



AWS Essentials  
Student Guide  
Version 2.5

Copyright © 2013 Amazon Web Services, Inc. and its affiliates.  
All rights reserved.

This work may not be reproduced or redistributed, in whole or in part,  
without prior written permission from Amazon Web Services, Inc.

Commercial copying, lending, or selling is prohibited.

Corrections or feedback on the course? Email [aws-course-feedback@amazon.com](mailto:aws-course-feedback@amazon.com).

Other questions? Email us at [aws-training-info@amazon.com](mailto:aws-training-info@amazon.com).

# AWS Essentials Student Guide

Welcome to AWS Essentials! This guide will walk you through the training including the hands-on lab portions of this course. If you have any questions, please don't hesitate to ask your instructor for assistance.

## TABLE OF CONTENTS

<u>AWS Essentials Student Guide .....</u>	<u>4</u>
AWS History .....	7
Cloud Computing Paradigm .....	12
Elastic Capacity .....	18
Security .....	25
Global Infrastructure .....	41
AWS Services .....	46
Compute Services .....	66
Programmable Infrastructure .....	89
Managed Services .....	95
Resources .....	99
<u>Appendix .....</u>	<u>106</u>
qwikLAB Guide .....	106
S3 Exercise .....	107
Elastic Load Balancing Exercise .....	114



# AWS Essentials



Instructor Name

Version 2.3

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## **AWS Essentials**

This training introduces AWS products and services with exercises and hands-on activities. It helps learners who do not come with a background on Amazon Web Services to gain proficiency in AWS services and empowers them to make informed decisions about IT solutions based on business requirements.

## Course Overview



- AWS History
- Cloud Computing
- Elastic Capacity
- Security and IAM
- Global Infrastructure
- AWS Services
- Additional Resources

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **Course Overview**

This is the high level agenda for our 1-day training.

## Learning Objectives



By the end of this course, you will be able to:

- Describe key AWS Compute and Storage products and services, including EC2, Elastic Block Store, Auto Scaling, CloudFormation templates, VPC, and Amazon Glacier.
- Describe key AWS Security approaches.
- Create an S3 bucket and manage objects associated with that bucket.
- Demonstrate how to launch and manage server instances with AWS cloud.
- Identify available AWS online tools and resources.

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Course Learning Objectives

High level learning objectives for this 1-day training.

# AWS History



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## **Background on AWS**

A little history on Amazon Web Services and Amazon as a company.

## AWS History



- Amazon and AWS
- AWS 3 Major families of services
  - Storage
  - Compute
  - Database
- History of AWS

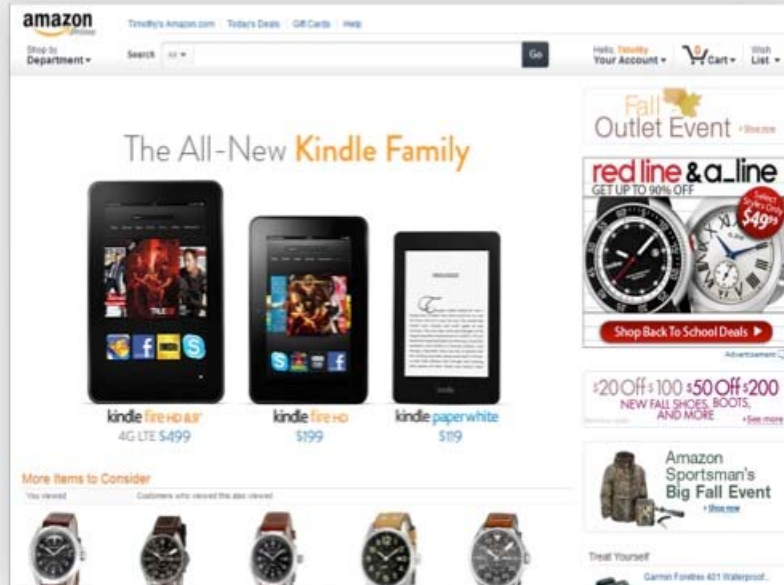
© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS History**

This is the high level agenda for this section.



## About Amazon



### About Amazon

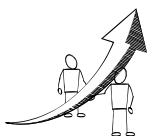
Amazon.com, Inc. is an American multinational electronic commerce company with headquarters in Seattle, Washington, United States. It is the world's largest online retailer. Amazon.com started as an online bookstore, but soon diversified, selling DVDs, CDs, MP3 downloads, software, video games, electronics, apparel, furniture, food, toys, and jewelry. The company also produces consumer electronics—notably the Amazon Kindle e-book reader and the Kindle Fire tablet computer—and is a major provider of cloud computing services.



## Amazon Web Services

Amazon Web Services offers a complete set of infrastructure and application services that enable you to run virtually everything in the cloud: from enterprise applications and big data projects to social games and mobile apps.

One of the key benefits of cloud computing is the opportunity to replace up-front capital infrastructure expenses with low variable costs that scale with your business.



### More Information

Learn more about Amazon Web Services (AWS):

<http://aws.amazon.com>

## History of AWS

- Over 10 Years in the Making
- Enablement of Sellers on Amazon
- Internal Need for Scalable Deployment Environment
- Early Forays Proved Developers Were Hungry For More

### AWS Mission

Enable businesses and developers to use web services\* to build scalable, sophisticated applications.

\*What people now call "the cloud"

### History of AWS

AWS Mission: Enable businesses and developers to use web services\* to build scalable, sophisticated applications.

\*What people now call "the cloud"

# Cloud Computing Paradigms & Utility Model for IaaS



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.



**Cloud Computing:** is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation.

**Infrastructure as a Service (IaaS):** also known as cloud computing is the most basic cloud-service model. Providers of IaaS offer computers - physical or (more often) virtual machines - and other resources. IaaS clouds often offer additional resources such as a virtual-machine disk image library, raw (block) and file-based storage, firewalls, load balancers, IP addresses, virtual local area networks (VLANs), and software bundles. IaaS-cloud providers like AWS supply these resources on-demand from their large pools installed in data centers.

## AWS Cloud Services Paradigm

- Infrastructure as a Service (not SaaS, PaaS)
- Fundamental building blocks that the customer assembles to meet their requirements
- Managed Services that shift the burden of undifferentiated heavy lifting to experts

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS Cloud Services Paradigm**

AWS is a collection of Infrastructure Services. It is the customer's responsibility to combine individual IaaS building blocks to meet their business requirements.

