại trung tâm dữ liệu của riêng bạn, với những lợi ích của việc sử dụng cơ sở hạ tầng mở rộng của AWS.  Amazon VPC cho phép bạn cung cấp một khu vực trên AWS Cloud mà tài nguyên có thể được khởi chạy trong một mạng ảo được định nghĩa bởi khách hang |
| Direct Connect | AWS Direct Connect là dịch vụ kết nối hơn 1 gigabit hoặc 10 gigabit Ethernet cáp sợi quang tiêu chuẩn đến một vị trí AWS Direct Connect mạng nội bộ của khách hàng  Một đầu của cáp được kết nối với router của bạn, khác với một router AWS Direct Connect.  Với kết nối này tại chỗ, bạn có thể tạo ra các giao diện ảo trực tiếp đến các đám mây AWS và Amazon Virtual Private Cloud, bỏ qua các nhà cung cấp dịch vụ Internet trong mạng lưới đường dẫn của bạn.  AWS Direct Connect giúp dễ dàng thiết lập kết nối mạng riêng giữa cơ sở hạ tầng của doanh nghiệp với AWS, giúp giảm thiểu chi phí, tăng băng thông, và cung cấp tính nhất quán |
| Route 53 | Route 53 là dịch vụ web DNS(hệ thống tên miền) có tính sẵn sang và mở rộng cao.  Nó chính là dịch vụ DNS, chứa dịch vụ phân giải tên miền và dịch vụ đăng ký tên miền của amazon.  Nó có 3 chức năng chính:  + Đăng ký tên miền: boi.jp  + Dịch vụ hệ thống tên miền: Tên miền là 1 định danh được đặt để thay thế cho địa chỉ ip, giúp nó trở nên dễ nhớ hơn.  Khi có request từ phía người dùng thì Route 53 sẽ giúp người dùng phân tích url, đồng thời đưa ra ip tương ứng với domain  + Kiểm tra health(sức khỏe) : Gửi request tự động đến ứng dụng thông qua internet , xác nhận đó là trong chức năng , có khả năng sử dụng , có khả năng đạt được của ứng dụng đó.  Nó chính là dịch vụ DNS, chứa dịch vụ phân giải tên miền và dịch vụ đăng ký tên miền của amazon.  Tham khảo:  <http://azsysadmin.com/dns-la-gi-gioi-thieu-ve-amazon-route-53/> |
| Analytics  Phân tích | EMR | AWS ERM là một dịch vụ web để quản lý một cách đơn giản việc phân chia lượng dữ liệu lớn một cách có năng xuất.  Tùy thuộc vào việc tổ hợp các dịch vụ đa dạng của AWS với quản lý Hadoop mà web index , data mining , phân tán file log,  Học máy, mô phỏng hóa học. Data wase housing(kho dữ liệu - keyword) được tiến hành tổ chức. |
| Data Pipeline | Data Pipeline(ống dữ liệu) là dịch vụ web được sử dụng để di chuyển và chuyển đổi dữ liệu một cách tự động.  Khi sử dụng AWS Pipeline thì bởi vì có thể định nghĩa work follow của dạng dữ liệu di động(driving force data)  Nên có thể thi hành task ban đầu khi các task trước nó kết thúc 1 cách đều đặn. |
| Kinesis | Amazon Kinesis có khả năng mở rộng một cách mềm dẻo , có dịch vụ quản lý dùng để quản lý dòng chảy big data bằng realtime.  Dịch vụ này có các dòng lưu lượng lớn của các bản ghi dữ liệu, tại EC2 instance thì ứng dụng quản lý dữ liệu của nhiều hoạt động sẽ  có thể tiêu thụ dữ liệu từ các dòng lưu lượng đó bằng realtime. Ứng dụng quản lý dữ liệu được gọi là “Amazon Kinesis Application”, sử  dụng Amazon Kenesis client library |
| Machine Learning | Amazon Machine Learning là dịch vụ có thể sử dụng một cách đơn giản các kĩ thuật học máy bất kể nhà phát triển có năng lực như thế nào.  Học mãy là 1 lĩnh vực của trí tuệ nhân tạo liên quan đến việc nghiên cứu và xây dựng các kĩ thuật cho phép các hệ thống học tự động từ tập dữ liệu để giải quyết những vấn đề cụ thể.  Bằng việc sử dụng Amazon Machine Learning, bạn có thể tạo ra một loạt các ứng dụng tiên đoán.  Sử dụng Amazon Machine Learning, gắn cờ vào các ứng dụng giao dịch đáng ngờ, các ứng dụng để phát hiện các đơn đặt hàng bất hợp pháp,  các ứng dụng để dự đoán nhu cầu, ứng dụng cá nhân hóa nội dung, lọc các ứng dụng, xem xét để dự đoán các ứng dụng hoạt động người dùng,  ứng dụng để theo dõi các phương tiện truyền thông xã hội, các ứng dụng để phân tích các văn bản miễn phí,  sẽ giúp bạn xây dựng các ứng dụng như vậy để giới thiệu sản phẩm. |
| Elastic  Search | Elasticsearch có thể phân phối tìm kiếm và phân tích trong thời gian thực, nó là engine rất phù hợp trong môi trường điện toán đám mây.  Trong tài liệu định hướng, việc định nghĩa các lược đồ là không cần thiết, hỗ trợ tìm kiếm một cách linh hoạt các query có cấu trúc, không có cấu trúc,Time-series, nó cũng hoạt động như 1 ứng dụng nền tảng của Visualization bao gồm các ứng dụng khác và đặc biệt là kibana |
| Development  Tools | Code Commit | AWS CodeCommit là một dịch vụ kiểm soát version(phiên bản) tồn tại riêng tư ,cho phép bạn lưu trữ và quản lý kho Git trong đám mây AWS. |
| Code Deploy | AWS CodeDeploy là một dịch vụ triển khai cho phép các nhà phát triển để tự động hóa việc triển khai, các ứng dụng để thể hiện và cập nhật các ứng dụng theo yêu cầu. |
| Code Pipeline | AWS CodePipeline là một dịch vụ phân phối liên tục,cho phép bạn mô hình hóa , hình dung và tự động hoá các bước cần thiết để phát hành phần mềm của bạn. |
| Managerment  Tools | Cloud  Watch | CloudWatch là 1 dịch vụ web cho phép bạn thu thập, hiển thị và phân tích số liệu.  CloudWatch sẽ theo dõi AWS Resource cùng với các ứng cụng của bạn chạy trên AWS trong suốt thời gian thực. Bạn có thể sử dụng CloudWatch để thu thập,  theo dõi số liệu , đó là các biến mà bạn muốn đo cho tài nguyên và ứng dụng của mình.  CloudWatch gửi thông báo hoặc tự động thay đổi resource mà bạn đang theo dõi được đựa trên quy tắc mà bạn đặt ra.  Vd: Bạn có thể giám sát việc sử dụng CPU và đọc,ghi đĩa của instance Amazon EC2.Sau đó bạn sử dụng dữ liệu này để xác định xem bạn có nên khởi động thêm  Các instance khác để tăng tải xử lý. Bạn cũng có thể sử dụng số liệu để xác định có loại bỏ các instance để tiết kiệm tiền.  Ngoài việc nắm bắt các số liệu được xây dựng đi kèm với AWS thì bạn có thể theo dõi các số liệu tùy chỉnh của riêng bạn.  Với CloudWatch bạn sẽ có cái nhìn hệ thống về việc sử dụng tài nguyên, hiệu suất ứng dụng cũng như tình trạng hoạt động.  AWS CloudWatch về cơ bản là 1 kho lưu trữ số liệu |
| Cloud Formation | AWS CloudFormation là dịch vụ để dự phòng “các nguồn tài nguyên được hình thành bằng các bộ sưu tập resource AWS liên quan tới nhau và quan trọng với việc triển khai, vận dụng” một cách thích hợp theo thứ tự  AWS CloudFormation cho phép bạn tạo và cung cấp cơ sở hạ tầng AWS cho các triển khai(deploy) dự đoán và liên tục. Nó sẽ giúp bạn tận dụng các sản phẩm AWS như Amazon EC2, Amazon Elastic Block Store, Amazon SNS, Elastic Load Balancing, và Auto Scaling để xây dựng, mở rộng khả năng của các ứng dụng với độ tin cậy cao và chi phí hiệu quả bên trong cloud mà không lo lắng về việc tạo ra và cấu hình AWS cơ bản cơ sở hạ tầng. AWS CloudFormation cho phép bạn sử dụng một tập tin mẫu để tạo ra và xóa một tập hợp các resource với nhau như một đơn vị duy nhất (một stack). |
| Cloud  Trail | Với AWS CloudTrail, bạn có thể giám sát việc triển khai AWS của bạn trong các đám mây bằng cách nhận được một lịch sử cuộc gọi API của AWS cho tài khoản của bạn,  bao gồm cả các cuộc gọi API thực hiện thông qua Management Console AWS, SDK AWS, các công cụ dòng lệnh, và các dịch vụ AWS cấp cao hơn |
| Config | AWS Config cung cấp một cái nhìn chi tiết về việc các tài nguyên liên kết với tài khoản AWS của bạn, bao gồm cách chúng được cấu hình, làm thế nào chúng có liên quan đến nhau, và làm thế nào các cấu hình và các mối quan hệ của chúng đã thay đổi theo thời gian. |
| OpsWorks | AWS OpsWorks cung cấp một cách đơn giản và linh hoạt để tạo và quản lý ngăn xếp và các ứng dụng.  Với AWS OpsWorks, bạn có thể cung cấp AWS Resource , quản lý cấu hình của chúng , deploy application với resource đó , và theo dõi trạng thái ứng dụng |
| Service Catalog | AWS Dịch vụ Catalog cho phép các quản trị viên IT tạo, quản lý danh mục đầu tư và phân phối các sản phẩm đã được phê duyệt cho người dùng, sau đó người dùng có thể truy cập vào các sản phẩm mà họ cần trong một cổng thông tin cá nhân kết thúc. Sản phẩm tiêu biểu bao gồm máy chủ, cơ sở dữ liệu, các trang web, hoặc các ứng dụng được triển khai sử dụng các AWS Resource (ví dụ, một instance Amazon EC2 hoặc một cơ sở dữ liệu Amazon RDS). Bạn có thể kiểm soát người dùng thực thi quyền truy cập vào các sản phẩm cụ thể phù hợp với các tiêu chuẩn kinh doanh của tổ chức, quản lý vòng đời sản phẩm, và giúp người sử dụng tìm kiếm và ra mắt sản phẩm với sự tự tin. |
| Trusted Advisor |  |
| Security and Identity | IAM  Identity & Access Management | AWS Identity và Access Manager (IAM) là một dịch vụ web cho phép khách hang của Amazon Web Services (AWS) quản lý người dùng và cho phép người dùng sử dụng trong AWS.  Dịch vụ này là mục tiêu của tổ chức với nhiều người dùng hoặc hệ thống trong các đám mây　sử dụng sản phẩm AWS như Amazon EC2, Amazon SimpleDB, và Management Console AWS. |
| Directory Service | Các dịch vụ thư mục AWS là một dịch vụ được quản lý mà làm cho nó dễ dàng để kết nối tại chỗ hiện có Microsoft Active Directory của bạn, hoặc để thiết lập và vận hành một thư mục mới trong các đám mây AWS, làm cho nó dễ dàng để triển khai và quản lý khối lượng công việc của Windows trong các đám mây AWS. Thư mục người dùng của bạn và các nhóm có thể truy cập Management Console AWS, và các ứng dụng AWS, chẳng hạn như không gian làm việc Amazon và Amazon WorkDocs, bằng các thông tin hiện có của họ. |
| Storage & Content Delivery  Lưu trữ  Các dịch vụ lưu trữ trên AWS | S3 | AWS S3 là 1 kho lưu trữ sử dụng mạng internet. Nếu sử dụng AWS S3 thì bạn có thể lấy được lượng dữ liệu như bạn mong muốn tại bất cứ lúc nào , từ bất cứ nơi nào trên web.  Những việc này có thể được thự thi bằng cách sử dụng 1 giao diện web trực quan trên công cụ AWS Managerment đơn giản.  Nói cho dễ hiểu thì nó chính là ổ cứng di động trên mây, dùng để chứa dữ liệu  Amazon S3, cung cấp một cơ sở hạ tầng cho việc lưu trữ toàn bộ những dữ liệu dự phòng, nó cho phép chúng ta lưu trữ và truy xuất tới bất kỳ khối dữ liệu nào, ở bất kỳ thời điểm nào và từ bất kỳ đâu trên Web |
| Cloud Front | AWS Cloud Front là một dịch vụ web gửi tốc độ cao nội dung tĩnh và động(html , css ,php,..) của web đến người dùng cuối.  Cloud Front gửi nội dung thông qua internet toàn cầu của edge location. Nghĩa là nếu người dùng ở nhật muốn truy cập vào boi.us có host đặt tại mỹ, như thế nó sẽ phải liên tục tải dữ liệu từ mỹ về nhật, rất mất thời gian và tốn dung lượng. AWS CloudFront hỗ trợ bằng cách sẽ lưu lại tất cả nội dung mà người dùng mong muốn(đã truy cập ) tại host aws tại nhật để rút ngắn khoảng cách địa lý, nên tốc độ sẽ nhanh hơn. Với lại nó chỉ lưu lại nội dung content người dùng cần chứ không phải lưu lại toàn bộ nội dung web app nên sẽ không tốn về mặt dung lượng bộ nhớ.  Tham khảo:  <http://chiasecoupon.com/kinh-nghiem/cach-dung-amazon-cloudfront-lam-cdn.html>  Keywork: CDN |
| EBS | Amazon Elastic Block Store (EBS) cung cấp block level storage volumes (nhiều thuật ngữ dịch ra tiếng việt sẽ gây khó hiểu và nhiều khi không nói lên được đúng ý nghĩa của từ/cụm từ đó nên cho phép mình để nguyên) cho việc sử dụng với Amazon EC2 instances  Mỗi kiểu máy chủ ảo sẽ được cấp kiểu vi xử lý và lượng RAM thích hợp, tuy nhiên trong cấu hình sẽ không có ổ cứng. Để ảo hóa ổ cứng, một công nghệ khác của Amazon được sử dụng – đó là EBS. Giả sử người dùng cần 25 GB, điều đó sẽ được đáp ứng ngay. Bất kỳ lúc nào, người dùng cũng có thể yêu cầu cung cấp thêm dung lượng. Loại ổ cứng như thế được gọi là Volume và được kết nối với Instance. Vậy là trong hệ thống đã có ổ cứng. Tất cả dữ liệu ghi lên nó sẽ được lưu trữ không phụ thuộc vào thời gian sống của Instance. |
| Glacier | Amazon Glacier là dịch vụ lưu trữ với chi phí rất thấp cho phép lưu trữ lâu dài với các tính năng bảo mật dành cho lưu trữ và sao lưu dữ liệu. Với Amazon Glacier, khách hàng lưu dữ liệu với chi phí hiệu quả cho từng tháng, năm và thậm chí cả thế kỷ. Amazon Glacier là sự lựa chọn lưu trữ tuyệt vời khi chi phí lưu trữ thấp là tối quan trọng, dữ liệu được truy xuất không thường xuyên và độ trễ truy xuất vài giờ là có thể chấp nhận. Nếu ứng dụng của bạn yêu cầu truy xuất nhanh đến dữ liệu, độ trễ thấp, xem xét dùng Amazon S3.  Amazon Glacier là một dịch vụ lưu trữ với chi phí cực kỳ thấp cung cấp một kho lưu trữ bảo mật và bền vững cho việc lưu trữ và sao lưu dữ liệu  Tham khảo: <http://dinhnn.com/2014/11/22/bat-dau-voi-amazon-glacier/> |
| Storage Gateway | AWS Storage Gateway là một dịch vụ cho phép kết nối một phần mềm on-premise với hệ thống lưu trữ trên Cloud một cách liên tục và bảo mật |
| AWS Import/  Export | AWS Import/Export cho phép đẩy nhanh tiến độ di chuyển một lượng dữ liệu lớn vào ra AWS sử dụng các thiết bị lưu trữ di động. |
| Database  Cơ sở dữ liệu  Các dịch vụ AWS Database | RDS  Amazon Relational Database Service | Amazon RDS:là 1 dịch vụ có thể dễ dàng mở rộng ,thu nhỏ, setup dữ liệu quan hệ trong cloud.  Dịch vụ này vừa tiến hành thay thế khách hang quản lý database tốn công sức , vừa cung cấp dung lượng có khả năng thay đổi theo kích thước , hiệu xuất cao mà giá cả tốt. Nhờ vào điều này mà khách hang có thể tập trung cho kinh doanh và phát triển ứng dụng của bản thân.  Chạy độc lập với EC2 dùng để quản lý csdl  Amazon RDS, một web services cho phép thiết lập, hoạt động và mở rộng cơ sở dữ liệu quan hệ trên Cloud |
| DynamoDB | Amazon DynamoDB, một dịch vụ cơ sở dữ liệu NoSQL, có hiệu suất cao, hoàn toàn được quản lý cho phép dễ dàng thiết lập, hoạt động và mở rộng |
| ElastiCache | Amazon ElastiCache là dịch vụ có thể thực thi 1 cách đơn giản việc scaling(nhân rộng), vận dụng , deploy bộ nhớ cache trong đám mây.  Một hình thức giống như clouldfront nhưng cloudfront dùng cache để tang tốc ở phía người dùng cuối, còn elasticache dùng ở giai đoạn xử lý, giúp người dùng giảm thiểu thời gian truy vấn dữ liệu bằng cách sử dụng memcached và redis (lưu trữ dữ liệu trên bộ nhớ đệm RAM làm cho tốc độ truy xuất rất nhanh)  Amazon ElasticCache, một web service dễ dàng để triển khai, hoạt động, và mở rộng một memory cache trên cloud. |
| Redshift | Amazon Redshift, một dịch vụ cơ sở dữ liệu quan hệ được thiết kế để lưu trữ dữ liệu trung tâm trong một tổ chức. |
| Mobile Service | Cognito | Amazon Cognito là một dịch vụ đơn giản mà cung cấp khả năng cho danh tính người dùng và dữ liệu đồng bộ hóa, giúp an toàn cũng quản lý việc đồng bộ hóa dữ liệu ứng dụng của người sử dụng giữa các thiết bị di động |
| Device Farm | AWS Device Farm là một dịch vụ thử nghiệm ứng dụng cho phép bạn thử nghiệm ứng dụng mobile trên cloud với môi trường tương ứng cung cấp bởi AWS |
| Mobile Analytics | Amazon Mobile Analytics là 1 dịch vụ giúp cho nhà phát triển có thể lĩnh hội, nắm vững 1 cách đơn giản việc tưởng tượng sau khi thu thập dữ liệu tình trạng sử dụng của ứng dụng theo quy mô lớn |
| SNS | Amazon SNS là một dịch vụ web cho phép các ứng dụng, người dùng, và các thiết bị để ngay lập tức gửi và nhận các thông báo từ các đám mây.  Tạo ra để người dùng có thể theo dõi ứng dụng đó. Nên khi có sự thay đổi nào đó thì nó sẽ thông báo đến người dùng. |
| Application  Service  Các dịch vụ ứng dụng của hệ sinh thái AWS | API Gateway | API Gateway là 1 dịch vụ quản lý đầy đủ và dễ dàng đối với các developer để công bố, duy trì , theo dõi và đảm bảo cho các API ở bất kì mở rộng nào.(Nó là dịch vụ quản lý các API cho dù ở bất kì mở rộng nào). Tạo 1 API để truy cập dữ liệu, logic kinh doanh hoặc các function từ dịch vụ back-end của bạn, chẳng hạn như các ứng dụng chạy trên EC2, mã chạy trên Amazon Lambda hoặc ứng dụng web bất kỳ |
| AppStream | Các dịch vụ web Amazon AppStream triển khai ứng dụng của bạn trên cơ sở hạ tầng Amazon Web Services và các luồng đầu vào và đầu ra giữa các ứng dụng và các thiết bị của bạn như máy tính cá nhân, máy tính bảng và điện thoại di động.Mọi hoạt động của ứng dụng của bạn sẽ xảy ra trong cloud nên nó có thể tự mở rộng để xử lý tải tính toán lớn. Và các thiết bị chỉ cần hiển thị đầu vào và đầu ra trả lại người dùng, do đó mà các ứng dụng client trên các thiết bị có thể nhẹ hơn về kích thước tập tin cũng như việc xử lý các yêu cầu. |
| CloudSearch | Là 1 dịch vụ ứng dụng tìm kiếm. Amazon CloudSearch là một dịch vụ hoàn toàn được quản lý trong các đám mây giúp cho nó có thể dễ dàng để thiết lập, quản lý, quy mô và giải pháp tìm kiếm cho trang web của bạn. Amazon CloudSearch cho phép bạn tìm kiếm các dữ liệu lớn như các trang web, các tập tin tài liệu, bài trên diễn đàn, hoặc các thông tin sản phẩm. Với Amazon CloudSearch, bạn có thể nhanh chóng thêm các khả năng tìm kiếm với trang web của bạn  mà không cần phải trở thành một chuyên gia trong lĩnh vưc tìm kiếm hoặc lo lắng về việc cung cấp phần cứng, cài đặt và bảo trì. Khi khối lượng của dữ liệu lớn và dao động, Amazon CloudSearch sẽ tự động quy mô để đáp ứng nhu cầu của bạn. |
| Elastic Transcoder | Amazon Elastic Transcoder cho phép bạn chuyển đổi các tập tin media mà bạn đã lưu trong Amazon S3 thành các tập tin media theo các định dạng theo yêu cầu của các thiết bị phát của người tiêu dùng.  Ví dụ, bạn có thể chuyển đổi lớn, các tập tin media kỹ thuật số chất lượng cao sang các định dạng mà người dùng có thể phát lại trên các thiết bị di động, máy tính bảng, các trình duyệt web,... |
| SES | Amazon Simple Email Service(Amazon SES) là một dịch vụ outbound-only email-sending, cung cấp một cách hiệu quả và dễ dàng cho bạn để gửi email. |
| SQS | Amazon Simple Queue Service (Amazon SQS) là một dịch vụ hàng đợi tin nhắn để xử lý tin nhắn hoặc công việc giữa các thành phần khác trong hệ thống. |
| SWF | Dịch vụ Amazon Simple Workflow (Amazon SWF) làm cho nó dễ dàng xây dựng ứng dụng phối hợp công việc giữa các thành phần phân tán.  Trong Amazon SWF, một task được đại diện cho một đơn vị logic của công việc và nó được thực hiện bởi một thành phần của ứng dụng của bạn. Phối hợp các nhiệm vụ trên các ứng dụng liên quan đến việc quản lý phụ thuộc intertask, lịch trình, và đồng thời phù hợp với dòng chảy logic của ứng dụng. Amazon SWF cung cấp cho bạn toàn quyền kiểm soát thực hiện nhiệm vụ và phối hợp chúng mà không cần lo lắng về sự phức tạp tiềm ẩn như theo dõi sự tiến bộ của họ và duy trì trạng thái của họ. |
| Enterprise Applications | WorkSpaces | Amazon WorkSpace cung cấp cho bạn dễ dàng có một trải nghiệm máy tính cloud-based desktop trong vai trò người dùng cuối. Bạn chỉ cần chọn từ một lựa chọn các gói cung cấp một loạt các thông số khác nhau của CPU, bộ nhớ, lưu trữ, và một sự lựa chọn của các ứng dụng.Người dùng có thể kết nối từ một máy tính, máy tính để bàn Mac, iPad, Kindle, hoặc máy tính bảng Android. |
| WorkDocs | Amazon WorkDocs là quản lý đầy đủ, lưu trữ doanh nghiệp an toàn và dịch vụ chia sẻ với các điều khiển hành chính mạnh mẽ và khả năng phản hồi để cải thiện năng suất người dùng. Tập tin của bạn được lưu trữ trong đám mây, một cách an toàn và an toàn. Amazon WorkDocs thậm chí bao gồm một ứng dụng đồng bộ mà luôn luôn giữ các thư mục được lựa chọn trên máy tính địa phương của bạn đồng bộ với các thư mục trên điện toán đám mây của bạn. File của bạn chỉ hiển thị cho bạn, và người đóng góp hoặc là những người xem được bạn chỉ định. Các thành viên khác của tổ chức của bạn không có quyền truy cập vào bất kỳ tập tin của bạn trừ khi bạn cấp cho họ truy cập.  VD: Giống với Google Driver hoặc là OneDriver |
| WorkMail | Amazon WorkMail là dịch vụ quản lý email và lịch với hệ thống điều khiển bảo mật mạnh mẽ , hỗ trợ cho máy tính desktop và các mobile email client. Bạn có thể truy cập email, địa chỉ liên lạc, và lịch tại bất cứ nơi nào bạn sử dụng Microsoft Outlook, trình duyệt của bạn, hoặc các thiết bị iOS hay các thiết bị di động Android. Bạn có thể tích hợp Amazon WorkMail với thư mục của công ty hiện tại của bạn, cũng như kiểm soát cả các mã hóa dữ liệu của bạn và vị trí nơi dữ liệu được lưu trữ. |