## AWS Utility Model



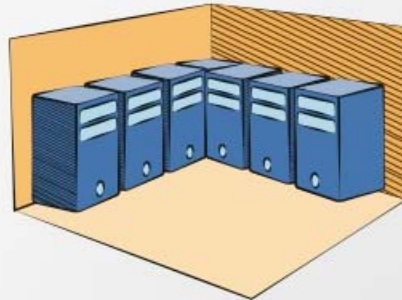
© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS Utility Model**

What are the benefits of utility based services?

## AWS Utility Model

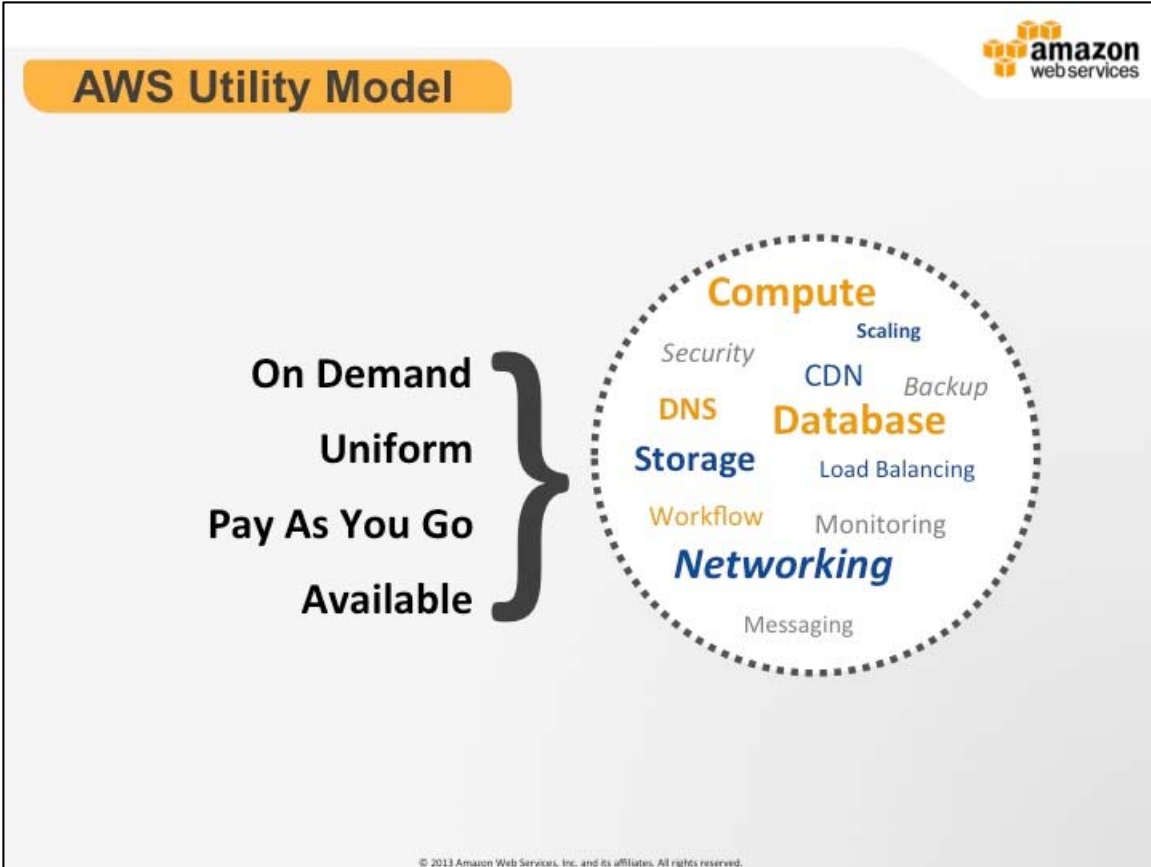
On Demand  
Uniform  
Pay As You Go  
Available



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS Utility Model**

Economy of scale and how to leverage AWS expertise.



**AWS Utility Model**

Since 2006, Amazon Web Services have been providing on demand, pay-as-you-go infrastructure to businesses of all sizes.



**On-Premise vs. Cloud Computing**

**On-Premise**  
Up-Front On-Premise Costs

- Physical Space
- Cabling
- Power
- Cooling
- Networking
- Racks
- Servers
- Storage
- Certification
- Labor

**VS**

**Cloud Computing**  
Variable Cloud Computing Costs

**\$0**  
to Get Started  
*no long-term contracts*

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **On-Premise vs. Cloud Computing**

Some benefits of cloud computing are zero capital expenditure and no long-term contracts.

Instead of paying for and organizing all of the physical requirements of an on-premise data center you can launch instances and resources in an on-demand nature.

# Elastic Capacity

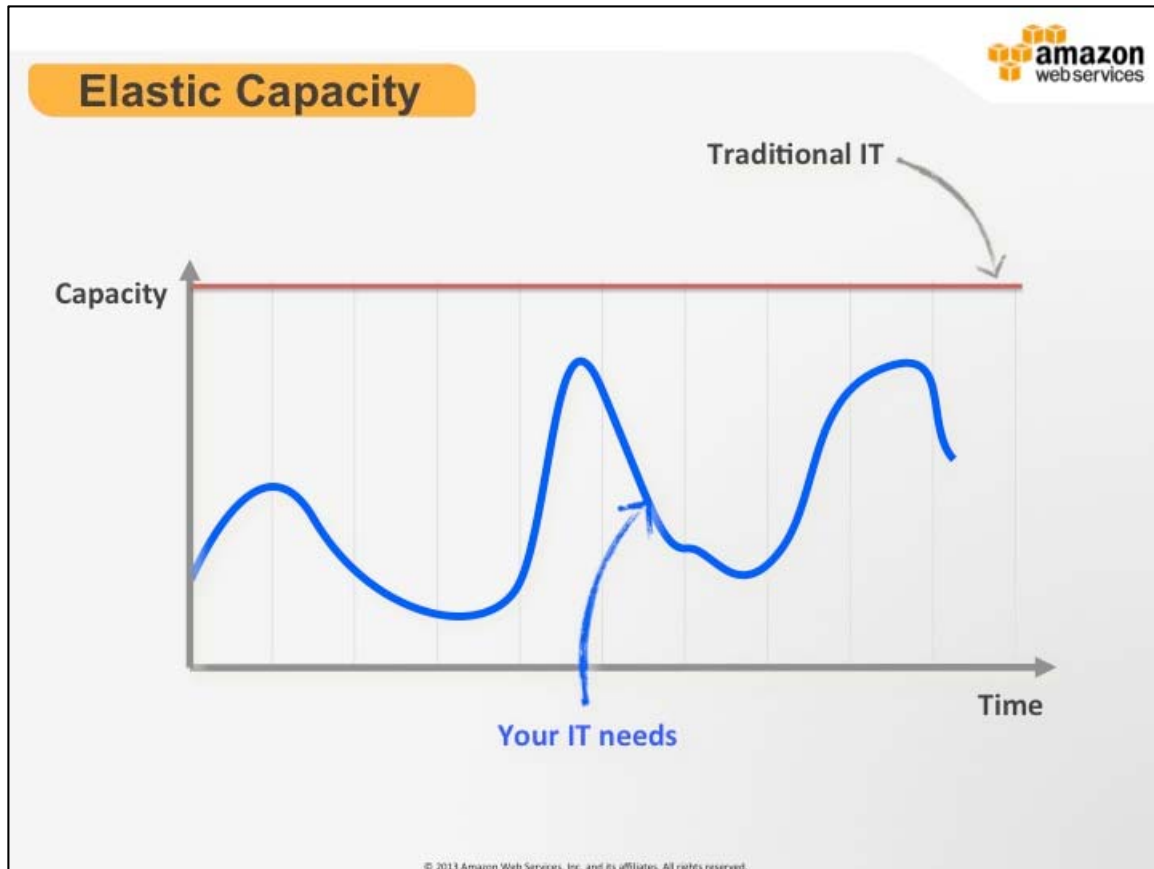


© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## **Elastic Capacity**

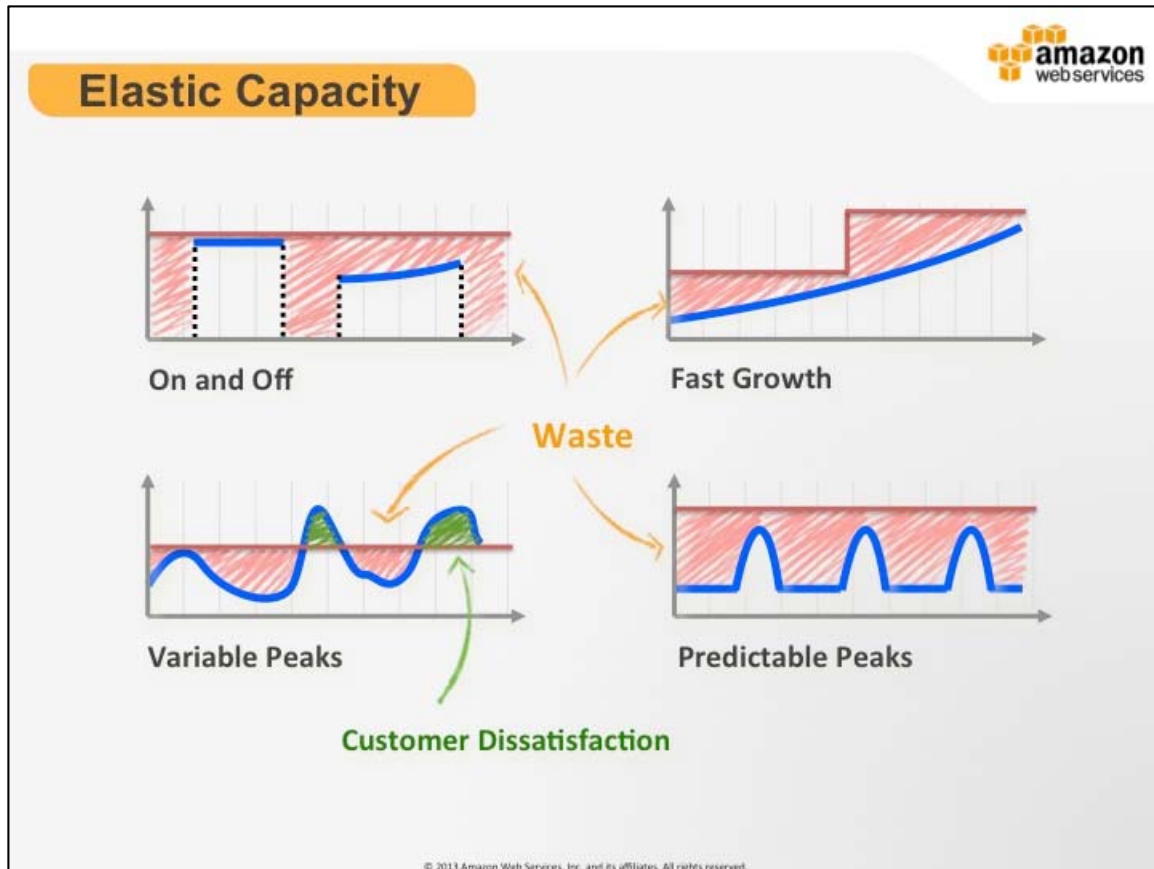
One of the paradigms of Cloud computing is Elastic capacity.

Virtual environments afford the ability to operate in an elastic way. The nature of Amazon's cloud, having on-demand, uniform and accessible components, allows you dynamically scale your computing resources to meet your business needs. The major difference between this and traditional IT virtualization is the scale and speed at which AWS Operates.