**In-Memory Database**: Cơ sở dữ liệu In-Memory ( IMDB; hay còn gọi là hệ thống cơ sở dữ liệu trên bộ nhớ chính hay MMDB) là một hệ quản trị dữ liệu chủ yếu đặt trên bộ nhớ chính của máy tính.

Đối lập với hệ quản trị được đặt trên bộ nhớ đĩa. IMDB nhanh hơn vì thuật toán bên trong đơn giản hơn và thực thi đơn giản hơn.

Việc truy cập dữ liệu trong hệ quản trị này giảm thiểu được các hoạt động đọc I/O khi truy vấn dữ liệu, điều này làm cho các thực thi nhanh hơn và hiệu quả hơn.

Với các ứng dụng đòi hỏi về mặt thời gian như viễn thông hay di động thì hệ quản trị In-Memory hay được sử dụng.

Các hệ quản trị cơ sở dữ liệu khác thì dữ liệu được đặt trong ổ đĩa cứng.

Tham khảo:

+ <http://forums.bsdinsight.com/threads/in-memory-computing-la-gi.5637/>

+ <http://www.pcworld.com.vn/articles/cong-nghe/cong-nghe/2011/06/1225702/imds-cho-doi-hoi-truy-xuat-nhanh/>

## **2.Một số dịch vụ cần nắm vững**

### **2.1.EC2**

Một bài viết kĩ về AWS EC2:

+ <http://www.ibm.com/developerworks/vn/library/ar-cloudaws3/>

**2.1.1. EC2 là gì ?**

Có những tình huống không lường trước được của thế giới web ngày nay, khi mà lưu lượng truy cập tới trang web của bạn có thể tăng đột biến như sóng thần nếu trang web của bạn được xuất hiện trên trang nhất của Yahoo — và giảm đột ngột sau đó vài giờ. Bây giờ, bạn có thể điều chỉnh dung lượng lưu lượng của bạn lên hay xuống một cách mềm dẻo.”

Amazon EC2 cung cấp khả năng mở rộng tinh toán trong AWS cloud. Amazon EC2 giúp loại bỏ nhu cầu đầu tư vào phần cứng, do đó bạn có thể phát triển và khai thác các ứng dụng nhanh hơn. Bạn có thể sử dụng Amazon EC2 để khởi động 1 hoặc nhiều máy chủ ảo theo nhu cầu. Cấu hình bảo mật , kết nối mạng, quản lý và lưu trữ.Ta bắt tay vào tìm hiểu Amazon EC2, khi tìm hiểu EC2 bạn sẽ thu được những hiểu biết các thanh phần cở sở hạ tầng cơ bản của nó.

“Amazon EC2 là dịch vụ Web cho phép bạn yêu cầu các máy ảo trong vòng một vài phút và dễ dàng thay đổi khả năng tính toán theo nhu cầu của bạn. Bạn chỉ cần phải trả chi phí cho khoảng thời gian mà bạn sử dụng. Nếu bạn muốn tăng khả năng tính toán của bạn nên, bạn có thể nhanh chóng tiếp cận khởi tạo máy áo và sau đó chấm dứt chúng khi mà nhu cầu của bạn giảm đi.”

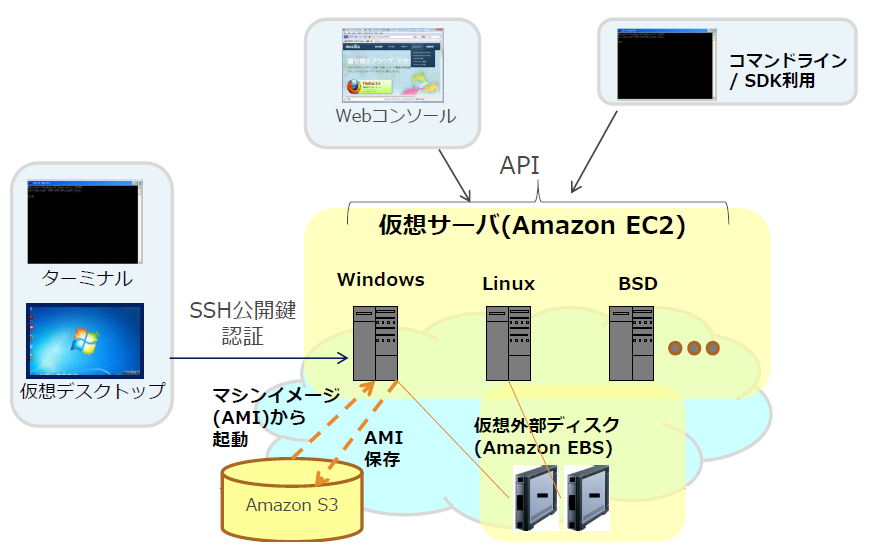
“Amazon Elastic Compute Cloud (EC2) là nền tảng cơ sở cho môi trường điện toán đám mây do Amazone cung cấp. **EC2 giúp cho việc tạo ra, khởi động và dự phòng các ứng dụng ảo cho cá nhân hay doanh nghiệp một cách đơn giản và bất cứ khi nào bạn cần. Bạn trả phí sử dụng cho những ứng dụng này theo loại ứng dụng và thời lượng sử dụng**. Các máy chủ ảo hoạt động bên trong môi trường bảo mật của các trung tâm dữ liệu của Amazon.”

Có thể nói AWS EC2 là dịch vụ quan trọng bậc nhất trong các service mà Amazon cung cấp.

**EC2 có thể cung cấp cho ứng dụng ảo của bạn khả năng để:**

**+ Cấu hình các yêu cầu tính toán của chúng ngay trong khi hoạt động.**

**+ Điều chỉnh dung lượng dựa trên yêu cầu.**

****

**2.1.2. Đặc điểm của EC2 và Các từ khóa cần biết trong EC2**

Các tinh năng EC2 cung cấp:

+ **Instance:** Một môi trường điện toán ảo hóa(môi trường tính toán), ta có thể hiểu đơn giản nó như 1 server (server là máy tính có cấu hình mạnh dùng làm máy chủ web, hoặc máy chủ db,...)

+ **Amazon Machine Images(AMI)**: Các mẫu(template) đã được cấu hình sẵn(định nghĩa trước) cho instance.Nó cung cáp các gói phần mềm cần thiết cho máy chủ bao gồm thông tin về hệ điều hành,các packages,các software đã được cài đặt vào instance.

+ **Instance Types**: là các cấu hình khác nhau của cpu, memory , storage , networking cho instances tùy theo mục đích tính toán, lưu trữ dữ liệu,..

+ **Key Pair**: Thông tin đăng nhập an toàn cho instance bằng cách sử dụng cặp khóa(key pairs). AWS sẽ lưu giữ khóa công khai(public key), còn bạn khi bạn khởi tao instance sẽ được cung cấp 1 key – khóa bí mật(private key) và bạn cất giữ nó tại 1 nơi an toàn.

+ **Instance Store Volumes**: Lưu trữ dữ liệu tạm thời của instance trong quá trình instance đó hoạt động , và nó sẽ bị xóa khi dừng(stop) hoặc hủy(terminate) instance

+ **Amazon EBS Volumes**: Nó sử dụng dịch vụ Amazon Elastic Block Store(Amazon EBS) của Amazon để lưu trữ dữ liệu của instance lâu dài, vĩnh viễn. Nó sẽ không bị mất ngay cả khi bạn stop hoặc terminate instance.

+ **Regions and Availability Zones**: Mỗi Region là một địa điểm vật lý ở thế giới, nơi mà Amazon có nhiều Availability Zones. Hiện tại Amazon có các Region ở: Bắc Mỹ, Nam Mỹ, Châu Âu, Trung Đông, Châu Phi và Châu Á Thái Bình Dương. Một Availability Zone (AZ) thông thường là một Data Center đơn lẻ, mặc dù chúng có thể được xây dựng bởi gộp của nhiều Data Center. Các AZ được tách biệt đủ để chúng có nguồn điện và kết nối internet riêng.Kết nối duy nhất giữ các AZ là một mạng có độ trễ thấp. Mạng kết nối có thể nhanh đủ để ứng dụng của bạn hoạt động như thể tất cả các AZ đang ở cùng một data center.

+ **Security Groups**: Một bức tường lửa cho phép bạn chỉ định các giao thức, các cổng và các địa chỉ IP để có thể truy cập đến instance trên môi trường internet.

+ **IP Elastic**: IP của instance.

+ **Virtual Private Cloud(VPC)**: Cũng như 1 mạng thông thường , VPC chính là môi trường internet giành cho instance của bạn. Nó là mạng ảo mà bạn có thể tạo ra một cách hợp lý, tách biệt với các phần còn lại của AWS,và tùy chọn kết nối đến mạng riêng của mình.

+ **Metadata (Tags)**: là tag mà bạn có thể tạo và gán cho Amazon EC2 resources.

Nếu vẫn chưa thực sự hiểu những định nghĩa trừu tượng trên thì trong phần dưới, bài viết nói một cách dễ hiểu về EC2 cũng như các thành phần của nó.

Công nghệ EC2, về bản chất chính là đưa khái niệm “điện toán đám mây” vào thực tế. Với sự trợ giúp của EC2, người sử dụng có thể cài đặt trên các “đám mây” một lượng máy tính bất kỳ, với cấu hình và hệ điều hành tùy theo nhu cầu, và chỉ cần vài phút để làm được điều đó. Mỗi một máy tính ảo như vậy gọi là một Instance. Sau khi khởi động máy (thường là một vài phút) người dùng sẽ có ngay quyền truy cập root thông qua SSH hoặc quyền truy cập tới desktop thông qua giao thức RDP (Remote Desktop Protocol), tùy thuộc vào hệ điều hành mà người dùng chọn. Phí sử dụng Instance như vậy được tính theo giờ, có nghĩa là người dùng có thể tắt máy chủ ảo (virtual server) của mình bất kỳ lúc nào và sẽ không phải trả thêm phí.

Mỗi kiểu máy chủ ảo sẽ được cấp kiểu vi xử lý và lượng RAM thích hợp, tuy nhiên trong cấu hình sẽ không có ổ cứng. Để ảo hóa ổ cứng, một công nghệ khác của Amazon được sử dụng – đó là EBS. Giả sử người dùng cần 25 GB, điều đó sẽ được đáp ứng ngay. Bất kỳ lúc nào, người dùng cũng có thể yêu cầu cung cấp thêm dung lượng. Loại ổ cứng như thế được gọi là Volume và được kết nối với Instance. Vậy là trong hệ thống đã có ổ cứng. Tất cả dữ liệu ghi lên nó sẽ được lưu trữ không phụ thuộc vào thời gian sống của Instance.

Công nghệ tiếp theo là S3, cũng là để lưu trữ file nhưng là một mảng hoàn toàn khác. Có thể nói, đó là một “kho” vô hạn để chứa file và khi cần, có thể truy cập tới các file này thông qua web. Ngườidùng sẽ được cấp không gian nhớ đúng theo nhu cầu: 10 MB, 1 GB, 5.000 GB hay thậm chí là không có một giới hạn nào cả, ngoài giới hạn đối với kích thước của file là 5 GB.

Trong phạm vi chương trình khuyến mại “AWS Free Usage Tier”, mỗi người dùng mới sẽ được thử các dịch vụ này miễn phí. “Thử” có nghĩa là tài nguyên miễn phí sẽ được cung cấp hạn chế (khoảng 750 giờ sử dụng Instance EC2, 10 GB cho EBS và 5GB cho S3). Nếu người dùng muốn sử dụng ngoài giới hạn thì phải trả phí. Như vậy, chúng ta đã có thể có được một máy chủ để thử nghiệm, máy chủ này làm việc dựa trên công nghệ điện toán đám mây. Những gì có thể làm với máy chủ này giờ đây chỉ còn tùy thuộc vào người dùng. Nhưng phương án thú vị nhất có lẽ là cài đặt một máy chủ VPN của mình và đặt nó ở tại nước Mỹ.

Tham khảo:

+ <https://viblo.asia/HjkaruBloom/posts/PjxMeVbNG4YL>

+ <https://viblo.asia/HjkaruBloom/posts/pVYRPj1KG4ng>

+ <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/get-set-up-for-amazon-ec2.html>

+ <http://antoanthongtin.vn/Detail.aspx?NewsID=51a89846-be3d-4938-9f15-621a3a17bb98&CatID=762c71a3-7540-4718-964e-b2bced858291>

Tham khảo:

+ <https://namvuhn.wordpress.com/2015/08/04/aws-tim-hieu-ve-dich-vu-amazon-ec2/>

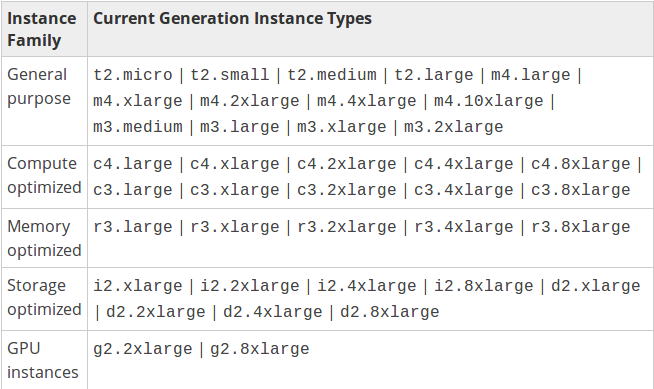
+ <http://codingsoup.com/2014/02/aws-tu-khi-dang-ky-dich-vu-den-hoan-thanh-website-phpapache/>

Tiếp theo mình sẽ đi sâu vào instance, các loại instance và instance lifecycle.

* + 1. **Instance Type**

Trước hết hãy nói về những loại instance. Amazon cung cấp rất nhiều loại instance khác nhau tùy thuộc vào mục đích sử dụng của bạn như cần server mạnh về tính toán, server mạnh về memory, hay đơn thuần chỉ là một server có khả năng lưu trữ nhiều dữ liệu.

Dưới đây là hình ảnh phân loai instance tùy theo mục đích sử dụng.