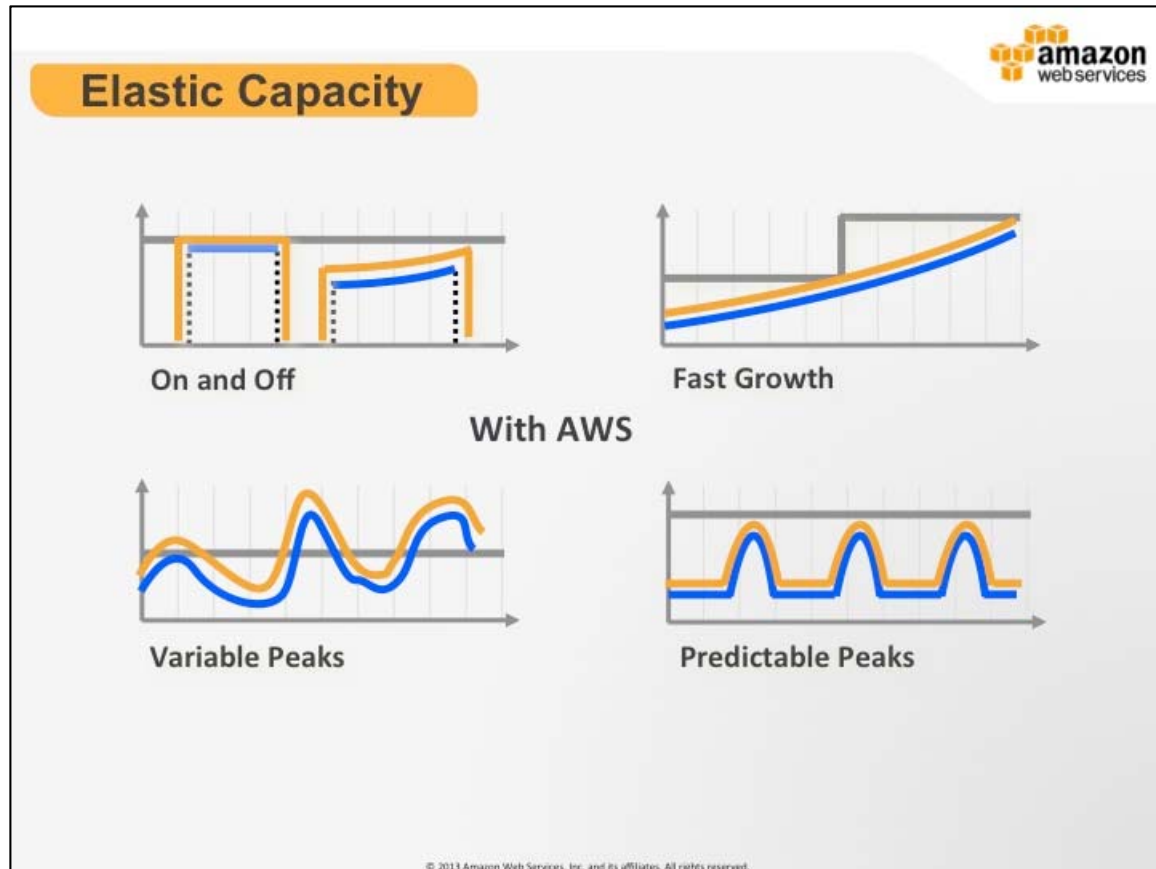
### **Elastic Capacity**

In Traditional IT you do capacity planning to predict IT spending. Customer needs are variable and hard to predict. End result is that IT departments end up provisioning hardware resources with very little accuracy and large lead times in changing levels of resources.



## Elastic Capacity

1. [On and Off] Batch Processing (transcoding, genomic research, simulations)
2. [Fast Growth] New Product Launch
3. [Variable Peaks] Social networking site with peaks at lunch and in the evening
4. [Predictable Peaks] Backup Jobs, Ticket Sales Website



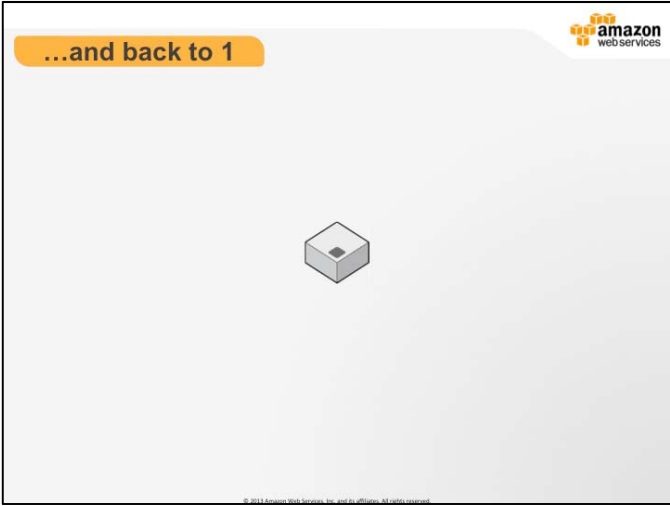
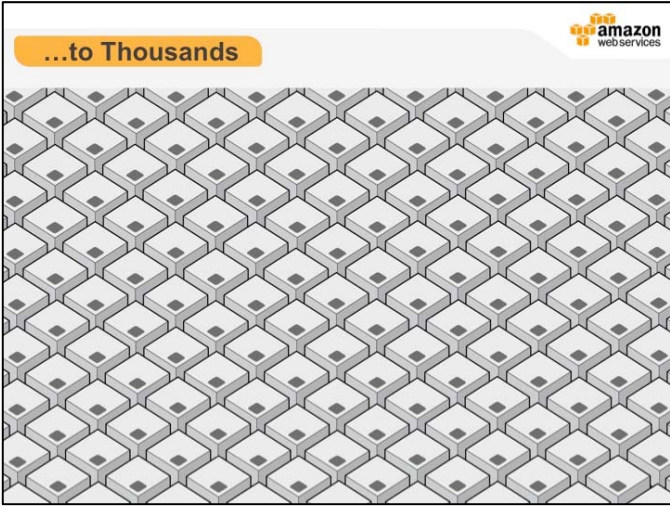
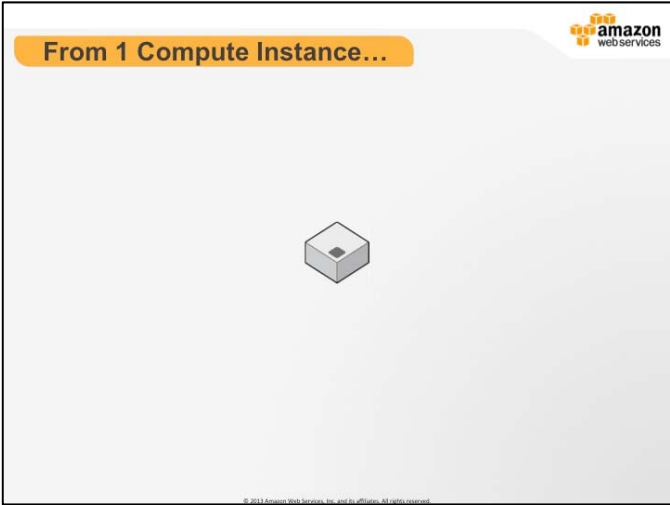
## Elastic Capacity

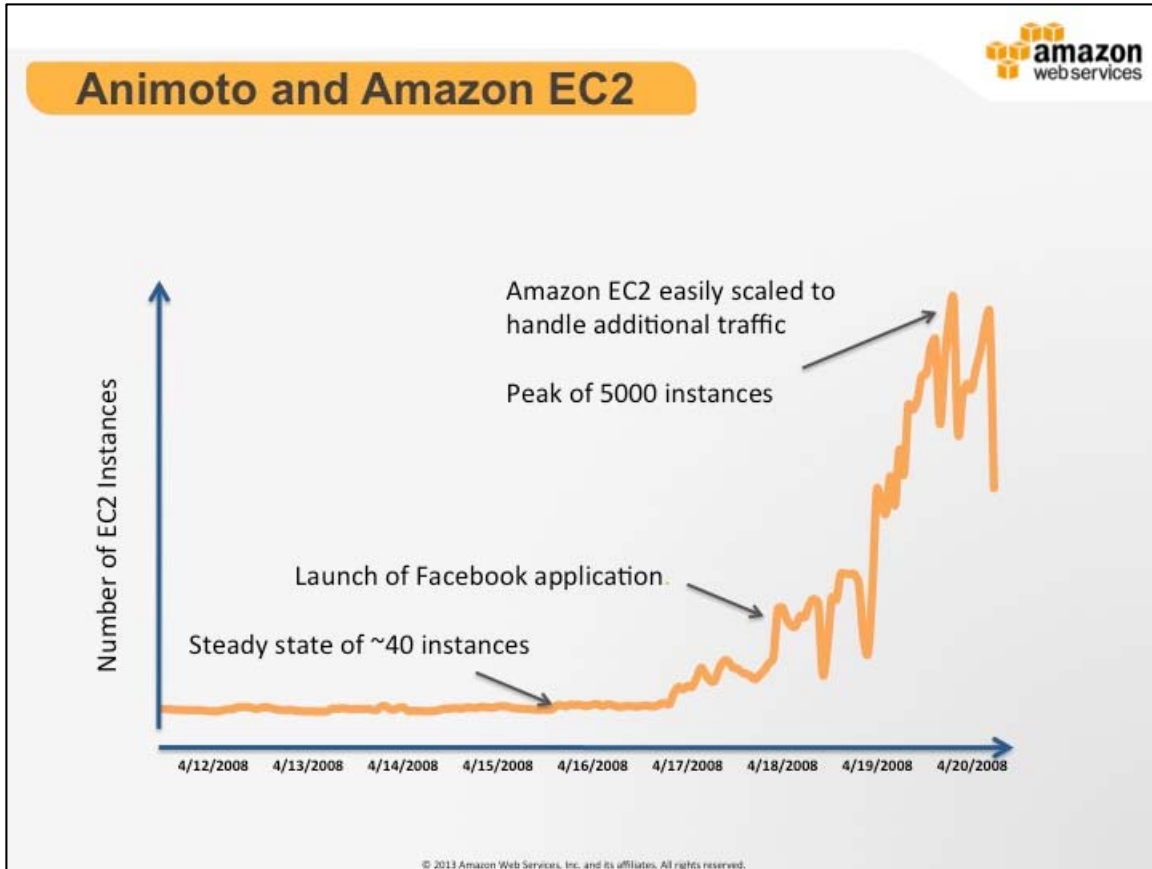
AWS component services empower you to control how and when you scale up or down. This concept can be boiled down into “Just-In-Time” provisioning, which essentially means that resources are there when you need them and not there when you don’t.

AWS offers on-demand provisioning and very fast boot times. Customers can leverage these features to reduce costs and increase operating efficiency by not running computing resources that are unused and by adding more resources when they would be better utilized.

The first example of “on/off” is a use case that would be like a gene processing job that would need to calculate a result set. While there is no work to be done the system could be off saving money and resources.

Another great example of leveraging elasticity is when you are running a website with variable peaks of utilization. In traditional IT, you have to estimate the level of resources that you need to provision to be able to have a starting point. This can translate over provisioning or under provisioning, which could lead to customer dissatisfaction because of poor performance or worse unfulfilled requests. Elasticity allows you to scale up to meet this demand and scale back down to make the best use of your resources.





## Animoto and Amazon EC2

In April of 2008, one of our customers, Animoto, saw a monster spike in traffic. Animoto has a product that helps you create web videos with music and graphics. They launched a Facebook app that lets people tell their friends when they've uploaded a video that includes that friend. You can see the spike in traffic that this new app caused. The X-axis represents time elapsed and the Y-axis represents the EC2 instances launched. Because they were using AWS, Animoto didn't have to do a thing—AWS took care of everything. Animoto used EC2 for processing the videos with music, SQS for queuing pictures and S3 for storage.

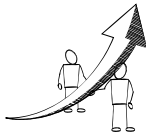


**Amazon Elastic Compute Cloud (EC2):** is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers.



**Amazon Simple Queue Service (SQS):** offers a reliable, highly scalable, hosted queue for storing messages as they travel between computers. By using Amazon SQS, developers can simply move data between distributed components of their applications that perform different tasks, without losing messages or requiring each component to be always available. Amazon SQS makes it easy to build an automated workflow, working in close conjunction with the Amazon Elastic Compute Cloud (Amazon EC2) and the other AWS infrastructure web services.

**Amazon Simple Storage Service (S3):** is storage for the Internet. It is designed to make web-scale computing easier for developers.



### More Information

- Learn more about EC2: <http://aws.amazon.com/ec2/>
- Learn more about SQS: <http://aws.amazon.com/sqs/>
- Learn more about S3: <http://aws.amazon.com/s3/>
- Customer success story video on Animoto: <http://aws.amazon.com/solutions/case-studies/animoto/>



# Security



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## Security

Amazon Web Services (AWS) delivers a highly scalable cloud computing platform with high availability and reliability, and the flexibility to enable customers to build a wide range of applications. In order to provide end-to-end security and end-to-end privacy, AWS builds services in accordance with security best practices, provides appropriate security features in those services, and documents how to use those features. AWS customers must use those features and best practices to architect an appropriately secure application environment. Enabling customers to ensure the confidentiality, integrity, and availability of their data is of the utmost importance to AWS, as is maintaining trust and confidence.

AWS provides a wide range of information regarding its IT control environment to customers through white papers, reports, certifications, and other third-party attestations. This information assists customers in understanding the controls in place relevant to the AWS services they use and how those controls have been validated by independent auditors. This information also assists customers in their efforts to account for and to validate that controls are operating effectively in their extended IT environment.