Có tất cả 5 dòng instance được cung cấp bao gồm:

+ General purpose: là dòng instance có sự cân bằng giữa 3 tiêu chí là cpu, memory, storage. Dòng này bao gồm có 2 dòng nhỏ là dòng t và m. Dòng t là những instance bình thường cấu hình không mạnh lắm. Trong khi đó dòng m lại là những instance có cấu hình khá mạnh như m4.2xlarge với vCPU là 8, 32Gb memory hay m4.10xlarge với 40 vCPU và 160Gb memory.

+ Compute optimized: dòng c với việc tối ưu hóa khả năng tính toán khi cung cấp số lương virtual process vCPU

+ Memory optimized: dòng r với việc tăng cường memory cho instance. Ví dụ như con r3.8xlarge có memory là 244Gb.

+ Storage optimized: dòng i với việc tối ưu hóa khả năng lưu trữ dữ liệu

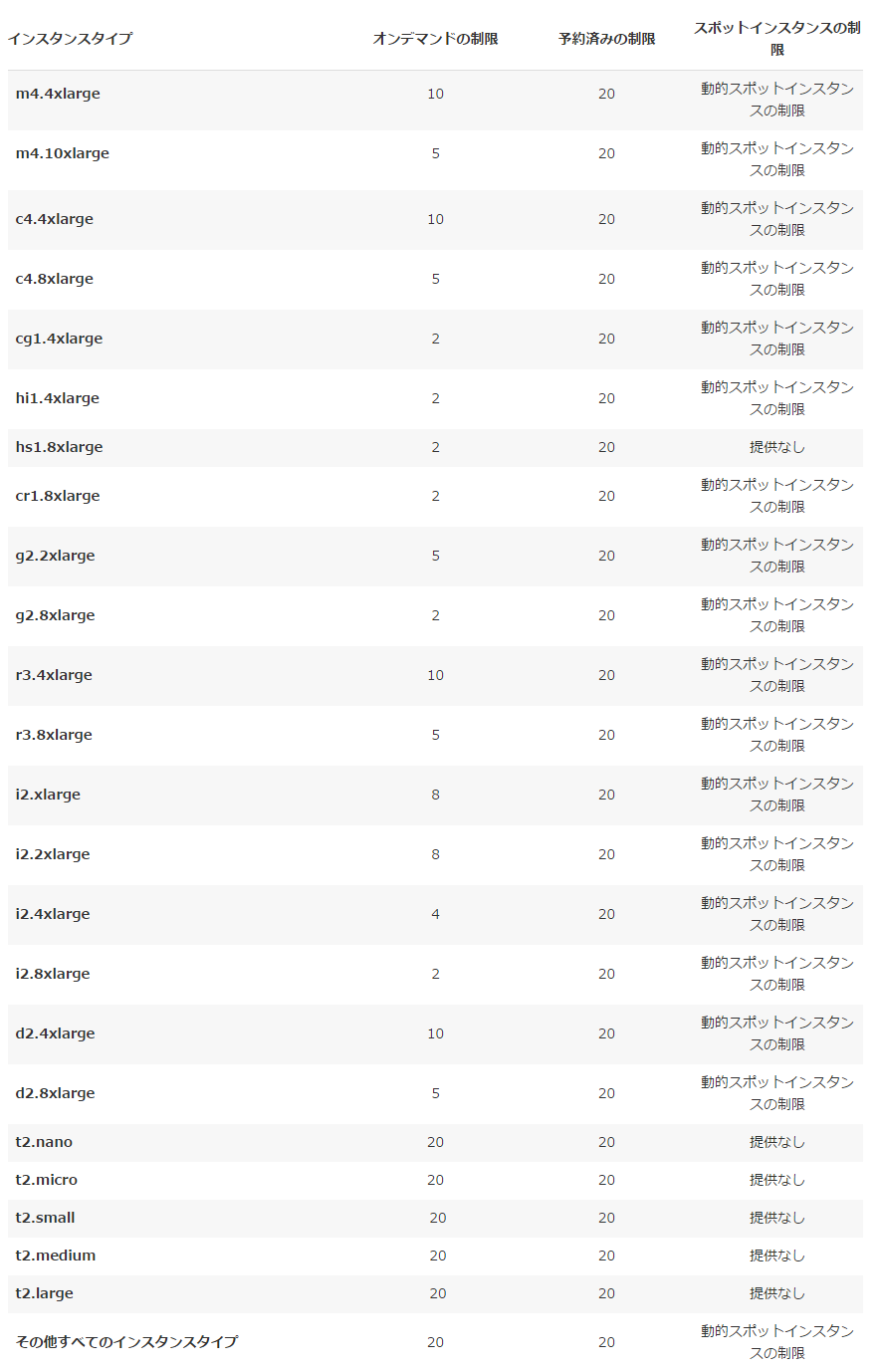
+ GPU instance: là instance có khả năng làm việc với app xử lý về graphic với việc cung cấp 1 đến 4 GPUs.

Dựa trên những loại trên thì tùy thuộc vào mục đích sử dụng của bạn và số tiền bạn có để có thể tạo instance phù hợp với app. Để tìm hiểu về cấu hình cụ thể của từng loại instance bạn có thể tham khảo linh dưới dây:

+ <http://aws.amazon.com/ec2/instance-types/>

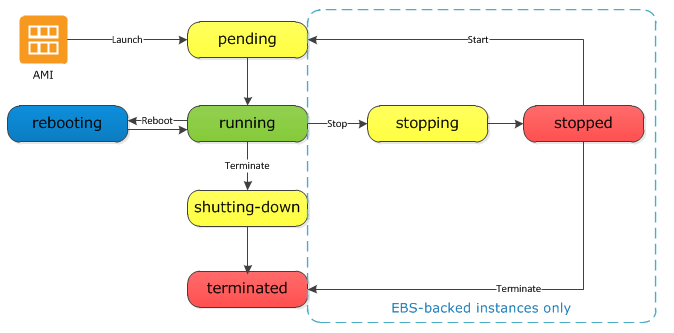
Ngoài ra còn một điểm cần lưu ý là bạn không thể tạo bao nhiêu instance cũng được. Mỗi account thì lại có giới hạn số lượng instance của từng loại mà bạn có thể tạo. Tham khảo trong link dưới đây:

+<http://aws.amazon.com/ec2/faqs/#How_many_instances_can_I_run_in_Amazon_EC2>



* + 1. **Instance Lifecycle**

Tiếp theo chúng ta sẽ tìm hiểu về lifecycle của một instance.



Instace khi được khởi tạo sẽ ở trạng thái pending. Sau đó sử dụng instance type để thiết lập cấu hình hardware. Tiếp đó instace sử dụng AMI với những thông tin cần thiết để boot instance. Sau khi instance đã sẵn sàng để chạy, nó sẽ chuyển sang trạng thái running.

Nếu như root device volumn được lưu ở Amazon EBS, instance có thể được stop và start lại. Khi stop instance đầu tiên nó sẽ chuyển về trạng thái stopping và sau đó là stopped. Khi instance ở trạng thái stopped, bạn có thể thay đổi thông tin, thuộc tính của instance kể cả instance type. Sau đó bạn có thể start lại instance. Có những điểm cần lưu ý ở đây là:

* Khi instance ở trạng thái running tất nhiên là bạn phải trả tiền cho usage và data transfer của instance. Tuy nhiên khi instance stopped thì bạn không phải trả tiền cho usage và data transfer của instance. Tuy nhiên nếu bạn sử dụng EBS Volume để back up volume thì bạn vẫn phải trả tiền cho phần này kể cả khi instance đã stopped.
* Khi start instance thì đầu tiên instance sẽ được chuyển về trạng thái pending. Ở đấy instance sẽ được chuyển về host computer mới. Do đó tất cả dữ liệu đươc lưu trữ ở instance store volumes sẽ mất.
* Khi restart instance nếu instance chạy trên EC2-Classic, nó sẽ có địa chỉ ip mới. Còn nếu chạy trên EC2-VPC instance vẫn có địa chỉ IP cũ.

Reboot instance cũng chính là reboot lại operating system. Điều này không ảnh hưởng đến data, public DNS name hay ip address của instance, tất cả đều được giữ nguyên. Tuy nhiên Amazon có recommend bạn nên reboot instance trên màn hình giao diện thay vì làm trên command. Chắc là vì trên giao diện mới thấy hết lỗi nếu có xảy ra .

Khi mà bạn không cần đến instance nào đó nữa thì bạn có thể tắt nó đi, chuyển về trạng thái terminated. Có 2 điểm cần lưu ý ở đây:

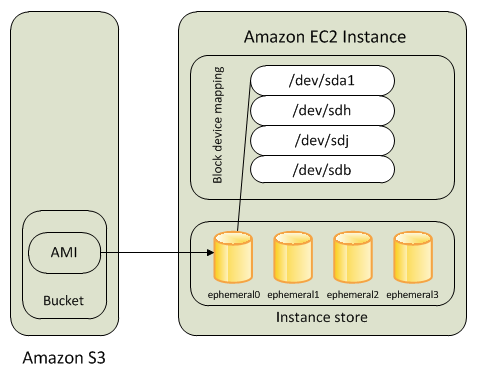
* Nếu bạn khởi động tính năng termination protection, bạn sẽ không thể tắt nó bằng command.
* Amazon EBS volume chứa một thuộc tính là DeleteOnTermination (kiểu như dependent: :destroy ở mongoid ấy) tức là volume sẽ tự động bị xóa khi mà instance khởi tạo từ volumn đó về trạng thái terminated.

Ngoài ra còn một trạng thái nữa mà trên sơ đồ không có là trạng thái Instance Retirement. Instance sẽ về trạng thái này khi mà xảy ra lỗi không thể sửa về phía hardware của instance. Nếu instance root device sử dụng Amazon EBS volume thì sau trạng thái retirement sẽ là stopped, tức là bạn có thể start lại instance. Còn nếu sử dụng instance store volumn, instance sẽ được chuyển về trạng thái terminated và không thể start lại.

* + 1. **EC2 Root Device Volume**

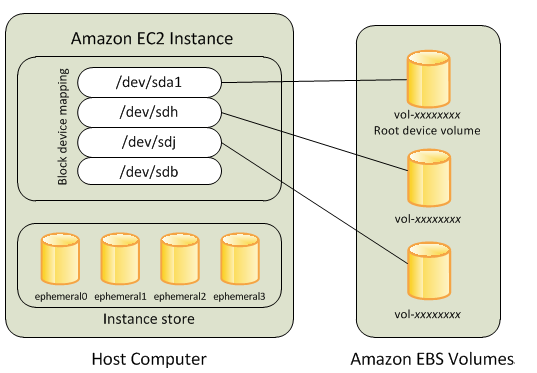
Root Device Volume chứa những image cần thiết để boot instance khi bạn khởi tạo instance. Như đã nói ở trên, những images này (AMIs) sẽ được lưu trữ ở instance store (Instance Store-backed) hoặc Amazon EBS (Amazon EBS-backed) nên chúng ta sẽ thấy Root Device Volume sẽ chứa mapping đến một trong 2 store này.

Hình ảnh dưới đây sẽ giúp bạn hiểu rõ về instance store-backed



Như bạn thấy image được sử dụng để boot instance được copy từ s3 vào root volume (đặc biệt là với sda1). Do đó khi instance dừng hoạt động thì những data này sẽ bị mất khi instance về trạng thái terminated.

Còn đối với EBS-backed sơ đồ sẽ như dưới đây:



Khi mà bạn khởi tạo instance, Amazon sẽ tạo cho bạn một Amazon EBS volumes tương ứng và lưu từng volume của instance ở đây. Do đó bạn có thể stop start instance mà không phải lo việc mất data. Ngoài ra với khi instance mà Amazon EBS volumes attach đến chuyển trạng thái về terminated, Amazon EBS volumes đó sẽ bỏ attach đến instance đó và có thể được attach vào một instance khác.

Việc sử dung Amazon EBS-backed giúp cho việc khởi tạo instance được nhanh hơn, cũng như giúp cho việc lưu trữ data an toàn và ổn định hơn.

Trong bài viết trước mình đã giới thiệu về EC2, về instance, các loại instance hay là instance lifecycle. Bài viết này mình sẽ tiếp tục giới thiệu về Amazon Machine Images để giúp các bạn hiểu rõ hơn về service EC2 của Amazon.

* + 1. **Amazon Machine Images**

AMI là một template chứa các thông tin để khởi tạo một instance như hệ điều hành, packages, software... mà bạn muốn nó có khi tạo instance.

Một AMI có thể được sử dụng để khởi tạo nhiều instance và ngược lại, một instance có thể sử dụng nhiều AMI để khởi tạo bản thân.

Một mẫu AMI sẽ bao gồm có:

* Template cho root volume của instance ví dụ như hệ điều hành (OS), application server, hay những application cần thiết...
* Launch Permissions: khai báo về quyền khởi tạo instance. Dùng để xác định những tài khoản nào được sử dụng AMI này để khởi tạo instance.
* Block Device Mapping: chỉ định volumes nào sẽ được map với instance khi instance khởi tạo.

1. **AMI Type**

Phân loại AMI thì chúng ta có rất nhiều tiêu chí ví dụ như:

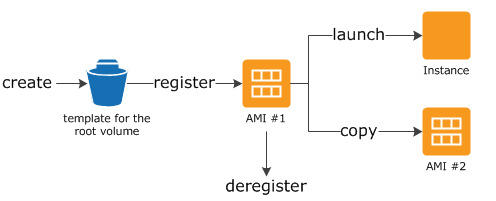
* Region: chia theo khu vực mà AMI được sử dụng, giống như region của instance. Bao gồm: ap-northeast-1, ap-southeast-1, ap-southeast-2...
* Operating System: chia theo hệ điều hành. Linux, Window...
* Architecture: 32-bit hay 64-bit
* Lauch Permission: quyền được truy cập và sử dụng AMI. Bao gồm có:
  + public: cho phép truy cập vào sử dụng co tất cả AWS accounts.
  + explicit: cho phép một vài AWS accounts được truy cập và sử dụng.
  + implicit: chỉ có owner của AMI mới được sử dụng.
* Storage cho Root Device: chia theo nơi lưu trữ Root Device. Lưu ở Amazon EBS - Backed hay là Amazon Instance Store - Backed

Trong những tiêu chí trên thì tiêu chí Storage cho Root Device là cái mà chúng ta cần quan tâm nhất vì nó ảnh hưởng lớn nhất đến việc khởi động, lưu trữ dữ liệu... của instance. Chúng ta sẽ xem bảng dưới đây để hiểu rõ hơn về sự khác nhau giữa Amazon EBS - Backed và Amazon Instance Store - Backed.

|  |  |  |
| --- | --- | --- |
| Đặc điểm | Amazon EBS-Backed | Amazon Instance Store-Backed |
| Thời gian Boot | Thường ít hơn 1 phút | Thường ít hơn 5 phút |
| Giới hạn dữ liệu | 16TiB | 10TiB |
| Upgrade Instance | Có thể thay đổi instance type, kernel, ram... khi instance stop | Không thể thay đổi thuộc tính của instance |
| Phí tổn | Tính phí cho phần sử dụng instance, sử dụng Amazon EBS volume và lưu trữ AMI | Tính phí cho phần sử dụng instance và lưu AMI trên S3 |
| Tạo AMI | Chỉ cần sử dụng 1 câu command | Yêu cầu cài đặt và sử dụng AMI tools |
| Stop State | Instance có thể đưa về stop state | Instance không thể đưa về stop state |

1. **AMI Lifecycle**

Một chu kì sử dụng của AMI mô tả đơn giản như hình bên dưới:



Đầu tiên là sẽ tạo AMI sử dụng template cho root volume có sẵn. Sau đó là đăng kí (register) cho AMI. Sau khi đăng kí thì AMI đó đã sẵn sàng để tạo một hay nhiều instance. Và nếu không cần sử dụng AMI đó thì chúng ta sẽ bỏ đăng kí (deregister) AMI..

**b.1. Tạo AMI**

Việc tạo AMI hay customize một AMI thường là sử dụng AMI có sẵn của Amazon để tạo instance. Thay đỗi những setting, cài đặt thêm aplication hay những thứ cần thiết trên instance và sau đó là sử dụng instance đó để tạo AMI. Để xem list những AMI public thì có thể sử dụng AWS CLI với câu lệnh sau:

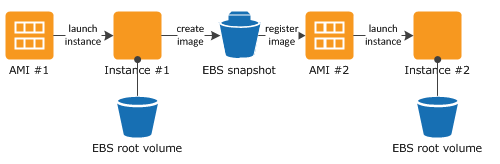
>$ aws ec2 describe-images --executable-users all # list những AMI public

>$ aws ec2 describe-images -executable-users self # list những AMI bạn tạo

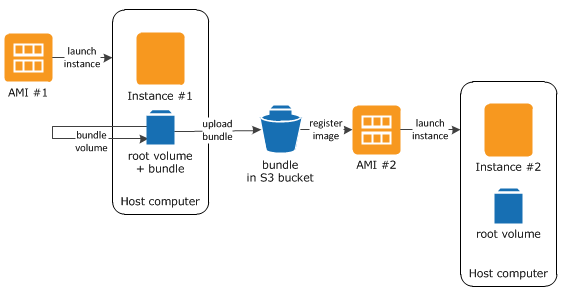
>$ aws ec2 describe-images --owners amazon # list AMI của Amazon

Dưới đây là 2 sơ đồ cho việc tạo Amazon EBS-Backed AMIs và Amazon Instance Store-Backed AMIs

* Amazon EBS-Backed AMIs



* Amazon Instance Store-Backed



Dựa trên sơ đồ chúng ta có thể thấy được sự khác nhau giữa việc tạo Amazon EBS-Backed AMIs và Amazon Instance Store-Backed AMIs.

Đối với Amazon EBS-Backed AMIs thì template sẽ được lưu ở EBS là 1 snapshot (snapshot hiểu đơn giản như là image của thằng Norton Ghost Image mà bạn hay dùng để backup toàn bộ data của Win ấy =)) Từ snapshot này một AMI sẽ được register.

Còn với Amazon Instance Store-Backed AMIs sau khi tạo instance, bạn phải sử dụng tool để bundle. Sau khi process kết thúc sẽ sinh ra một bundle bao gồm 1 file image image.manifest.xml và 1 file template image.part.xx. Sau đó bundle này được upload lên S3 từ đó register 1 AMI cho bạn.

Nếu như ở Amazon EBS-Backed AMIs bạn phải trả tiền cho phần lưu snapshot ở EBS thì ở Amazon Instance Store-Backed AMIs bạn phải trả tiền cho phần lưu bundle trên S3.

Ngoài ra đối với Amazon Instance Store-Backed AMIs bạn cần chú ý những điều sau:

* Cài đặt AMI tools
* Cài đặt API tools
* Đảm bảo rằng bạn có S3 bucket để lưu bundle

Cụ thể về các bước tạo AMIs với 2 loại trên bạn có thể đọc và làm theo ở 2 link sau, rất dễ sử dụng =))

+ <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-instance-store.html#creating-ami-linux-instance>

+ <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html#how-to-create-ebs-ami>

Từ 1 snapshot để tạo 1 AMI thì bạn tham khảo theo link sau:

+ <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html#creating-launching-ami-from-snapshot>

Ngoài ra còn 1 điểm cần chú ý đó là bạn có thể chuyển một Amazon Instance Store-Backed AMIs thành một Amazon EBS-Backed AMIs theo các bước như link dưới đây. Tuy nhiên chỉ có thể chuyển với OS là Linux, không thể chuyển với OS là Window.

+ <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-instance-store.html#Using_ConvertingS3toEBS>

**b.3. Deregister AMIs**

Nếu như AMI không cần thiết nữa, bạn có thể bỏ đăng kí AMI đó đi. Việc bỏ đăng kí AMI không ảnh hưởng đến snapshot hay bundle tạo từ AMI đó cũng như instance tạo từ AMI đó. Tuy nhiên là vì bạn vẫn phải trả tiền cho phần lưu snapshot, bundle và chạy instance nên thường thì khi bỏ đăng kí cho AMI, bạn cũng nên xóa snapshot, bundle và terminate instance tạo từ AMI đó.