## Built to Standards

### Physical



### Certifications & Accreditations



### Hardware, Software & Network



### Security & Compliance Resources



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Quadrants of Security

Four quadrants of security built for Enterprise and Government standards.

## Physical Security

### Physical



- Datacenters in nondescript facilities
- Physical access strictly controlled
- Physical access logged and audited
- CTO can't get into the data centers
- Personnel separated into physical and logical access

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Physical Security

Physical (supplemental):

- Must pass two-factor authentication at least twice for floor access

## Certifications & Accreditations

- ISO 27001
- SSAE 16 / ISAE 3402 / SOC1 (formerly U.S. standard SAS-70 Type II)
- FISMA Moderate & DIACAP Controls; ITAR region
- HIPAA applications certified on AWS

### Certifications & Accreditations



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **Certification & Accreditations**

Certifications (supplemental):

- Payment Card Industry (PCI) Data Security Standard (DSS) Level 1
- Your applications do not automatically have this credentials
- Your applications also have to be compliant

## Hardware, Software & Network

- Systematic change management
- Phased updates deployment
- Safe storage decommission
- Automated monitoring and self-audit

### Hardware, Software & Network



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Hardware, Software & Network

Hardware, Software & Network (supplemental):

- Advanced network protection systems
- Manage our data standard to exacting standards

### Define the Groups

Tier	Port	Source
<b>Web</b>	80	0.0.0.0/0 or <u>amazon-elb-sg</u>
	443	0.0.0.0/0 or <u>amazon-elb-sg</u>
	22	App
<b>App</b>	22	207.171.191.60/32
	8000	Web
<b>DB</b>	3306	207.171.191.92/32
	3306	App
	22	App

## Security & Compliance Resources

- Security Overview & Best Practices
- AWS Risk & Compliance Whitepaper
- Creating HIPAA Compliant Applications
- We are audited by third parties on a regular basis

### Security & Compliance Resources



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Security & Compliance Resources

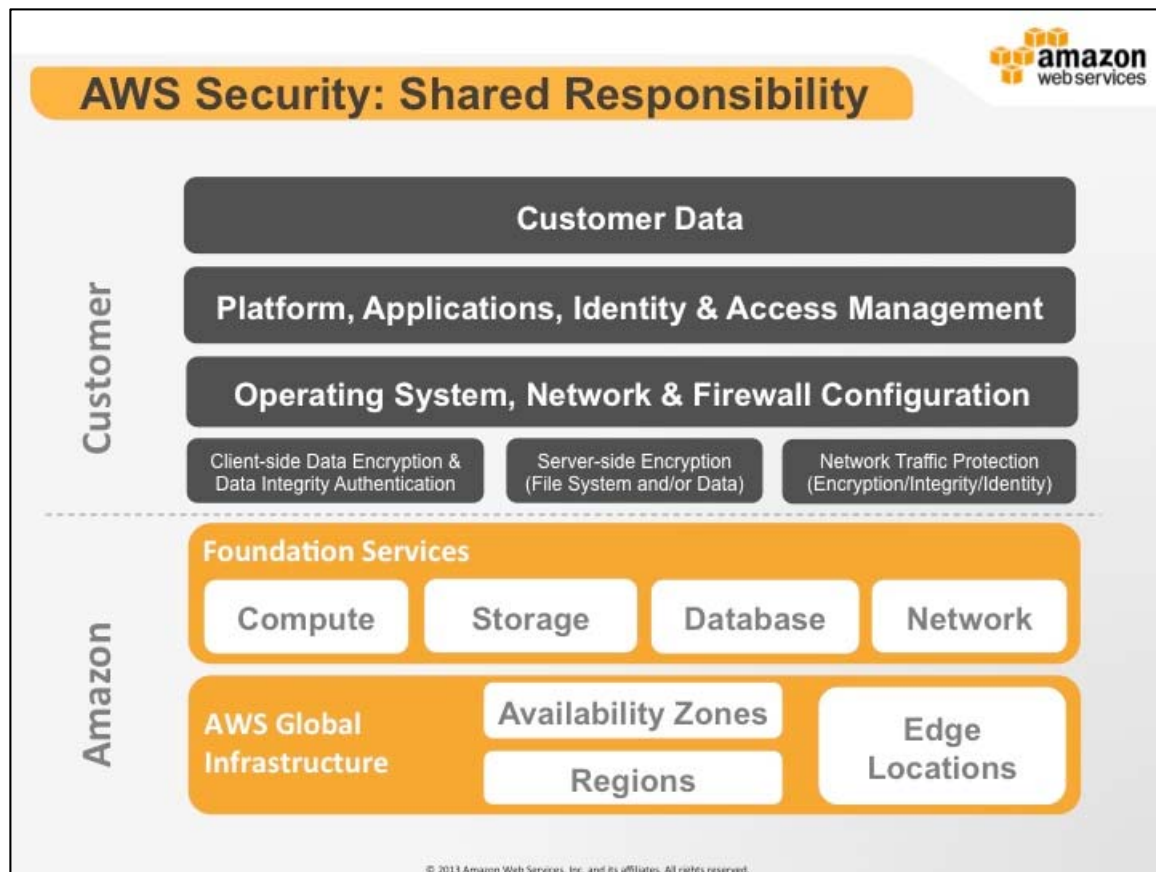
Security and Compliance (supplemental):

- Enterprise customers under NDA can get support from our security team in certifying their applications