Để deregister AMI thì bạn sử dụng câu lệnh sau ` \_r\_Default`` ec2-deregister ami\_id

### **2.2. VPC**

Việc tích hợp tốt giữa điện toán đám mây(Cloud Computing) và mạng riêng ảo(Virtual Private Network) của doanh nghiệp đã tạo nên 1 khái niệm mới là “đám mây riêng ảo” (Virtual Private Cloud - VPC).

VPC là một cầu nối an toàn và thông suốt giữa cơ sở hạ tầng hiện có của doanh nghiệp với các đám mây AWS.Amazon VPC cho phép các doanh nghiệp kết nối cơ sở hạn tầng hiện có của họ với tập hợp các tài nguyên tính toán của AWS thông qua kết nối VPN(mạng riêng ảo) giúp cho các doanh nghiệp có thể quản lý các dịch vụ bảo mật, tường lửa và các hệ thống phát hiện xâm nhập đến các tài nguyên AWS của họ.

**2.2.1. Mạng riêng ảo (VPN)**

Mạng riêng ảo (hay Virtual Private Network) là một mạng thường dành riêng để kết nối các máy tính trong nhà hay trong một tổ chức với nhau thông qua Internet. Với VPN, bạn có thể thiết lập kết nối từ xa để duyệt web an toàn hơn, tạo mạng ảo để cùng bạn bè chơi game như khi chơi bằng mạng LAN mà chúng ta không phải ở gần nhau, truy cập các tập tin được chia sẻ,...

Một VPN là một mạng bao gồm các máy tính được kết nối với nhau ngay cả khi chúng nằm ở những vị trí địa lí cách xa nhau, hoặc khi chúng sử dụng các phương pháp kết nối mạng khác nhau. VPN thường được dùng để giúp các thiết bị ở xa truy cập vào một mạng riêng nào đó (ví dụ, mạng LAN nội bộ của công ty) thông qua kết nối Internet công cộng. Một khi đã truy cập thành công thì máy đó cũng trở thành một phần của LAN y như khi bạn đang ở công ty. Và bởi vì phải dùng kết nối công cộng nên VPN tạo ra các "đường ống" (tunneling) kết hợp với cơ chế mã hóa để truyền tải dữ liệu giữa thiết bị ở xa với mạng riêng nhằm đảm bảo an toàn. Các gói dữ liệu đi trong đường ống này sẽ không bị xem trộm bởi những tác nhân xấu bên ngoài.

Lợi ích lớn nhất của VPN đó là tất cả kết nối giữa các máy tính đều được mã hóa và có độ an toàn cao. Một lợi ích nữa đó là các máy tính trong VPN có thể giao tiếp với nhau một cách hiệu quả, giống như khi chúng cùng được kết nối với một router vậy. Nếu khó hiểu, bạn hãy tưởng tượng rằng hai máy tính được kết nối với nhau bởi một cái ống nước, và những gì chạy trong cái ống này sẽ không thể thoát ra ngoài. Chỉ có hai máy chơi với nhau mà thôi.

Nếu bạn không quan tâm lắm đến vấn đề bảo mật, bạn có thể thuê hoặc dùng các dịch vụ VPN, nhưng tại sao chúng ta lại phải phí tiền cho những công ty khác trong khi bạn có thể tự xây dựng riêng một VPN cho mình? Việc này rất dễ dàng, và khi đã có một VPN của riêng mình, bạn có thể làm nhiều thứ, từ lướt web, chia sẻ tập tin nhạc, hình ảnh giữa các máy cho đến việc chơi game với bạn bè mà không cần phải ngồi chung trong một căn phòng. Thật tuyệt vời!

Tham khảo:

+ <https://sites.google.com/site/rocbin/vpn-la-gi>

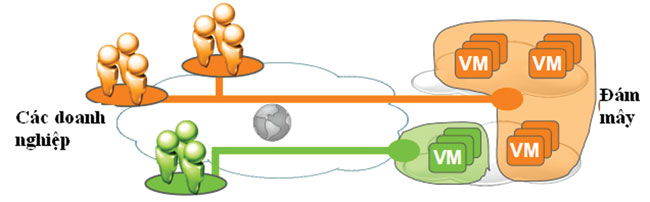
+ <https://tinhte.vn/threads/huong-dan-thiet-lap-mang-rieng-ao-vpn-cua-rieng-ban-tren-windows-va-mac.1197292/>

+ <http://quantrimang.com/tim-hieu-ve-virtual-private-network-vpn-va-tunneling-84202>

**2.2.2. Thử thách với mạng riêng ảo khi kết hợp với đám mây**

DN sử dụng DV đám mây sẽ dễ dàng tạo các máy ảo (Virtual Machine - VM), nền tảng để chạy các ứng dụng. Tuy nghiên, khi một số thành phần của ứng dụng bị rớt ra ngoài do kết nối hoặc do bị tấn công thì DN sẽ gặp rất nhiều khó khăn. Rủi ro về bảo mật và độ chính xác dữ liệu trên đám mây rất lớn: mất dữ liệu, dữ liệu bị nhầm lẫn (hoặc trộn lẫn) với dữ liệu của khách hàng khác, thông tin bị ăn cắp… Để bảo mật thông tin và đảm bảo tính độc lập giữa mức hệ thống mạng và máy chủ DV, các DN thường muốn tự quản lý cơ sở hạ tầng mạng hơn là đi thuê dùng trên các đám mây. DN muốn các địa chỉ IP là của mình và các nhân viên của DN khi sử dụng tài nguyên cũng phải được đặt ở một mức độ ưu tiên nhất định.

Bên cạnh đó, động lực để đám mây phát triển chính là việc các tài nguyên vật lý có thể được kết nối hoặc gỡ bỏ dễ dàng trên đám mây. Các nhà cung cấp đám mây phải kết hợp với các nhà quản lý mạng thiết lập động trên các máy chủ, tường lửa, định tuyến, lưu trữ…



Tham khảo:

+ <http://www.pcworld.com.vn/articles/kinh-doanh/quan-tri/2010/06/1219383/dien-toan-may-ket-hop-mang-rieng-ao/>

**2.2.3. VPC**

Amazon = Đám mây + Mạng riêng ảo (Cloud + Virtual Private Network)

VPC sử dụng VPN để tạo các kênh kết nối an toàn với tài nguyên của doanh nghiệp.

Amazon Virtual Private Cloud (Amazon VPC) cho phép bạn xác định một mạng ảo của riêng bạn(bạn đã tự định nghĩa).Mạng ảo này gần giống như một mạng truyền thống mà bạn muốn hoạt động tại trung tâm dữ liệu của riêng bạn, với những lợi ích của việc sử dụng cơ sở hạ tầng mở rộng của AWS.

Bắt đầu bằng bài viết tại trang chủ của AWS:

<http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/VPC_Introduction.html#Overview>

Định nghĩa: VPC là 1 mạng riêng ảo đã được định nghĩa mà trong đó ta có thể khởi động các tài nguyện AWS.Mạng ảo này giống hệt với các mạng mà bạn đã thao tác từ trước đến giờ tại các trung tâm dữ liệu của bạn nhưng nó có ưu điểm là có cấu trúc hạ tầng co dãn(スケーラブルなインフラストラクチャ)

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Khái niệm của Amazon VPC

+ Phương pháp bắt đầu sử dụng VPC

+ Sử dụng VPC bằng các dịch vụ khác của AWS

+ Truy cập tới VPC

+ Giải phóng và xóa bỏ VPC

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

**a.VPCの概念**

Khi bắt đầu với Amazon VPC thì bạn cần hiểu những khai niệm quan trọng của mạng ảo này, và những thanh phần nào giống-khác nhau với mạng truyền thống.

+ Amazon VPC là networking layer của Amazon EC2.

**a.1.VPC và Subnetの概念**

+ VPC là 1 mạng ảo riêng(virtual private cloud) dành riêng cho tài khoản AWS chuyên dụng .Nó được phân lập(tách ra và cô lập) về mặt logic(論理的に切り離される) từ các mạng ảo khác trong đám mây AWS. Bạn có thể cài đặt các tài nguyên của AWS như là EC2 instance vào bên trong VPC của bạn. Bạn có thể config cho VPC ví dụ như: bạn có thể chọn phạm vi địa chỉ IP cho VPC , tạo ra mạng con, và cấu hình bảng định tuyến, cổng mạng, và thiết lập bảo mật

+ Subnet là phạm vi của địa chỉ IP của VPC(Theo định nghĩa thì subnet là 1 phần của 1 mạng nào đó và sẽ được hình thành bằng cách sử dụng 1 mặt nạ mạng dài hơn – Tham khảo slide của thầy Sơn để hiểu rõ hơn). Bạn có thể khởi tạo các tài nguyên AWS(AWSリソース) bên trong một subnet mà bạn đã lựa chọn. Hãy sử dụng public subnet với các tài nguyên cần kết nối internet và sử dụng private subnet cho các tài nguyên không cần kết nối internet.Để hiểu rõ hơn về VPC và Subnet thì đọc tiếp phần a.1.1)

+ Để bảo vệ nguồn tài nguyên AWS trong mỗi subnet thì bạn có thể sử dụng security group , danh sách kiểm soát truy cập(ネットワークアクセスコントロールリスト- ACL) , hoặc các security layer.

Tham khảo: <https://tinhtoandammay.wordpress.com/page/2/>

**a.1.1 VPC とサブネット**

Bài viết dịch từ : <http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/VPC_Subnets.html#vpc-subnet-diagram>

Với khái niệm như trên, bạn đã có một VPC và thiết lập trong đó các tài nguyên AWS mà bạn muốn. Khi thành lập VPC,bạn chỉ định tập hợp các địa chỉ IP cho mạng ảo đó bằng hình thức trong các khối Class Inter-Domain Routing(CIDR). Ví dụ 10.0.0.0/16

+ Lược đồ địa chỉ CIDR: CIDR là một lược đồ địa chỉ mới cho internet, nó cho phép sử hiệu quả hơn tài nguyên IP hơn là mô hình địa chỉ chia làm các lớp A,B,C cũ. Đại khái là nó sắp xếp lại địa chỉ IP để hạn chế sự lãng phí của IP không được sử dụng theo mô hình 3 lớp cũ.

Tham khảo: <http://vietbao.vn/Vi-tinh-Vien-thong/Luoc-do-dia-chi-CIDR-la-gi/20486521/217/> và <http://www.vnpro.org/forum/forum/ccent%C2%AE/icnd-1-osi-model-basic-tcp-ip/14688-kh%C3%B4ng-hi%E1%BB%83u-b%E1%BA%A3n-ch%E1%BA%A5t-c%E1%BB%A7a-cidr-classless-interdomain-routing>

**a.1.1.1. VPC mới (新しいVPC)**

Trước khi khởi động 1 instance trong VPC thì việc thêm subnet vào là cần thiết.

**a.1.1.2.Thiết lập Size của VPC**

Có thể dùng 1 khối CIDR để phân chia VPC. Từ /28 netmask đến /16 netmask thì trong không gian của khối CIDR đó, VPC có thể chứa từ 16 đến 65536 địa chỉ IP. VPC đã được khởi tạo thì không thể thay đổi kích thước của nó. Trong trường hợp VPC quá nhỏ thì ta có thể tạo VPC mới có kích thước lớn hơn và chuyển instance qua đó.Để làm được việc đó thì từ instance đang thực thi, ta sẽ hình thành AMI tương ứng với nó và tại VPC mới khởi tạo, ta thực hiện việc khởi động lại 1 instance khác từ AMI đó. Và sau đó ta sẽ đóng instance cũ lại và xóa VPC kia đi.

**a.1.1.3.Kết nối tới Local Network và VPC khác**

**a.1.1.4.Khởi tạo VPC**

**a.1.1.5.Xóa VPC**

**a.2. Phương pháp bắt đầu sử dụng VPC**

**a.3　Sử dụng VPC bằng các dịch vụ khác của AWS**

**a.4　Truy cập tới VPC**

**a.5　Giải phóng và xóa bỏ VPC**

### **2.3.Route 53**

Là dịch vụ làm chức năng đăng ký và quản lý tên miền Domain Name Service

### **2.4.S3**

Trong bài viết này, tìm hiểu thêm về dịch vụ lưu trữ đơn giản của Amazon, Amazon Simple Storage Service (S3). S3 là hệ thống lưu trữ dữ liệu nhanh và có thể mở rộng được trên internet giúp bạn lưu trữ và lấy dữ liệu một cách đơn giản với bất cứ dung lượng nào, bất cứ khi nào và bất cứ đâu trên thế giới. Bạn trả phí dựa trên dung lượng lưu trữ và băng thông mà bạn dùng. Không phí cài đặt, không phí tối thiểu hay phí phụ trội.

Amazon cung cấp sự quản lý và duy trì của nền tảng lưu trữ, giúp bạn thoải mái tập trung vào các chức năng quan trọng của hệ thống và ứng dụng của bạn. S3 là nền tảng công nghiệp mà luôn sẵn sàng cho nhu cầu lưu trữ dữ liệu của bạn. Nó tuyệt vời cho:

+ Lưu trữ dữ liệu cho các ứng dụng của bạn.

+ Sao lưu dữ liệu cho doanh nghiệp hoặc cá nhân.

+ Phân tán nhanh và rẻ các nội dung làm tiêu hao băng thông lớn tới khách hàng của bạn.

Tính năng của S3 bao gồm:

+ Tính ổn định

Nó được thiết kế để chịu được các hỏng hóc và phục hồi hệ thống rất nhanh với thời gian tối thiểu. Amazon cung cấp một thỏa thuận cấp dịch vụ (service-level agreement - SLA) để duy trì tính sẵn sàng ở mức 99.99 phần trăm.

+ Đơn giản, dễ dùng

S3 được xây dụng trên các khái niệm đơn giản và cung cấp tính mềm dẻo cao cho việc phát triển các ứng dụng của bạn. Bạn có thể xây dựng các lược đồ lưu trữ phức tạp hơn, nếu cần, bằng cách thêm các hàm vào các thành phần của S3.

+ Tính mở rộng

Thiết kế của S3 cung cấp một cấp độ cao về tính mở rộng và cho phép sự điều chỉnh dễ dàng trong dịch vụ khi lượng truy cập vào ứng dụng web của bạn tăng đột biến với lưu lượng khổng lồ.

+ Rẻ

Chi phí sử dụng S3 rất cạnh tranh với các giải pháp của công ty và cá nhân khác trên thị trường.

Ba khái niệm nền tảng cho khung làm việc S3 là thùng (buckets), đối tượng (objects), và khóa (keys).

Thùng (Buckets)

Thùng là các nền tảng cơ bản nhất. Mỗi đối tượng được lưu trữ trong Amazon S3 đều được chứa trong một thùng. Thùng có thể được hiểu như là một thư mục trên hệ thống tệp. Điểm mấu chốt để phân biệt giữa thư mục và thùng là mỗi thùng và các nội dung của nó có thể truy cập được thông qua URL. Ví dụ, nếu bạn có một thùng tên là "prabhakar," nó có thể truy cập được thông qua URL http://prabhakar.s3.amazonaws.com.

Mỗi tài khoản S3 có thể có tối đa 100 thùng. Một thùng không thể chứa thùng khác, nên bạn không thể tạo thùng trong thùng được. Bạn có thể thay đổi vị trí địa lý của các thùng của bạn bằng cách chỉ rõ một vị trí khi bạn tạo chúng. Điều này sẽ đảm bảo một cách tự động rằng tất cả các đối tượng mà bạn lưu trữ trong thùng đó sẽ được lưu trữ trong cùng vị trí địa lý. Tại thời điểm này, bạn có thể đặt các thùng của bạn ở Mỹ hoặc liên minh châu Âu. Nếu bạn không chỉ rõ một vị trí khi tạo thùng, thùng đó và nội dung của nó sẽ được lưu trữ tại vị trí gần địa chỉ thanh toán trên tài khoản của bạn nhất.

Các tên của thùng cần tuân theo các yêu cầu của S3:

Tên phải bắt đầu bằng một số hoặc một chữ cái.

Độ dài của tên tối thiểu là 3 và dài nhất là 255.

Một tên có hiệu lực có thể chỉ chứa các ký tự viết thường, chữ số, dấu chấm, dấu gạch chân, và gấu gạch ngang.

Mặc dù tên có thể có chữ số và dấu chấm, nhưng chúng không được sử dụng định dạng số IP. Bạn không thể có một thùng với tên là 192.168.1.254.

Không gian tên thùng được chia sẻ giữa các thùng trong tất cả các tài khoản trong S3. Vì vậy, tên của bạn phải là duy nhất trong S3.

Các thùng mà sẽ chứa các đối tượng mà sẽ được truy cập bởi các địa chỉ URLs phải tuân theo các yêu cầu phụ của S3 như sau:

Tên của các thùng không được chứa dấu gạch dưới.

Độ dài từ 3 đến 63 ký tự.

Tên không thể kết thúc với một dấu gạch ngang. Ví dụ, tên myfavorite-.bucket.com là không hợp lệ.

Không chứa dấu gạch ngang đứng ngay trước dấu chấm. Tên my-.bucket.com là không hợp lệ.

Bạn có thể sử dụng một quy ước đặt tên miền cho các thùng của bạn, như là media.yourdomain.com, và ánh xạ các tên miền và tên miền con web sẵn có của bạn tới Amazon S3. Việc ánh xạ thực sự sẽ được hoàn tất khi bạn thêm DNS CNAME để trỏ tới S3. Lợi ích to lớn với lược đồ này là bạn có thể sử dụng chính tên miền của bạn trong các địa chỉ URL của bạn để tải các tệp. Ánh xạ CNAME sẽ chịu trách nhiệm cho việc chuyển đổi giữa các địa chỉ S3 cho các thùng của bạn. Ví dụ, http://media.yourdomain.com.s3.amazonaws.com trở thành địa chỉ thân thiện hơn <http://media.yourdomain.com>.

Đối tượng (Objects)

Các đối tượng chứa dữ liệu được lưu trữ trong các thùng ở S3. Các đối tượng được coi như là các tệp mà bạn muốn lưu trữ. Mỗi đối tượng được lưu trữ được cấu thành bởi hai thực thể: dữ liệu (data) và dữ liệu thông tin (metadata). Dữ liệu là dữ liệu thực mà bạn bạn muốn lưu trữ, như là tệp PDF, tài liệu Word, một tệp video, vân vân. Dữ liệu lưu trữ được gắn với dữ liệu thông tin để mô tả đối tượng. Một vài ví dụ của dữ liệu thông tin như là kiểu của đối tượng được lưu trữ, ngày chỉnh sửa cuối cùng, và bất cứ thông tin cụ thể nào khác. Dữ liệu thông tin cho một đối tượng được chỉ ra bởi người phát triển như là cặp khóa giá trị (key value pairs) khi đối tượng đó được gửi tới S3 để lưu trữ.

Không giống như sự giới hạn về số lượng các thùng, số lượng đối tượng không bị hạn chế. Bạn có thể lưu trữ một số lượng vô hạn các đối tượng trong các thùng của bạn, và mỗi đối tượng có thể chứ đến 5GB dữ liệu.

Dữ liệu trong các đối tượng S3 công cộng của bạn có thể lấy được qua HTTP, HTTPS, hoặc BitTorrent. Phân tán các tệp nghe nhìn lớn từ tài khoản S3 của bạn sẽ trở lên rất đơn giản khi sử dụng BitTorrent; Amazon không chỉ tạo torrent cho đối tượng của bạn, mà còn lưu trữ (seed) nó.