#### More Information

- Security and Compliance Center: <http://aws.amazon.com/security>



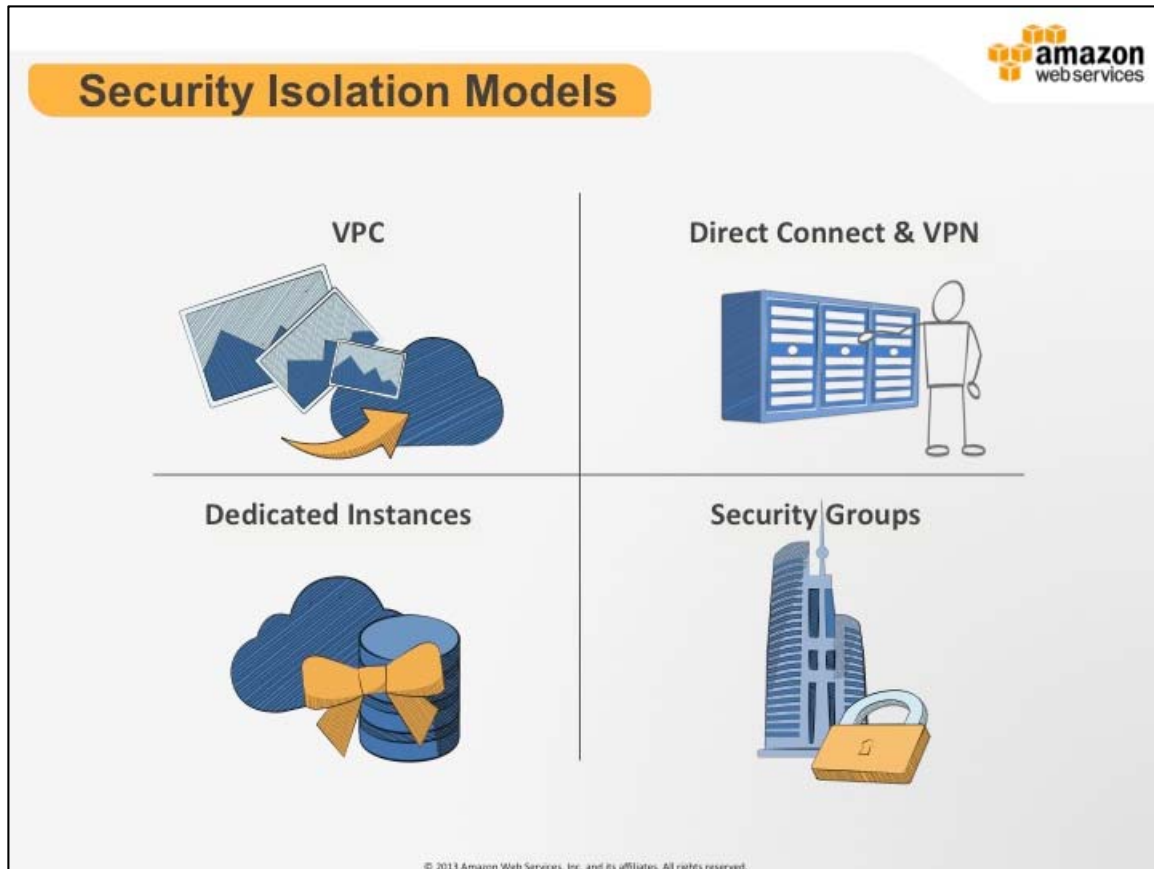
## AWS Security: Shared Responsibility

### AWS:

- Obtaining industry certifications and independent third party attestations described in this document
- Publishing information about the AWS security and control practices in whitepapers and web site content
- Providing certificates, reports, and other documentation directly to AWS customers under NDA (as required)

### Customer:

- Everything from the hypervisor up
- We provide security tools that you configure properly to meet your needs and maintain a strong security posture
- Can Amazon perform Windows Updates for your Instances? No. Because we have zero visibility into the Operating System layer



### AWS Security: Isolation Models

AWS security posture is divided in two major places. Everything that isn't the customer responsibility through the shared security model is covered by the isolation/deny by default model.

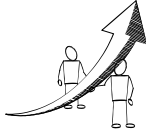
We provide a number of security controls to allow customers to define virtual network configurations and high-level firewall rules. Additionally, we have other techniques and features that you can use to control how data flows between components such as VPC and direct connect.

Lastly, all of our services are API and CLI accessible and access control policies can be written to restrict access where necessary. Policies follow a deny by default model, so specific access must be granted.



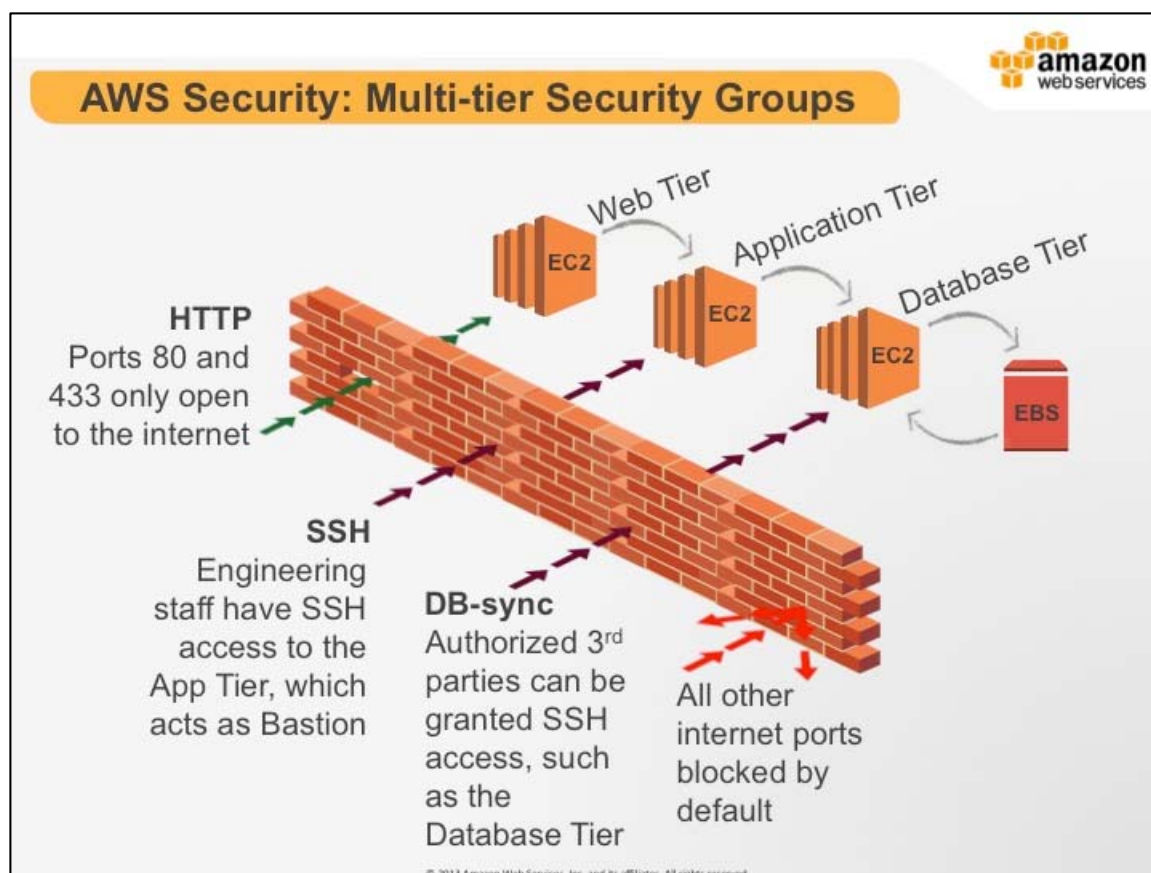


**Virtual Private Cloud (VPC):** is Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the Amazon Web Services (AWS) Cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.