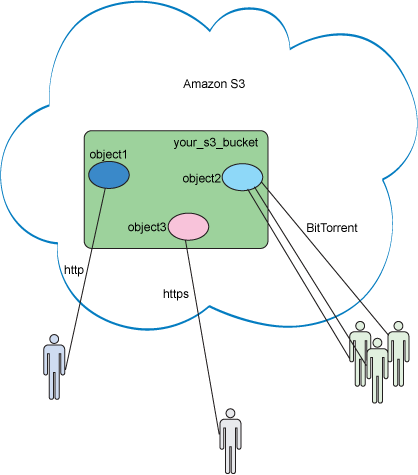
Khóa

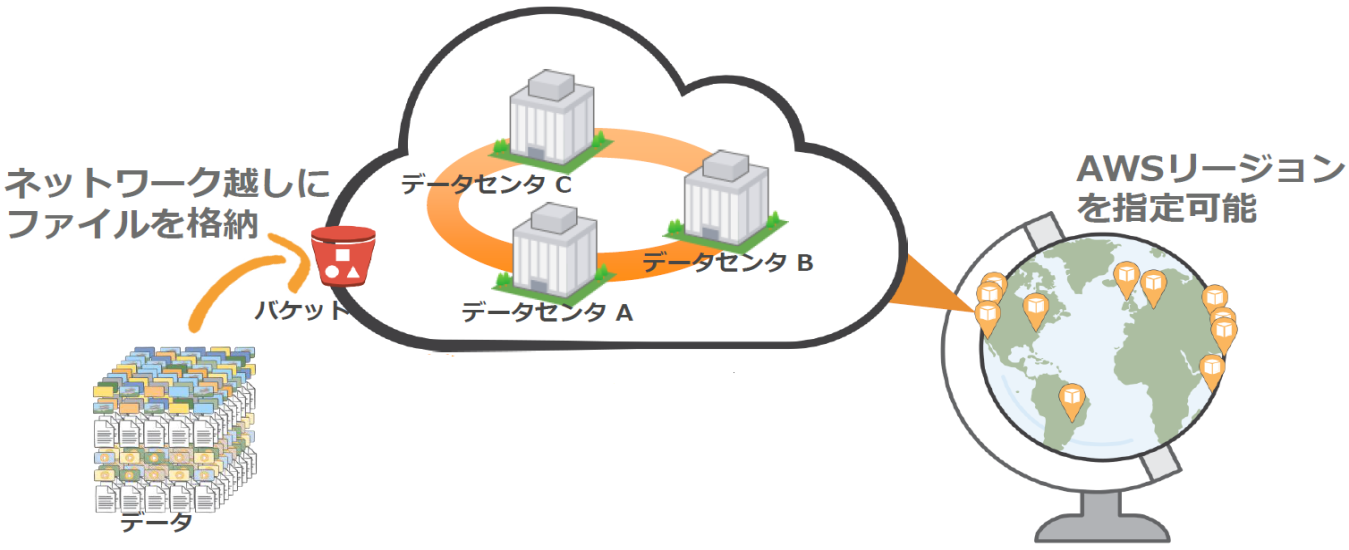
Mỗi đối tượng lưu trữ trong một thùng S3 được định danh bởi một khóa duy nhất. Nó cũng giống như khái niệm tên tệp trong một thư mục trên hệ thống tệp của bạn. Tên tệp trong một thư mục trên đĩa cứng của bạn phải là duy nhất. Mỗi đối tượng trong một thùng chỉ có một khóa. Tên của thùng và khóa được sử dụng cùng nhau để cung cấp định danh duy nhất cho mỗi đối tượng lưu trữ trong S3.

Mọi đối tượng trong S3 đều có thể truy cập bằng cách sử dụng một địa chỉ URL kết hợp địa chỉ URL của S3, tên thùng và khóa duy nhất. Nếu bạn lưu trữ một đối tượng với khóa my\_favorite\_video.mov bên trong một thùng tên là prabhakar, đối tượng đó có thể được truy cập thông qua địa chỉ URL http://prabhakar.s3.amazonaws.com/ my\_favorite\_video.mov.

Mặc dù các khái niệm là đơn giản, như trong hình 1, các thùng, các đối tượng, các khóa cùng nhau cung cấp tính mềm dẻo cho việc xây dựng các giải pháp lưu trữ dữ liệu của bạn. Bạn có thể sử dụng những viên gạch móng này để lưu trữ dữ liệu một cách đơn giản trên S3, hoặc sử dụng tính mềm dẻo của chúng để xây dựng kho lưu trữ và các ứng dụng phức hợp trên S3 để cung cấp các tính năng mới.

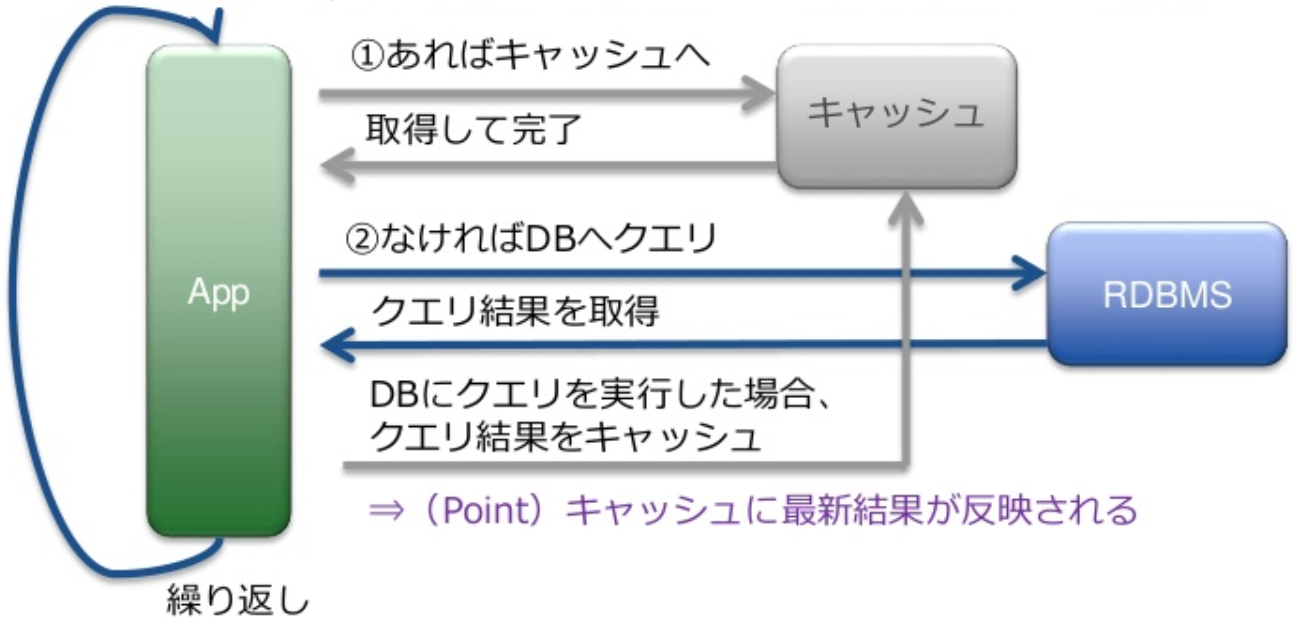
Hình 1. Khái niệm của S3

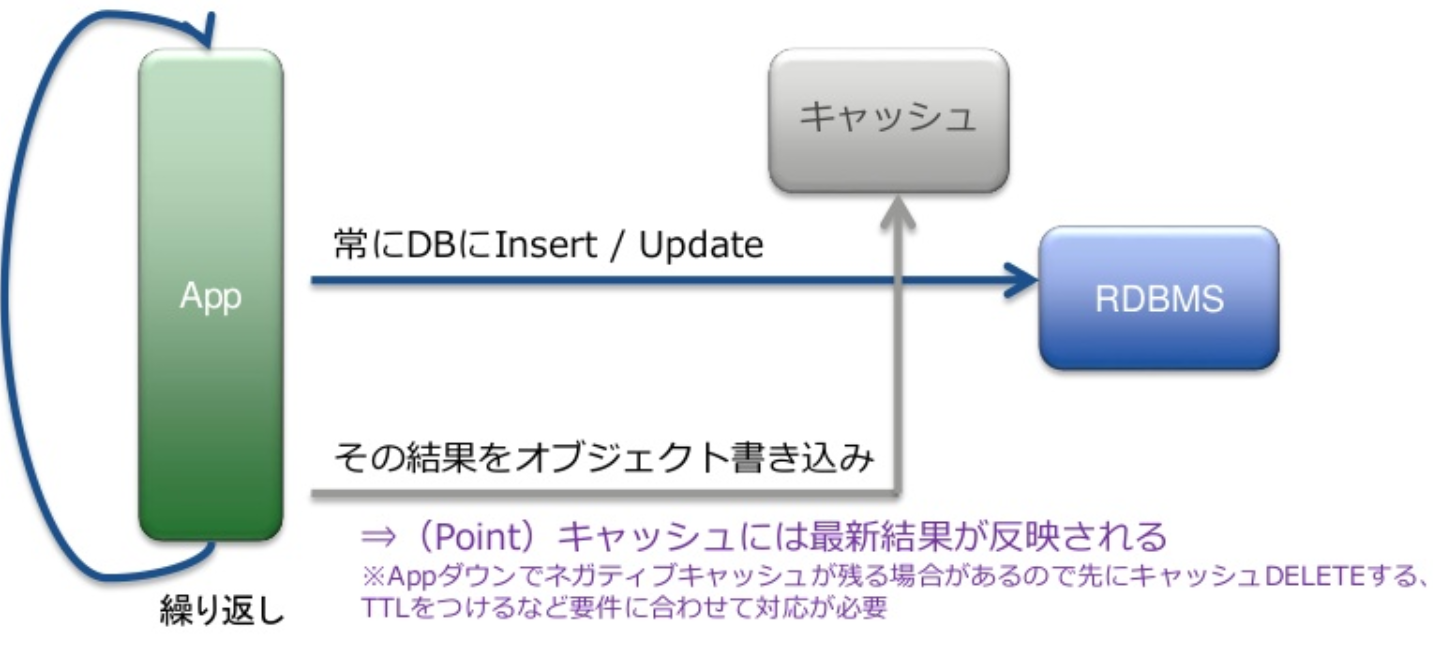
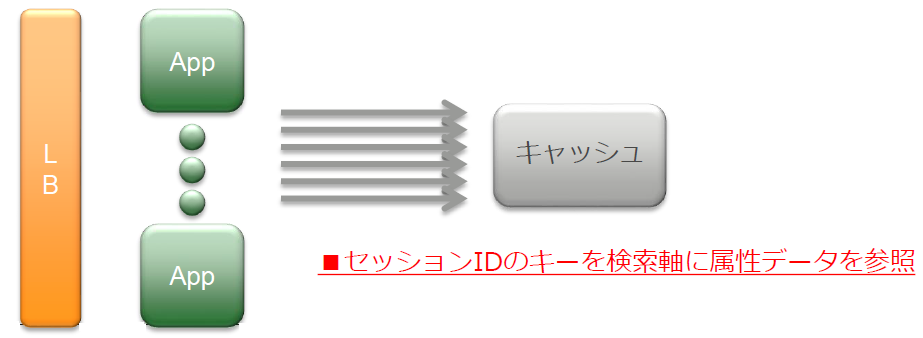


****

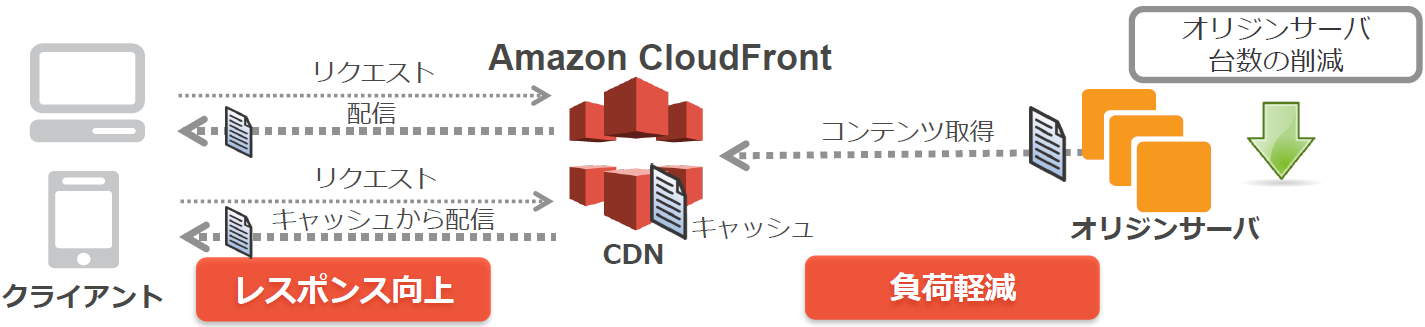
**Tham khảo:** [**http://www.ibm.com/developerworks/vn/library/ar-cloudaws2/**](http://www.ibm.com/developerworks/vn/library/ar-cloudaws2/)

### **2.5.ElasticCache**

****

**** ****

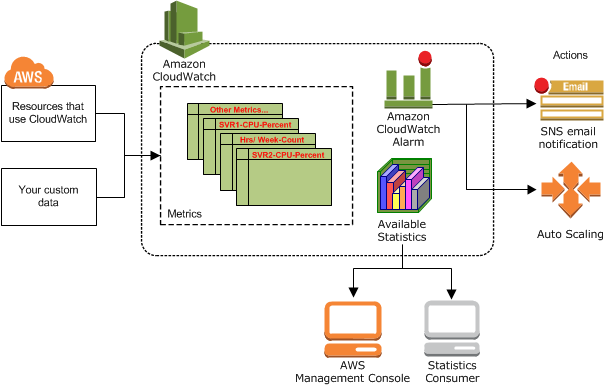
### **2.6.Cloud Front**

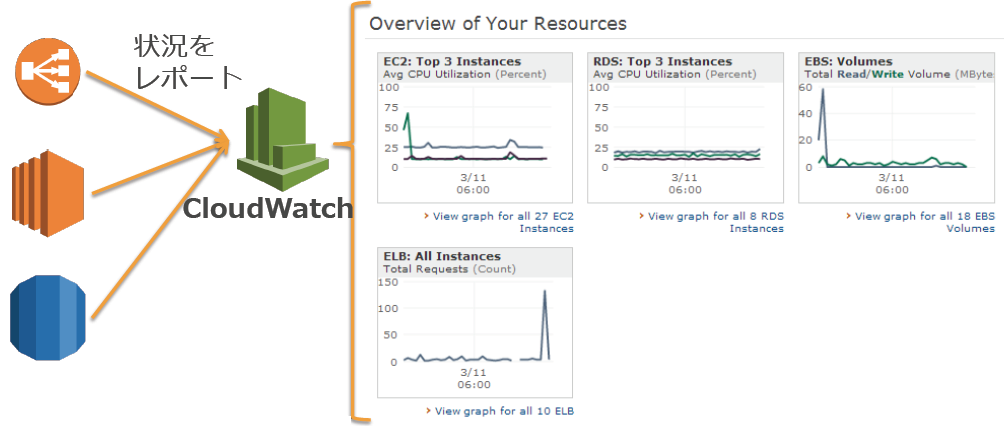
****

### **2.7.RDS**

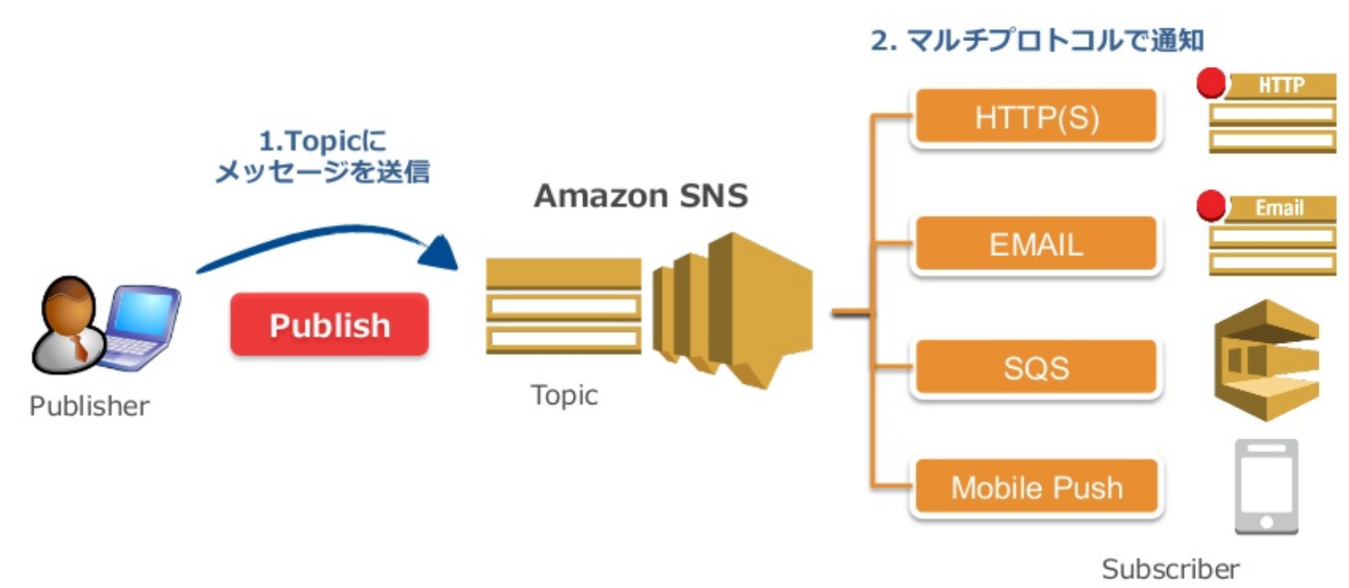
****

### **2.8.CloudWatch**

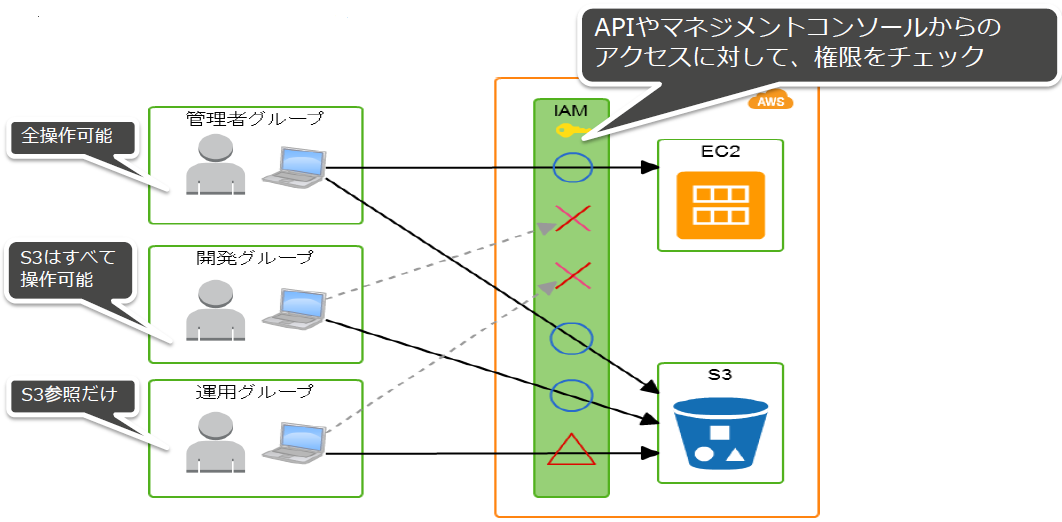




### **2.9.SNS**

****

### **2.10.IAM**

****

# **C.Ứng dụng thực tế**

## **1. Tạo ứng dụng Rails App trên môi trường Amazon Linux**

**B1:**Tạo 1 instance Amazon EC2

<http://azsysadmin.com/huong-dan-tao-vps-amazon-ec2/>

Bài viết dễ hiểu về AWS:

+ <http://dev.classmethod.jp/cloud/aws/aws-overview-for-non-engineers/>

Thao tác Console AWS va tao EC2 account

+ <https://www.youtube.com/watch?v=dBDJ_9ulSF0>

Sau khi tao duoc tai khoan ec2 thi

**B2:**Sử dụng putty để khởi động giao thức ssh

Tham khao:

+ <http://tech.recruit-mp.co.jp/infrastructure/retry-aws-minimum-vpc-server-environment/>

+ <http://qiita.com/Takumi_Kasuga/items/d55b8035708652a79547>

+ <https://namvuhn.wordpress.com/2015/08/04/aws-tim-hieu-ve-dich-vu-amazon-ec2/>

**B3:** Cài đặt Ruby và Rails

Thực hiện các bước cài đặt như trong step 2(Vagrant)

**B4:** Tạo Rails app và thực thi nó trên aws ec2

Mặc dù đã khai báo thêm cổng 3000 trong security group theo hướng dẫn:

<http://serverfault.com/questions/317903/aws-ec2-open-port-8080>

Config lỗi Javaruntime theo cách sau (không thực hiện theo cách đã hướng dẫn trong step2 được)

<https://tamosblog.wordpress.com/2013/12/20/linux-rbenb-ruby/>

Mục đích chính vẫn là đi sửa gemfile và sau đó thực thi lệnh “bundle install”

Tuy nhiên để truy cập từ brower vào webserver webrick của rails thì ta cần thực hiện các thao tác sau:

+ Thêm 1 cổng(port 3000) vào trong sercurity group

<http://serverfault.com/questions/317903/aws-ec2-open-port-8080>

+ Truy cập từ brower

Thực thi lệnh run server như sau:

rails server -b ec2-XX-XX-XX-XX.us-west-2.compute.amazonaws.com -p 3000

Và access tới:

Truy câp theo đường dẫn

http://ec2-XX-XX-XX-XX.us-west-2.compute.amazonaws.com:3000

Trong đó : ec2-XX-XX-XX-XX.us-west-2.compute.amazonaws.com chính là public DNS mà aws cung cấp

vd: http:// ec2-54-178-177-143.ap-northeast-1.compute.amazonaws.com:3000/

thì sẽ thực thi được

Tham khảo : <http://www.sakc.jp/blog/archives/32865>

29/09/2015:

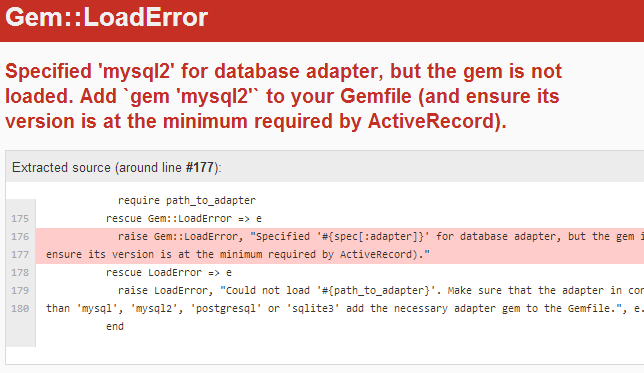
+ Học cách Deploy 1 sản phẩn Rails với AWS

Tham khảo: <https://www.youtube.com/watch?v=nHu8fCp9GR4>

>cd rails\_examples

>rails new demoror –d mysql # Khởi tạo rails project sử dụng db là mysql(thay vì mặc đinh là sqlite)

Mỗi lần truy cập vào rails app được khởi tạo với mysql đều gặp 1 lỗi là



Thì nguyên nhân là do mysql phiên bản 4.x trở lên sẽ không sử dụng được nên ta phải giới hạn phiên bản cho nó

Cách giải quyết:

+ <http://qiita.com/shizuma/items/0f9660d5d46a0012eb9e>

Ta sửa trong file Gemfile theo như hướng dẫn và save lại, sau đó thực thi “bundle install”

Tiếp theo ta gặp lỗi như sau:



Đây là lỗi phát sinh do mysql được đặt pass nhưng trong khai báo congfig/database.yml thì password được để trống. Vì vậy ta có thể vào sửa file database.yml và khai báo password của mysql. Trong trường hợp ko nhớ(hoặc ko biết mysql có pass là gì) thì ta sẽ đi reset lại pass bằng cách

<http://www.rackspace.com/knowledge_center/article/mysql-resetting-a-lost-mysql-root-password>

Tuy nhiên mình thì làm nhanh hơn bằng cách

<https://gist.github.com/kenzo0107/94509f453f2d49fc2500>

>service mysqld start

>mysql\_secure\_installation

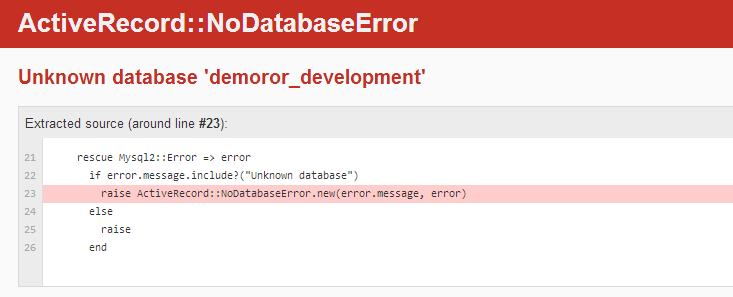
Nó sẽ hỏi có muốn reset ko? Nhập pass hiện tại(do lúc đầu mình ko cài pass nên mình enter luôn). Và nó yêu cầu nhập pass mới.Sau khi tạo xong pass mới thì mình test thử mysql

>mysql –u root –p password

Enter pass: <Nhập pass vừa đổi vào>

mysql>exit # Thoát

Tiếp theo mình build lại 1 lần nữa thì gặp lỗi này



Như vậy ta chỉ cần khởi tạo db và thực hiện

Bây giờ ta sẽ tạo 1 rails app demo và thực thi deploy lên aws ec2

Theo trên mạng hướng dẫn thì có 2 cách để deploy rails app lên aws , ta sẽ đi tìm hiểu từng cách trong đó.

* Không thể deploy cái nào cả, đkm senpai

**c.Network trong AWS**

d.LoadBracer trong AWS

Tham khảo:

<https://namvuhn.wordpress.com/2015/08/12/aws-tim-hieu-ve-ec2-elastic-load-balancing/>

<https://namvuhn.wordpress.com/2015/08/04/aws-tim-hieu-ve-dich-vu-amazon-ec2/>

<https://namvuhn.wordpress.com/2015/08/08/aws-tong-quan-ve-security-credentials-trong-aws/>

<http://bloghoctap.com/web-programming/xay-dung-he-thong-load-balancer-cho-web.html>

<http://bloghoctap.com/web-programming/cai-dat-web-server-cho-amazon-ec2-instance-video-clip.html>

<http://azsysadmin.com/huong-dan-tao-vps-amazon-ec2/>

<http://dinhnn.com/?s=aws>

<http://dinhnn.com/2014/09/21/aws-2-gioi-thi%CC%A3u-v-amazon-elastic-compute-cloud-amazon-ec2/>

Load Balancer là một reverse proxy phân chia request đến nhiều backend server nhằm mở rộng bang thông, giảm độ trễ của công tác xử lý, giảm tải cho mỗi backend server, đảm bảo fault-tolerance.

Khi đăng nhập vào tk aws ec2 cần chú ý về việc thiết lập security group. Nếu để mặc định ở dạng anywhere thì ở đâu bạn cũng có thể sử dụng, và nó sẽ dễ dàng bị tin tặc tấn công và sử dụng giúp bạn. Nên vì thế hãy để ở dạng myip hoặc customip. Và sau đó khi chỉnh sửa trong putty thì vần host name bạn nên để ở dạng: ec2-uer@public\_dns thay vì để ở dạng ec2-user@public\_ip như trước nữa, port vẫn sử dụng cổng 22 để sử dụng giao thức ssh đăng nhập vào hệ thống.

## **2.Hướng dẫn cài đặt Redmine trên EC2**

login as: ec2-user

Authenticating with public key "ec2-52-40-183-46.us-west-2.compute.amazonaws.com "

\_\_| \_\_|\_ )

\_| ( / Amazon Linux AMI

\_\_\_|\\_\_\_|\_\_\_|

https://aws.amazon.com/amazon-linux-ami/2016.03-release-notes/

9 package(s) needed for security, out of 17 available

Run "sudo yum update" to apply all updates.

[ec2-user@ip-172-31-27-156 ~]$

[ec2-user@ip-172-31-27-156 ~]$ pwd

/home/ec2-user

[ec2-user@ip-172-31-27-156 ~]$ sudo mkdir redmine\_setup

[ec2-user@ip-172-31-27-156 ~]$ cd redmine\_setup/

[ec2-user@ip-172-31-27-156 redmine\_setup]$ cd ..

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Kiểm tra version của OS

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ cat /etc/os-release

NAME="Amazon Linux AMI"

VERSION="2016.03"

ID="amzn"

ID\_LIKE="rhel fedora"

VERSION\_ID="2016.03"

PRETTY\_NAME="Amazon Linux AMI 2016.03"

ANSI\_COLOR="0;33"

CPE\_NAME="cpe:/o:amazon:linux:2016.03:ga"

HOME\_URL="http://aws.amazon.com/amazon-linux-ami/"

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Kiểm tra version của Ruby , gem

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ ruby -v

ruby 2.0.0p648 (2015-12-16) [x86\_64-linux]

[ec2-user@ip-172-31-27-156 ~]$ gem -v

2.0.14.1

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Update OS

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ sudo yum update

Installed:

kernel.x86\_64 0:4.4.10-22.54.amzn1

Updated:

aws-cfn-bootstrap.noarch 0:1.4-11.6.amzn1

ca-certificates.noarch 0:2015.2.6-65.0.1.15.amzn1

curl.x86\_64 0:7.40.0-8.57.amzn1

iputils.x86\_64 0:20121221-7.13.amzn1

kernel-tools.x86\_64 0:4.4.10-22.54.amzn1

libcurl.x86\_64 0:7.40.0-8.57.amzn1

nspr.x86\_64 0:4.11.0-1.37.amzn1

nss.x86\_64 0:3.21.0-9.76.amzn1

nss-softokn.x86\_64 0:3.16.2.3-14.2.38.amzn1

nss-softokn-freebl.x86\_64 0:3.16.2.3-14.2.38.amzn1

nss-sysinit.x86\_64 0:3.21.0-9.76.amzn1

nss-tools.x86\_64 0:3.21.0-9.76.amzn1

nss-util.x86\_64 0:3.21.0-2.2.50.amzn1

sudo.x86\_64 0:1.8.6p3-20.22.amzn1

tzdata.noarch 0:2016d-1.62.amzn1

tzdata-java.noarch 0:2016d-1.62.amzn1

Complete!

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Kiểm tra Apache đã được cài đặt sẵn trong OS hay chưa

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ httpd -v

-bash: httpd: command not found

↓

Như vậy là apache chưa được cài đặt trong hệ thống.

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Apacheのセットアップ

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ sudo yum install httpd

Installed:

httpd.x86\_64 0:2.2.31-1.7.amzn1

Dependency Installed:

apr.x86\_64 0:1.5.1-1.12.amzn1

apr-util.x86\_64 0:1.4.1-4.17.amzn1

apr-util-ldap.x86\_64 0:1.4.1-4.17.amzn1

httpd-tools.x86\_64 0:2.2.31-1.7.amzn1

Complete!

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Kiểm tra Version Apache và check hoạt động

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ httpd -v

Server version: Apache/2.2.31 (Unix)

Server built: Mar 7 2016 23:50:21

[ec2-user@ip-172-31-27-156 ~]$

[ec2-user@ip-172-31-27-156 ~]$ sudo service httpd status

httpd is stopped

[ec2-user@ip-172-31-27-156 ~]$ sudo service httpd start

Starting httpd: [ OK ]

[ec2-user@ip-172-31-27-156 ~]$ sudo chkconfig httpd on

[ec2-user@ip-172-31-27-156 ~]$

[ec2-user@ip-172-31-27-156 ~]$ sudo service httpd stop

Stopping httpd: [ OK ]

[ec2-user@ip-172-31-27-156 ~]$ sudo service httpd status

httpd is stopped

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt các gói quan trọng　I

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ sudo yum --enablerepo=epel install ruby-devel mysql-devel ImageMagick ImageMagick-devel

Installed:

ImageMagick.x86\_64 0:6.7.8.9-13.20.amzn1

ImageMagick-devel.x86\_64 0:6.7.8.9-13.20.amzn1

mysql-devel.noarch 0:5.5-1.6.amzn1

ruby-devel.noarch 1:2.0-0.3.amzn1

Dependency Installed:

avahi-libs.x86\_64 0:0.6.25-12.17.amzn1

bzip2-devel.x86\_64 0:1.0.6-8.12.amzn1

cairo.x86\_64 0:1.12.14-6.8.amzn1

cups-libs.x86\_64 1:1.4.2-67.21.amzn1

freetype-devel.x86\_64 0:2.3.11-15.14.amzn1

ghostscript.x86\_64 0:8.70-19.23.amzn1

ghostscript-devel.x86\_64 0:8.70-19.23.amzn1

ghostscript-fonts.noarch 0:5.50-23.2.7.amzn1

gnutls.x86\_64 0:2.8.5-19.15.amzn1

jasper-libs.x86\_64 0:1.900.1-16.9.amzn1

jbigkit-libs.x86\_64 0:2.0-11.4.amzn1

keyutils-libs-devel.x86\_64 0:1.5.8-3.12.amzn1

krb5-devel.x86\_64 0:1.13.2-12.40.amzn1

lcms2-devel.x86\_64 0:2.5-4.4.amzn1

libICE-devel.x86\_64 0:1.0.6-1.4.amzn1

libSM-devel.x86\_64 0:1.2.1-2.6.amzn1

libX11-devel.x86\_64 0:1.6.0-2.2.12.amzn1

libXau-devel.x86\_64 0:1.0.6-4.9.amzn1

libXdamage.x86\_64 0:1.1.3-4.7.amzn1

libXext-devel.x86\_64 0:1.3.2-2.1.10.amzn1

libXfixes.x86\_64 0:5.0.1-2.1.8.amzn1

libXt.x86\_64 0:1.1.4-6.1.9.amzn1

libXt-devel.x86\_64 0:1.1.4-6.1.9.amzn1

libXxf86vm.x86\_64 0:1.1.3-2.1.9.amzn1

libcom\_err-devel.x86\_64 0:1.42.12-4.40.amzn1

libdrm.x86\_64 0:2.4.52-4.12.amzn1

libgomp.x86\_64 0:4.8.3-9.109.amzn1

libjpeg-turbo-devel.x86\_64 0:1.2.90-5.14.amzn1

libpciaccess.x86\_64 0:0.13.1-4.1.11.amzn1

libselinux-devel.x86\_64 0:2.1.10-3.22.amzn1

libsepol-devel.x86\_64 0:2.1.7-3.12.amzn1

libtiff.x86\_64 0:4.0.3-20.20.amzn1

libtiff-devel.x86\_64 0:4.0.3-20.20.amzn1

libtool-ltdl.x86\_64 0:2.4.2-20.4.8.3.31.amzn1

libverto-devel.x86\_64 0:0.2.5-4.9.amzn1

libwmf-lite.x86\_64 0:0.2.8.4-41.11.amzn1

libxcb-devel.x86\_64 0:1.8.1-1.18.amzn1

mesa-dri-drivers.x86\_64 0:10.1.2-2.32.amzn1

mesa-dri-filesystem.x86\_64 0:10.1.2-2.32.amzn1

mesa-dri1-drivers.x86\_64 0:7.11-8.7.amzn1

mesa-libGL.x86\_64 0:10.1.2-2.32.amzn1

mysql-config.x86\_64 0:5.5.46-1.10.amzn1

mysql55.x86\_64 0:5.5.46-1.10.amzn1

mysql55-devel.x86\_64 0:5.5.46-1.10.amzn1

mysql55-libs.x86\_64 0:5.5.46-1.10.amzn1

openssl-devel.x86\_64 1:1.0.1k-14.91.amzn1

pixman.x86\_64 0:0.32.4-4.11.amzn1

ruby20-devel.x86\_64 0:2.0.0.648-1.29.amzn1

urw-fonts.noarch 0:2.4-10.7.amzn1

xorg-x11-proto-devel.noarch 0:7.7-9.10.amzn1

zlib-devel.x86\_64 0:1.2.8-7.18.amzn1

Complete!

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt các gói quan trọng II

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ sudo yum groupinstall "Development Tools"

Installed:

autoconf.noarch 0:2.69-11.9.amzn1 automake.noarch 0:1.13.4-3.15.amzn1 bison.x86\_64 0:2.7-4.11.amzn1 byacc.x86\_64 0:1.9.20070509-7.8.amzn1 crash.x86\_64 0:7.1.2-1.17.amzn1

cscope.x86\_64 0:15.6-6.5.amzn1 ctags.x86\_64 0:5.8-2.6.amzn1 cvs.x86\_64 0:1.11.23-16.9.amzn1 diffstat.x86\_64 0:1.51-2.5.amzn1 doxygen.x86\_64 1:1.8.2-1.11.amzn1

elfutils.x86\_64 0:0.163-3.18.amzn1 flex.x86\_64 0:2.5.37-3.13.amzn1 gcc.noarch 0:4.8.3-3.20.amzn1 gcc-c++.noarch 0:4.8.3-3.20.amzn1 gcc-gfortran.noarch 0:4.8.3-3.20.amzn1

gdb.x86\_64 0:7.6.1-64.33.amzn1 gettext.x86\_64 0:0.18.1.1-9.1.11.amzn1 git.x86\_64 0:2.7.4-1.47.amzn1 indent.x86\_64 0:2.2.10-7.6.amzn1 intltool.noarch 0:0.41.0-1.1.5.amzn1

kexec-tools.x86\_64 0:2.0.0-273.11.amzn1 latrace.x86\_64 0:0.5.9-2.2.amzn1 libtool.x86\_64 0:2.4.2-20.4.8.3.31.amzn1 ltrace.x86\_64 0:0.5-23.45svn.6.amzn1 patch.x86\_64 0:2.7.1-8.9.amzn1

patchutils.x86\_64 0:0.3.1-3.1.5.amzn1 rcs.x86\_64 0:5.7-37.7.amzn1 rpm-build.x86\_64 0:4.11.2-2.73.amzn1 strace.x86\_64 0:4.8-11.16.amzn1 subversion.x86\_64 0:1.8.15-1.54.amzn1

swig.x86\_64 0:2.0.10-4.24.amzn1 system-rpm-config.noarch 0:9.0.3-42.27.amzn1 systemtap.x86\_64 0:3.0-2.33.amzn1 systemtap-runtime.x86\_64 0:3.0-2.33.amzn1 texinfo.x86\_64 0:5.1-4.10.amzn1

valgrind.x86\_64 1:3.10.0-16.14.amzn1

Dependency Installed:

busybox.x86\_64 1:1.19.3-2.11.amzn1 cpp48.x86\_64 0:4.8.3-9.109.amzn1 dyninst.x86\_64 0:8.1.2-6.3.amzn1 elfutils-libs.x86\_64 0:0.163-3.18.amzn1 gcc48.x86\_64 0:4.8.3-9.109.amzn1

gcc48-c++.x86\_64 0:4.8.3-9.109.amzn1 gcc48-gfortran.x86\_64 0:4.8.3-9.109.amzn1 gettext-common-devel.noarch 0:0.18.1.1-9.1.11.amzn1 gettext-devel.x86\_64 0:0.18.1.1-9.1.11.amzn1 gettext-libs.x86\_64 0:0.18.1.1-9.1.11.amzn1

glibc-devel.x86\_64 0:2.17-106.167.amzn1 glibc-headers.x86\_64 0:2.17-106.167.amzn1 hardlink.x86\_64 1:1.0-10.4.amzn1 json-c.x86\_64 0:0.11-6.8.amzn1 kernel-devel.x86\_64 0:4.4.10-22.54.amzn1

kernel-headers.x86\_64 0:4.4.10-22.54.amzn1 libdwarf.x86\_64 0:20130207-3.3.amzn1 libmpc.x86\_64 0:1.0.1-3.3.amzn1 libquadmath.x86\_64 0:4.8.3-9.109.amzn1 libquadmath48-devel.x86\_64 0:4.8.3-9.109.amzn1

libserf.x86\_64 0:1.3.7-1.7.amzn1 libstdc++48-devel.x86\_64 0:4.8.3-9.109.amzn1

libunistring.x86\_64 0:0.9.3-6.1.amzn1 lzo.x86\_64 0:2.08-1.5.amzn1

m4.x86\_64 0:1.4.16-9.10.amzn1

mpfr.x86\_64 0:3.1.1-4.14.amzn1

perl-Data-Dumper.x86\_64 0:2.145-3.5.amzn1

perl-Error.noarch 1:0.17020-2.9.amzn1

perl-Git.noarch 0:2.7.4-1.47.amzn1

perl-TermReadKey.x86\_64 0:2.30-20.9.amzn1

perl-Test-Harness.noarch 0:3.28-2.5.amzn1

perl-Text-Unidecode.noarch 0:0.04-20.7.amzn1

perl-Thread-Queue.noarch 0:3.02-2.5.amzn1

perl-XML-Parser.x86\_64 0:2.41-10.9.amzn1

perl-libintl.x86\_64 0:1.20-12.7.amzn1

snappy.x86\_64 0:1.0.5-1.1.amzn1

subversion-libs.x86\_64 0:1.8.15-1.54.amzn1

systemtap-client.x86\_64 0:3.0-2.33.amzn1

systemtap-devel.x86\_64 0:3.0-2.33.amzn1

Complete!

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt các gói quan trọng III

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ sudo yum install openssl-devel readline-devel zlib-devel curl-devel libyaml-devel

5/5

Installed:

libcurl-devel.x86\_64 0:7.40.0-8.57.amzn1 libyaml-devel.x86\_64 0:0.1.6-6.7.amzn1 readline-devel.x86\_64 0:6.2-9.14.amzn1

Dependency Installed:

automake19.noarch 0:1.9.6-3.12.amzn1 ncurses-devel.x86\_64 0:5.7-3.20090208.13.amzn1

Complete!

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt font tiếng Nhật cho hệ thống

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ sudo yum install -y vlgothic-p-fonts

Installed:

vlgothic-p-fonts.noarch 0:20091202-2.2.amzn1

Dependency Installed:

vlgothic-fonts-common.noarch 0:20091202-2.2.amzn1

Complete!

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt MySQL

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ sudo yum -y install mysql-server mysql-devel

Installed:

mysql-server.noarch 0:5.5-1.6.amzn1

Dependency Installed:

mysql55-server.x86\_64 0:5.5.46-1.10.amzn1

perl-Compress-Raw-Bzip2.x86\_64 0:2.061-3.11.amzn1

perl-Compress-Raw-Zlib.x86\_64 1:2.061-4.1.amzn1

perl-DBD-MySQL55.x86\_64 0:4.023-5.23.amzn1

perl-DBI.x86\_64 0:1.627-4.8.amzn1

perl-IO-Compress.noarch 0:2.061-2.12.amzn1

perl-Net-Daemon.noarch 0:0.48-5.5.amzn1

perl-PlRPC.noarch 0:0.2020-14.7.amzn1

Complete!

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Config default character cho Mysql

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 ~]$ sudo vi /etc/my.cnf

Ctr V -> ấn phím C để chuyển mode insert

[mysqld]

datadir=/var/lib/mysql

socket=/var/lib/mysql/mysql.sock

symbolic-links=0

character-set-server=utf8

[mysqld\_safe]

log-error=/var/log/mysqld.log

pid-file=/var/run/mysqld/mysqld.pid

[mysql]

default-character-set=utf8

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ MySQL初期設定ツールの実行

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

Sau khi config mysql thì ta sẽ khởi động mysql và thiết lập ban đầu:

[ec2-user@ip-172-31-27-156 ~]$ sudo service mysqld status

mysqld is stopped

[ec2-user@ip-172-31-27-156 ~]$ sudo service mysqld start

Initializing MySQL database: Installing MySQL system tables...

160528 5:02:55 [Note] /usr/libexec/mysql55/mysqld (mysqld 5.5.46) starting as process 23094 ...

OK

Filling help tables...

160528 5:02:55 [Note] /usr/libexec/mysql55/mysqld (mysqld 5.5.46) starting as process 23101 ...

OK

To start mysqld at boot time you have to copy

support-files/mysql.server to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !

To do so, start the server, then issue the following commands:

/usr/libexec/mysql55/mysqladmin -u root password 'new-password'

/usr/libexec/mysql55/mysqladmin -u root -h ip-172-31-27-156 password 'new-password'

Alternatively you can run:

/usr/libexec/mysql55/mysql\_secure\_installation

which will also give you the option of removing the test

databases and anonymous user created by default. This is

strongly recommended for production servers.

See the manual for more instructions.

You can start the MySQL daemon with:

cd /usr ; /usr/libexec/mysql55/mysqld\_safe &

You can test the MySQL daemon with mysql-test-run.pl

cd /usr/mysql-test ; perl mysql-test-run.pl

Please report any problems at http://bugs.mysql.com/

[ OK ]

Starting mysqld: [ OK ]

[ec2-user@ip-172-31-27-156 ~]$

[ec2-user@ip-172-31-27-156 ~]$ sudo mysql\_secure\_installation

Change the root password? [Y/n] Y

New password: \*\*\*\*\*\*\*\* [Nhập pass vào]

Re-enter new password: \*\*\*\*\*\*\*\* [Nhập pass vào]

Password updated successfully!

Reloading privilege tables..

... Success!

Remove anonymous users? [Y/n] Y

... Success!

Disallow root login remotely? [Y/n] Y

... Success!

Remove test database and access to it? [Y/n] Y

... Success!

Reload privilege tables now? [Y/n] Y

... Success!

Thanks for using MySQL!

[ec2-user@ip-172-31-27-156 ~]$ sudo mysql -uroot -p

Enter password: \*\*\*\*\*\*\*\* [Nhập pass vào]

mysql>

Tiếp theo:

ENTER

1. mysql> CREATE DATABASE db\_redmine default character set utf8;

Query OK, 1 row affected (0.00 sec)

2. mysql> CREATE USER 'user\_redmine'@'localhost' IDENTIFIED BY 'kens6566';

Query OK, 0 rows affected (0.00 sec)

3. mysql> GRANT ALL PRIVILEGES ON db\_redmine.\* TO 'user\_redmine'@'localhost';

Query OK, 0 rows affected (0.00 sec)

4. mysql> FLUSH PRIVILEGES;

Query OK, 0 rows affected (0.00 sec)

5. mysql> quit;

Bye

[ec2-user@ip-172-31-27-156 ~]$

↑

Giải thích ý nghĩa những dòng lệnh trên:

1. Tạo cơ sở dữ liệu db\_redmine

2. Tạo đối tượng người dùng là user\_redmine với pass là: kens6566

3. Thiết lập tất cả các quyền thao tác với db db\_redmine cho đối tượng user\_redmine

4. Đảm bảo các quyền được thiết lập có hiệu lực ngay.

5. Thoát khỏi mysql

Kiểm tra lại việc thiết lập người dùng và database

[ec2-user@ip-172-31-47-94 ~]$ mysql -u root -p　[Đăng nhập với tk root]

Enter password: \*\*\*\*\*\*\*\* [Nhập pass vào]

mysql> show databases;

+--------------------+

| Database |

+--------------------+

| information\_schema |

| db\_redmine |

| mysql |

| performance\_schema |

+--------------------+

4 rows in set (0.00 sec)

mysql> quit

Bye

[ec2-user@ip-172-31-47-94 ~]$ mysql -u user\_redmine -p [Đăng nhập với người dùng là user\_redmine]

Enter password: \*\*\*\*\*\*\*\* [Nhập pass vào]

mysql> show databases;

+--------------------+

| Database |

+--------------------+

| information\_schema |

| db\_redmine |

+--------------------+

2 rows in set (0.01 sec)

mysql> use db\_redmine;

Database changed

mysql> create table test (

id int(10) auto\_increment primary key,

first\_name varchar(50) default null ,

last\_name varchar(50) default null,

email varchar(50) null,

register\_time timestamp

);

Query OK, 0 rows affected (0.01 sec)

mysql> show tables;

+----------------------+

| Tables\_in\_db\_redmine |

+----------------------+

| test |

+----------------------+

1 row in set (0.00 sec)

mysql> insert into test value(1 , 'chu quang' , 'vien' , 'vienbk91@gmail.com' , NOW() );

Query OK, 1 row affected (0.00 sec)

mysql> select \* test;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'test' at line 1

mysql> select \* from test;

+----+------------+-----------+--------------------+--------------------+

| id | first\_name | last\_name | email | register\_time |

+----+------------+-----------+--------------------+--------------------+

| 1 | chu quang | vien | vienbk91@gmail.com | 2016-05-28 05:33:25 |

+----+------------+-----------+--------------------+--------------------+

1 row in set (0.00 sec)

mysql> quit;

Bye

[ec2-user@ip-172-31-27-156 ~]$

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Dowload gói Redmine 2.5.2

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

Tạo 1 folder tên redmine, và tất cả các gói redmine down về sẽ chứa trong đây

[ec2-user@ip-172-31-27-156 ~]$ pwd

/home/ec2-user

[ec2-user@ip-172-31-27-156 ~]$ sudo mkdir redmine

[ec2-user@ip-172-31-27-156 ~]$ cd redmine/

[ec2-user@ip-172-31-27-156 redmine]

[ec2-user@ip-172-31-27-156 redmine]$ sudo curl -O http://www.redmine.org/releases/redmine-2.5.2.tar.gz

[ec2-user@ip-172-31-27-156 redmine]$ ls -a -l

[ec2-user@ip-172-31-27-156 redmine]$ sudo tar xvf redmine-2.5.2.tar.gz

[ec2-user@ip-172-31-27-156 redmine]$ sudo mv redmine-2.5.2 /var/lib/redmine #[Thực thi lệnh đổi tên và chuyển thư mục mới sang 1 folder mới theo đường dẫn thiết lập]

[ec2-user@ip-172-31-27-156 redmine]$ cd /var/lib/redmine/

ec2-user@ip-172-31-27-156 redmine]$ pwd

/var/lib/redmine

Tìm hiểu về lệnh nén và giải nén trong linux:

++ [ http://blog.hostvn.net/server/vps/vi-du-co-ban-ve-lenh-tar-trong-linux.html ]

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Config database và file configuration

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[ec2-user@ip-172-31-27-156 redmine]$ cd config

[ec2-user@ip-172-31-27-156 config]$ ls -a -l

Bởi vì không có file database.yml và configuration.yml nên ta sẽ coppy từ 2 file có sẵn

trong thư mục config là: configuration.yml.example và database.yml.example

+ Sửa file database.yml

production:

adapter: mysql2

database: db\_redmine

host: localhost

username: user\_redmine

password: "kens6566"

encoding: utf8

+ Sửa file configuration.yml

Thêm vào phần production

production:

email\_delivery:

delivery\_method: :smtp

smtp\_settings:

address: 'localhost'

port: 25

domain: 'ec2-52-39-208-22.us-west-2.compute.amazonaws.com'

rmagick\_font\_path: /usr/share/fonts/vlgothic/VL-PGothic-Regular.ttf

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt gói bundler with ec2-user (失敗しました。でも、念のため、メモを記録する)

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

Trước hết hãy thử làm việc với user: ec2-user

[ec2-user@ip-172-31-27-156 redmine]$ gem install bundler --no-rdoc --no-ri

[ec2-user@ip-172-31-27-156 redmine]$ bundle install --without development test

/usr/share/ruby/vendor\_ruby/2.0/rubygems/core\_ext/kernel\_require.rb:55:in `require': cannot load such file -- io/console (LoadError)

[ec2-user@ip-172-31-27-156 redmine]$ gem install io-console

[ec2-user@ip-172-31-27-156 redmine]$ bundle install --without development test

There was an error while trying to write to `/var/lib/redmine/.bundle/config`. It is likely that you need to grant write permissions for that path.

[ec2-user@ip-172-31-27-156 redmine]$ ls -a -l

total 96

drwxrwxr-x 16 1000 1000 4096 Mar 29 2014 .

drwxr-xr-x 20 root root 4096 May 31 01:46 ..

drwxrwxr-x 7 1000 1000 4096 Mar 29 2014 app

drwxrwxr-x 5 1000 1000 4096 May 31 02:03 config

-rw-rw-r-- 1 1000 1000 160 Mar 29 2014 config.ru

-rw-rw-r-- 1 1000 1000 240 Mar 29 2014 CONTRIBUTING.md

drwxrwxr-x 3 1000 1000 4096 Mar 29 2014 db

drwxrwxr-x 2 1000 1000 4096 Mar 29 2014 doc

drwxrwxr-x 5 1000 1000 4096 Mar 29 2014 extra

drwxrwxr-x 2 1000 1000 4096 Mar 29 2014 files

-rw-rw-r-- 1 1000 1000 3505 Mar 29 2014 Gemfile

-rw-rw-r-- 1 1000 1000 731 Mar 29 2014 .gitignore

-rw-rw-r-- 1 1000 1000 607 Mar 29 2014 .hgignore

drwxrwxr-x 8 1000 1000 4096 Mar 29 2014 lib

drwxrwxr-x 2 1000 1000 4096 Mar 29 2014 log

drwxrwxr-x 2 1000 1000 4096 Mar 29 2014 plugins

drwxrwxr-x 7 1000 1000 4096 Mar 29 2014 public

-rw-rw-r-- 1 1000 1000 275 Mar 29 2014 Rakefile

-rw-rw-r-- 1 1000 1000 205 Mar 29 2014 README.rdoc

drwxrwxr-x 2 1000 1000 4096 Mar 29 2014 script

drwxrwxr-x 9 1000 1000 4096 Mar 29 2014 test

drwxrwxr-x 8 1000 1000 4096 Mar 29 2014 tmp

-rw-rw-r-- 1 1000 1000 1326 Mar 29 2014 .travis.yml

drwxrwxr-x 2 1000 1000 4096 Mar 29 2014 vendor

↓

Kiểm tra nếu không có .bundle thì ta cần chuyển qua mod root để thực thi,

còn nếu không thì ta sử dụng lệnh chmod để đổi quyền của thư mục.

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt gói bundler with root

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

Có 1 chú ý là không nên thực thi tất cả các hoạt động với bundler gem ở user: ec2-user vì rất hay có lỗi.

Cho nên từ giờ ta sẽ hoạt động ở mod root

[ec2-user@ip-172-31-27-156 redmine]$ sudo su -

Last login: Mon May 30 11:31:16 UTC 2016 on pts/1

[root@ip-172-31-27-156 ~]# pwd

/root

[root@ip-172-31-27-156 ~]# cd /var/lib/redmine/

[root@ip-172-31-27-156 redmine]# gem install bundler --no-rdoc --no-ri

Successfully installed bundler-1.12.5

1 gem installed

[root@ip-172-31-27-156 redmine]# gem install io-console

Successfully installed io-console-0.4.6

1 gem installed

[root@ip-172-31-27-156 redmine]# bundle install --without development test

Don't run Bundler as root. Bundler can ask for sudo if it is needed, and installing your bundle as root will break

Bundle complete! 22 Gemfile dependencies, 40 gems now installed.

Gems in the groups development and test were not installed.

Use `bundle show [gemname]` to see where a bundled gem is installed.

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Khởi tạo khóa bảo mật

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# bundle exec rake generate\_secret\_token

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Khởi tạo cấu trúc database mặc định cho Redmine

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# bundle exec rake db:migrate RAILS\_ENV=production --trace

[root@ip-172-31-27-156 redmine]#

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Điền dữ liệu mẫu vào database

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# bundle exec rake redmine:load\_default\_data RAILS\_ENV=production

Select language: ar, az, bg, bs, ca, cs, da, de, el, en, en-GB, es, et, eu, fa, fi, fr, gl, he, hr, hu, id, it, ja, ko, lt, lv, mk, mn, nl, no, pl, pt, pt-BR, ro, ru, sk, sl, sq, sr, sr-YU, sv, th, tr, uk, vi, zh, zh-TW [en] ja

====================================

Default configuration data loaded.

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt gói Passenger

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# sudo gem install passenger --no-rdoc --no-ri

Successfully installed passenger-5.0.28

1 gem installed

[root@ip-172-31-27-156 redmine]#

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt Passenger-Apache2-Module (注意、全部を読んだ後実行)

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# passenger-install-apache2-module

Welcome to the Phusion Passenger Apache 2 module installer, v5.0.28.

Use <space> to select.

If the menu doesn't display correctly, press '!'

> (\*) Ruby

(\*) Python

( ) Node.js

( ) Meteor

--------------------------------------------

Checking for required software...

\* Checking for C compiler...

Found: yes

Location: /usr/bin/cc

\* Checking for C++ compiler...

Found: yes

Location: /usr/bin/c++

\* Checking for Curl development headers with SSL support...

Found: yes

curl-config location: /usr/bin/curl-config

Header location: /usr/include/curl/curl.h

Version: libcurl 7.40.0

Usable: yes

Supports SSL: yes

\* Checking for Zlib development headers...

Found: yes

Location: /usr/include/zlib.h

\* Checking for Apache 2...

Found: yes

Location of httpd: /usr/sbin/httpd

Apache version: 2.2.31

\* Checking for Apache 2 development headers...

Found: no

\* Checking for Rake (associated with /usr/bin/ruby2.0)...

Found: yes

Location: /usr/bin/ruby2.0 /usr/local/bin/rake

\* Checking for OpenSSL support for Ruby...

Found: yes

\* Checking for RubyGems...

Found: yes

\* Checking for Ruby development headers...

Found: yes

Location: /usr/include/ruby/2.0/ruby.h

\* Checking for rack...

Found: yes

\* Checking for Apache Portable Runtime (APR) development headers...

Found: no

\* Checking for Apache Portable Runtime Utility (APU) development headers...

Found: no

Some required software is not installed.

But don't worry, this installer will tell you how to install them.

Press Enter to continue, or Ctrl-C to abort.

^C

[root@ip-172-31-27-156 redmine]#

↓

Cài đặt thất bại do thiếu các gói thư viện httpd-devel.

Như vậy bước cài đặt httpd-devel cần được làm trước

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt gói httpd-devel

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# sudo yum install httpd-devel

Installed:

httpd-devel.x86\_64 0:2.2.31-1.7.amzn1

Dependency Installed:

apr-devel.x86\_64 0:1.5.1-1.12.amzn1

apr-util-devel.x86\_64 0:1.4.1-4.17.amzn1

cyrus-sasl-devel.x86\_64 0:2.1.23-13.16.amzn1

db4-cxx.x86\_64 0:4.7.25-18.11.amzn1

db4-devel.x86\_64 0:4.7.25-18.11.amzn1

expat-devel.x86\_64 0:2.1.0-8.18.amzn1

openldap-devel.x86\_64 0:2.4.40-7.28.amzn1

Complete!

[root@ip-172-31-27-156 redmine]#

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Cài đặt Passenger-Apache2-Module （再実行　－mod root）

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# passenger-install-apache2-module

Use <space> to select.

If the menu doesn't display correctly, press '!'

> (\*) Ruby

(\*) Python

( ) Node.js

( ) Meteor

--------------------------------------------

Checking for required software...

\* Checking for C compiler...

Found: yes

Location: /usr/bin/cc

\* Checking for C++ compiler...

Found: yes

Location: /usr/bin/c++

\* Checking for Curl development headers with SSL support...

Found: yes

curl-config location: /usr/bin/curl-config

Header location: /usr/include/curl/curl.h

Version: libcurl 7.40.0

Usable: yes

Supports SSL: yes

\* Checking for Zlib development headers...

Found: yes

Location: /usr/include/zlib.h

\* Checking for Apache 2...

Found: yes

Location of httpd: /usr/sbin/httpd

Apache version: 2.2.31

\* Checking for Apache 2 development headers...

Found: yes

Location of apxs2: /usr/sbin/apxs

\* Checking for Rake (associated with /usr/bin/ruby2.0)...

Found: yes

Location: /usr/bin/ruby2.0 /usr/local/bin/rake

\* Checking for OpenSSL support for Ruby...

Found: yes

\* Checking for RubyGems...

Found: yes

\* Checking for Ruby development headers...

Found: yes

Location: /usr/include/ruby/2.0/ruby.h

\* Checking for rack...

Found: yes

\* Checking for Apache Portable Runtime (APR) development headers...

Found: yes

Location: /usr/bin/apr-1-config

Version: 1.5.1

\* Checking for Apache Portable Runtime Utility (APU) development headers...

Found: yes

Location: /usr/bin/apu-1-config

Version: 1.4.1

--------------------------------------------

Checking whether there are multiple Apache installations...

Only a single installation detected. This is good.

--------------------------------------------

Your system does not have a lot of virtual memory

Compiling Phusion Passenger works best when you have at least 1024 MB of virtual

memory. However your system only has 995 MB of total virtual memory (995 MB

RAM, 0 MB swap). It is recommended that you temporarily add more swap space

before proceeding. You can do it as follows:

sudo dd if=/dev/zero of=/swap bs=1M count=1024

sudo mkswap /swap

sudo swapon /swap

See also https://wiki.archlinux.org/index.php/Swap for more information about

the swap file on Linux.

If you cannot activate a swap file (e.g. because you're on OpenVZ, or if you

don't have root privileges) then you should install Phusion Passenger through

DEB/RPM packages. For more information, please refer to our installation

documentation:

https://www.phusionpassenger.com/library/install/apache/

Press Ctrl-C to abort this installer (recommended).

Press Enter if you want to continue with installation anyway.

--------------------------------------------

Compiling and installing Apache 2 module...

--------------------------------------------

Almost there!

Please edit your Apache configuration file, and add these lines:

LoadModule passenger\_module /usr/local/share/ruby/gems/2.0/gems/passenger-5.0.28/buildout/apache2/mod\_passenger.so

<IfModule mod\_passenger.c>

PassengerRoot /usr/local/share/ruby/gems/2.0/gems/passenger-5.0.28

PassengerDefaultRuby /usr/bin/ruby2.0

</IfModule>

After you restart Apache, you are ready to deploy any number of web

applications on Apache, with a minimum amount of configuration!

Press ENTER when you are done editing.

--------------------------------------------

Validating installation...

\* Checking whether this Passenger install is in PATH... ✓

\* Checking whether there are no other Passenger installations... ✓

\* Checking whether Apache is installed... ✓

\* Checking whether the Passenger module is correctly configured in Apache... (!)

You did not specify 'LoadModule passenger\_module' in any of your Apache onfiguration files. Please paste the configuration snippet that this nstaller printed earlier, into one of your Apache configuration files, such as /etc/httpd/conf/httpd.conf.

Detected 0 error(s), 1 warning(s).

Press ENTER to continue.

--------------------------------------------

Deploying a web application

To learn how to deploy a web app on Passenger, please follow the deployment guide:

https://www.phusionpassenger.com/library/deploy/apache/deploy/

Enjoy Phusion Passenger, a product of Phusion (www.phusion.nl) :-)

https://www.phusionpassenger.com

Phusion Passenger is a registered trademark of Hongli Lai & Ninh Bui.

[root@ip-172-31-27-156 redmine]#

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Tạo file config passenger.conf

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# passenger-install-apache2-module --snippet

LoadModule passenger\_module /usr/local/share/ruby/gems/2.0/gems/passenger-5.0.28/buildout/apache2/mod\_passenger.so

<IfModule mod\_passenger.c>

PassengerRoot /usr/local/share/ruby/gems/2.0/gems/passenger-5.0.28

PassengerDefaultRuby /usr/bin/ruby2.0

</IfModule>

[root@ip-172-31-27-156 redmine]# sudo vi /etc/httpd/conf.d/passenger.conf

#\*\*\*\*\*\*\*\*\*\*\*Create file with content \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

LoadModule passenger\_module /usr/local/share/ruby/gems/2.0/gems/passenger-5.0.28/buildout/apache2/mod\_passenger.so

<IfModule mod\_passenger.c>

PassengerRoot /usr/local/share/ruby/gems/2.0/gems/passenger-5.0.28

PassengerDefaultRuby /usr/bin/ruby2.0

</IfModule>

# rails settings [Cần thiết để định nghĩa đường dẫn tới thư mục redmine]

RailsBaseURI /redmine

Header always unset "X-Powered-By"

Header always unset "X-Rack-Cache"

Header always unset "X-Content-Digest"

Header always unset "X-Runtime"

PassengerMaxPoolSize 20

PassengerMaxInstancesPerApp 4

PassengerPoolIdleTime 3600

PassengerHighPerformance on

PassengerStatThrottleRate 10

PassengerSpawnMethod smart

RailsAppSpawnerIdleTime 86400

PassengerMaxPreloaderIdleTime 0

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Sau khi tạo config passenger.conf thì thực thi lại 1 lần nữa

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# passenger-install-apache2-module

Use <space> to select.

If the menu doesn't display correctly, press '!'

> (\*) Ruby

(\*) Python

( ) Node.js

( ) Meteor

--------------------------------------------

Validating installation...

\* Checking whether this Passenger install is in PATH... ✓

\* Checking whether there are no other Passenger installations... ✓

\* Checking whether Apache is installed... ✓

\* Checking whether the Passenger module is correctly configured in Apache... ✓

--------------------------------------------

Everything looks good. :-)

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Phân quyền cho uesr apache vào thư mục Redmine

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# sudo chown -R apache:apache /var/lib/redmine

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Tạo symbol link từ thư mục redmine đến document root của apache

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# sudo ln -s /var/lib/redmine/public /var/www/html/redmine

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Chỉnh sửa lại document root của apache

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 redmine]# cd /etc/httpd/conf/

[root@ip-172-31-27-156 conf]# sudo vi httpd.conf

#\*\*\*\*\*\*\*\*\*\*\*\*\*Document Root add content \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

DocumentRoot "/var/www/html"

<Location /redmine>

Options -Indexes

RailsBaseURI /redmine

</Location>

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[root@ip-172-31-27-156 conf]#

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Khởi động lại apache

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 ec2-user]# sudo service httpd status

httpd is stopped

[root@ip-172-31-27-156 ec2-user]# sudo service httpd start

Starting httpd: [ OK ]

[root@ip-172-31-27-156 ec2-user]# sudo chkconfig httpd on

[root@ip-172-31-27-156 ec2-user]# sudo service httpd configtest

Syntax OK

[root@ip-172-31-27-156 ec2-user]# sudo service httpd graceful

[root@ip-172-31-27-156 ec2-user]#

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Tạo thư mục plugin\_assets trong folder publics và phân quyền cho nó

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 ec2-user]# cd /var/lib/redmine/public/

[root@ip-172-31-27-156 public]# pwd

/var/lib/redmine/public

[root@ip-172-31-27-156 public]#

[root@ip-172-31-27-156 public]# sudo mkdir plugin\_assets

[root@ip-172-31-27-156 public]# sudo chown -R apache:apache /var/lib/redmine/public/plugin\_assets

[root@ip-172-31-27-156 public]# sudo service httpd restart

Stopping httpd: [ OK ]

Starting httpd: [ OK ]

[root@ip-172-31-27-156 public]# sudo chkconfig httpd on

[root@ip-172-31-27-156 public]#

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Dowload plugins redmine\_backlogs

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@ip-172-31-27-156 public]# cd /home/ec2-user/redmine\_setup/

[root@ip-172-31-27-156 redmine\_setup]# pwd

/home/ec2-user/redmine\_setup

[root@ip-172-31-27-156 redmine\_setup]#

[root@ip-172-31-27-156 redmine\_setup]# ls -a -l

total 3820

drwxr-xr-x 3 root root 4096 May 31 06:32 .

drwx------ 5 ec2-user ec2-user 4096 May 31 06:37 ..

drwxrwxr-x 16 1000 1000 4096 Jul 6 2014 redmine-2.5.2

-rw-r--r-- 1 root root 3897300 May 31 06:32 redmine-2.5.2.tar.gz

[root@ip-172-31-27-156 redmine\_setup]# cd ..

[root@ip-172-31-27-156 ec2-user]# gem install holidays --version 1.0.3

Fetching: holidays-1.0.3.gem (100%)

Successfully installed holidays-1.0.3

Parsing documentation for holidays-1.0.3

Installing ri documentation for holidays-1.0.3

Done installing documentation for holidays after 1 seconds

1 gem installed

[root@ip-172-31-27-156 ec2-user]# gem install holidays

Fetching: holidays-4.1.0.gem (100%)

Successfully installed holidays-4.1.0

Parsing documentation for holidays-4.1.0

Installing ri documentation for holidays-4.1.0

Done installing documentation for holidays after 2 seconds

1 gem installed

[root@ip-172-31-38-217 redmine]# sudo yum install -y libxml2 libxml2-devel libxslt libxslt-devel

Installed:

libxml2-devel.x86\_64 0:2.9.1-6.2.50.amzn1 libxslt-devel.x86\_64 0:1.1.28-5.12.amzn1

Dependency Installed:

libgcrypt-devel.x86\_64 0:1.5.3-12.18.amzn1 libgpg-error-devel.x86\_64 0:1.11-1.12.amzn1

xz-devel.x86\_64 0:5.1.2-12alpha.12.amzn1

Complete!

[root@ip-172-31-27-156 ec2-user]# sudo service httpd stop

Stopping httpd: [ OK ]

[root@ip-172-31-27-156 ec2-user]#

[root@ip-172-31-27-156 ec2-user]#

[root@ip-172-31-27-156 ec2-user]# cd redmine\_setup/

[root@ip-172-31-27-156 redmine\_setup]# sudo git clone git://github.com/backlogs/redmine\_backlogs.git

Cloning into 'redmine\_backlogs'...

remote: Counting objects: 22226, done.

remote: Total 22226 (delta 0), reused 0 (delta 0), pack-reused 22225

Receiving objects: 100% (22226/22226), 11.07 MiB | 2.27 MiB/s, done.

Resolving deltas: 100% (13963/13963), done.

Checking connectivity... done.

[root@ip-172-31-27-156 redmine\_setup]# ls -a -l

total 3824

drwxr-xr-x 4 root root 4096 May 31 07:35 .

drwx------ 5 ec2-user ec2-user 4096 May 31 06:37 ..

drwxrwxr-x 16 1000 1000 4096 Jul 6 2014 redmine-2.5.2

-rw-r--r-- 1 root root 3897300 May 31 06:32 redmine-2.5.2.tar.gz

drwxr-xr-x 11 root root 4096 May 31 07:35 redmine\_backlogs

[root@ip-172-31-27-156 redmine\_setup]#

[root@ip-172-31-27-156 redmine\_setup]#

[root@ip-172-31-27-156 redmine\_setup]# sudo mv redmine\_backlogs /var/lib/redmine/plugins

[root@ip-172-31-27-156 redmine\_setup]# sudo chown -R apache:apache /var/lib/redmine/plugins/redmine\_backlogs

[root@ip-172-31-27-156 redmine\_setup]# cd /var/lib/redmine/plugins/redmine\_backlogs

[root@ip-172-31-27-156 redmine\_backlogs]# git tag

[中断]...

v1.0.6

[root@ip-172-31-27-156 redmine\_backlogs]#

[root@ip-172-31-27-156 redmine\_backlogs]# sudo git checkout v1.0.6

Note: checking out 'v1.0.6'.

You are in 'detached HEAD' state. You can look around, make experimental

changes and commit them, and you can discard any commits you make in this

state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may

do so (now or later) by using -b with the checkout command again. Example:

git checkout -b <new-branch-name>

HEAD is now at e60fbdb... v1.0.6

[root@ip-172-31-27-156 redmine\_backlogs]#

[root@ip-172-31-38-217 redmine]# gem install nokogiri -v '1.5.11'

Building native extensions. This could take a while...

Successfully installed nokogiri-1.5.11

Parsing documentation for nokogiri-1.5.11

Installing ri documentation for nokogiri-1.5.11

Done installing documentation for nokogiri after 3 seconds

1 gem installed

[root@ip-172-31-38-217 redmine]# bundle update nokogiri

[root@ip-172-31-27-156 redmine\_backlogs]# RAILS\_ENV=production

[root@ip-172-31-27-156 redmine\_backlogs]# export RAILS\_ENV

[root@ip-172-31-27-156 redmine\_backlogs]#

[root@ip-172-31-27-156 redmine\_backlogs]# cd /var/lib/redmine

[root@ip-172-31-27-156 redmine]# bundle install --without development test

Don't run Bundler as root. Bundler can ask for sudo if it is needed, and installing your bundle as root will break this application for all non-root users on this machine.

Bundle complete! 45 Gemfile dependencies, 50 gems now installed.

Gems in the groups development and test were not installed.

Use `bundle show [gemname]` to see where a bundled gem is installed.

[root@ip-172-31-27-156 redmine]# bundle exec rake db:migrate RAILS\_ENV=production

[root@ip-172-31-27-156 redmine]# bundle exec rake redmine:plugins:migrate RAILS\_ENV=production

[root@ip-172-31-27-156 redmine]#

[root@ip-172-31-27-156 redmine]# sudo service httpd restart

Stopping httpd: [ OK ]

Starting httpd: [ OK ]

[root@ip-172-31-27-156 redmine]# sudo chkconfig httpd on

[root@ip-172-31-27-156 redmine]#

[root@ip-172-31-27-156 redmine]# bundle exec rake tmp:sessions:clear RAILS\_ENV=production

[root@ip-172-31-27-156 redmine]#

[root@ip-172-31-27-156 redmine]# sudo service httpd restart

Stopping httpd: [ OK ]

Starting httpd: [ OK ]

[root@ip-172-31-27-156 redmine]# sudo chkconfig httpd on

[root@ip-172-31-27-156 redmine]#

[root@ip-172-31-38-217 redmine]# bundle exec rake db:migrate RAILS\_ENV=production

[root@ip-172-31-38-217 redmine]# bundle exe rake tmp:cache:clear

[root@ip-172-31-38-217 redmine]# bundle exe rake tmp:sessions:clear

[root@ip-172-31-38-217 redmine]# bundle exec rake redmine:backlogs:install RAILS\_ENV=production

2.5.2.stable. You are running backlogs v1.0.6, latest version is 1.0.6

=====================================================

Redmine Backlogs Installer

=====================================================

Installing to the production environment.

Fetching card labels from http://git.gnome.org...done!

Story and task trackers are now set.

Migrating the database...\*\* Invoke redmine:plugins:migrate (first\_time)

\*\* Invoke environment (first\_time)

\*\* Execute environment

\*\* Execute redmine:plugins:migrate

\*\* Invoke db:schema:dump (first\_time)

\*\* Invoke environment

\*\* Invoke db:load\_config (first\_time)

\*\* Execute db:load\_config

\*\* Execute db:schema:dump

\*\* Invoke redmine:backlogs:fix\_positions (first\_time)

\*\* Invoke environment (first\_time)

\*\* Execute environment

\*\* Execute redmine:backlogs:fix\_positions

done!

Installation complete. Please restart Redmine.

Thank you for trying out Redmine Backlogs!

[root@ip-172-31-38-217 redmine]# bundle exec rake redmine:plugins:migrate RAILS\_ENV=production

Migrating redmine\_backlogs (Redmine Backlogs)...

[root@ip-172-31-38-217 redmine]#

Environment:

Redmine version 2.5.2.stable

Ruby version 2.0.0-p648 (2015-12-16) [x86\_64-linux]

Rails version 3.2.19

Environment production

Database adapter Mysql2

SCM:

Subversion 1.8.15

Git 2.7.4

Filesystem

Redmine plugins:

redmine\_backlogs v1.0.6

Phụ lục:

2016/06/07: Trong quá trình nghiên cứu và phát triển trên nền tảng AWS thì có 1 số câu hỏi phát sinh.

1. Hiện tại mình đang sử dụng AWS EC2 với AMI free. Nếu bây giờ mình muốn mở rộng thêm storage của EC2 thì có mất phí hay không ? Nếu mất thì sẽ mất bao nhiêu ?
2. Mở rộng storage của EC2 thì nên sử dụng EBS hay S3 ? Sự khác nhau giữa EBS và S3 là gì ?
3. Nếu dùng EBS mở rộng EC2 thì cách mở rộng như thế nào ? Backup data thì nên sử dụng EBS hay S3

Trả lời:

TK1: <http://stackoverflow.com/questions/29575877/aws-efs-vs-ebs-vs-s3-differences-when-to-use>

EBS, S3 là 1 trong những dịch vụ lưu trữ được AWS cung cấp.Ngoài ra còn có EFS , Glacier cũng có những chức năng lưu trữ tương tự. Tuy nhiên trong những TH cụ thể thì việc lựa chọn sử dụng EBS hay S3 cũng khác nhau, và vấn đề quan trọng nhất của việc sử dụng những dịch vụ này là dùng cái gì sao cho phù hợp với mục đích của bạn và rẻ nhất (AWS cung cấp các dịch vụ tính phí theo thời gian và dung lượng sử dụng).

Trước tiên ta sẽ tổng hợp những dịch vụ được AWS cung cấp cùng với giá cả của nó:

1GB lưu trữ: (dữ liệu cập nhật ngày 17/09/2015)

+ Glacier: 0,007$/month

+ S3: 0,03$/month

+ EBS: 0,05$~0,1$/month (phụ thuộc vào tốc độ và SSD hay không + chi phí IOPS)

+ EFS: $0,3$/month

Các chi phí trên chỉ là mẫu. Có thể có sự khác biệt theo vùng, và nó có thể thay đổi bất cứ lúc nào. Cũng có những chi phí thêm cho việc chuyển giao dữ liệu (ra internet). Tuy nhiên họ cho thấy một tỷ lệ giữa giá cả của các dịch vụ.

Tiếp theo thì ta sẽ phân tích tác dụng của các dịch vụ:

EBS: Nó là 1 khối lưu trữ mà bạn cần định dạng, vì thế mà bạn có thể chọn loại hệ thống tập tin mà bạn muốn. Vì nó là 1 khối lưu trữ nên bạn có thể dụng Raid 1(hoặc 0 hoặc 10) với nhiều khối lưu trữ. Nó nhanh và tương đối rẻ. Bạn có thể lưu trữ đến 16Tb dữ liệu trên mỗi SSD-s. Bạn có thể

<https://www.quora.com/What-is-the-difference-between-S3-and-Elastic-Block-Store-services-What-are-the-correct-use-cases-for-each-one>

<http://www.cloudiquity.com/2009/03/differences-between-s3-and-ebs/>