### **More Information**

- Learn more about VPC: <http://aws.amazon.com/vpc/>



## AWS Security: Multi-tier Security Groups

The diagram above is an example of a security group configuration for a simple web application.

The Web Tier security group can accept traffic on port 80/443 from anywhere on the Internet if you select source 0.0.0.0/0. Alternatively, it might make more sense to only accept traffic from a load balancer so that individual clients cannot hammer a single server and the load balancer can perform its job.

Similarly the App tier can only accept traffic from the Web tier, and the DB tier can only accept traffic from the app tier.

Lastly, we have also added a set of rules to allow remote administration over SSH port 22. We have restricted remote access by funneling all traffic through the app tier and allowing access only from a specific IP. Once you have SSHed into an App tier server you can then connect to machines in the Web and DB security groups.

## AWS Security: Account Control

- AWS Master Accounts
- Unrestricted access
- Multi-Factor Authentication
- Consolidated Billing
- Invoice Billing

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS Security: Account Control**

- Logging in as the master account is akin to using Administrator or root account
- Master accounts have an implicit “ALLOW ALL” IAM policy applied to them. This cannot be changed or restricted.
- Master accounts shouldn’t be used for production systems because there is no way to restrict access to the EC2 Instance terminate command and if an account is revoked all services will be terminated. Instead we recommend using IAM users with policies and permissions that can be individually revoked.
- Consider using an MFA device on your master account
- Gemalto, Open Source authentication standard, and Google Authenticator
- There are use cases for companies to have multiple accounts, to facilitate this process we have consolidated billing and invoice billing
- Should everybody log in with the master account? Or would it be better if there was a way to delegate access?

## AWS Security: IAM

- “IAM” - AWS Identity and Access Management
- Service level access controls
- Non-master user accounts
- Identity Federation
- Roles
- Cross account API access

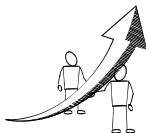
© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### AWS Security: IAM

IAM provides service level access control and limited identity management related to service level access. It should not be used for application level security and does not provide resource level access control.



**AWS Identity and Access Management (IAM):** enables you to securely control access to AWS services and resources for your users. Using IAM you can create and manage AWS users and groups and use permissions to allow and deny their permissions to AWS resources.



#### More Information

Learn more about IAM: <http://aws.amazon.com/iam/>

## AWS Security: IAM Cont.

- Users and Groups within Accounts
- Unique security credentials
- Policies control access to AWS APIs
- API calls must be signed by either:
- Deep integration into some Services
- AWS Management Console supports User log on
- Not for Operating Systems or Applications

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS Security: IAM**

The AWS IAM service allows customers to create Users, Groups, and Roles and associate IAM access control policies against them. You can also assign users to groups as a means of easing user management. If multiple policies apply to a single end user, AWS IAM aggregates those permissions in a least privileged model, which means that we have a deny bias.

## AWS API Authentication

- Access ID & Secret Key
- Used by Command Line, 3rd Party Tools or programs to sign (authenticate) AWS API requests!
  - Launch, Start, Stop, Terminate EC2 instances
  - Create, attach, delete storage
  - Configure Security Groups
  - Configure AWS infrastructure changes
  - Things you can do through the AWS Console

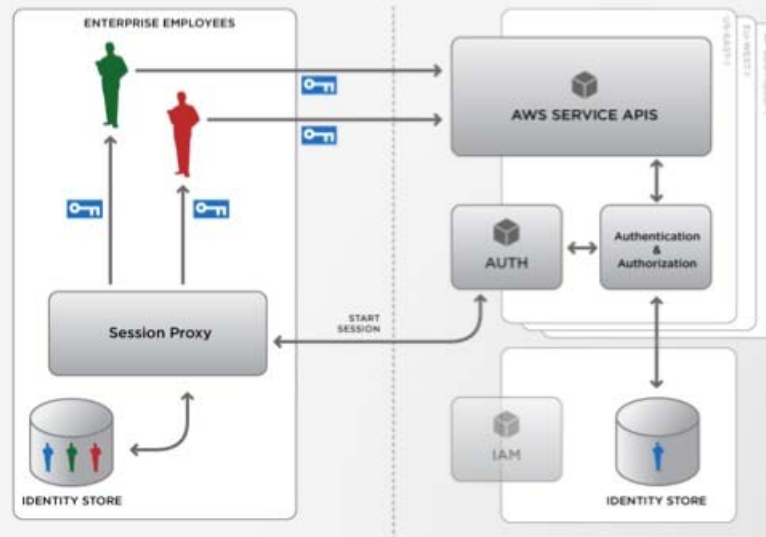
© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS API Authentication**

AWS IAM Users are created without any credentials associated with them by default. In order for an IAM user to be of any use you must provision access credentials. There are three major types of access credentials.

## AWS Security: IAM

### Identity Federation to AWS Management Console and APIs



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

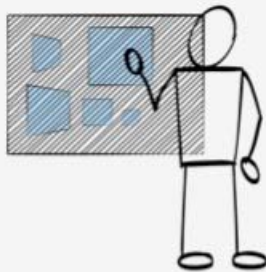
### AWS Security: IAM

There is a hard limit of 5000 IAM users that can be created for any single account. Even at that scale, user management can become problematic and cumbersome.

To allow IAM to scale to tens of thousands of users we created a Secure Token Service that will allow you to generate secure tokens that can then be used to authenticate with AWS service APIs or even the console. Token can be configured to have valid duration of between 15 minutes and 36 hours.

AWS maintains sample boilerplate code to help get you started in creating a session proxy that will authenticate your users with your own Identity store and then go through the process of generating tokens. We sometimes call this a token vending machine or a session proxy that will handle authentication and authorization, potentially in a single sign-on fashion.

# IAM Demo



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## **IAM Demo**

1. How to create an IAM user
2. Assign the user a policy
3. Generate a password
4. Download the CSV



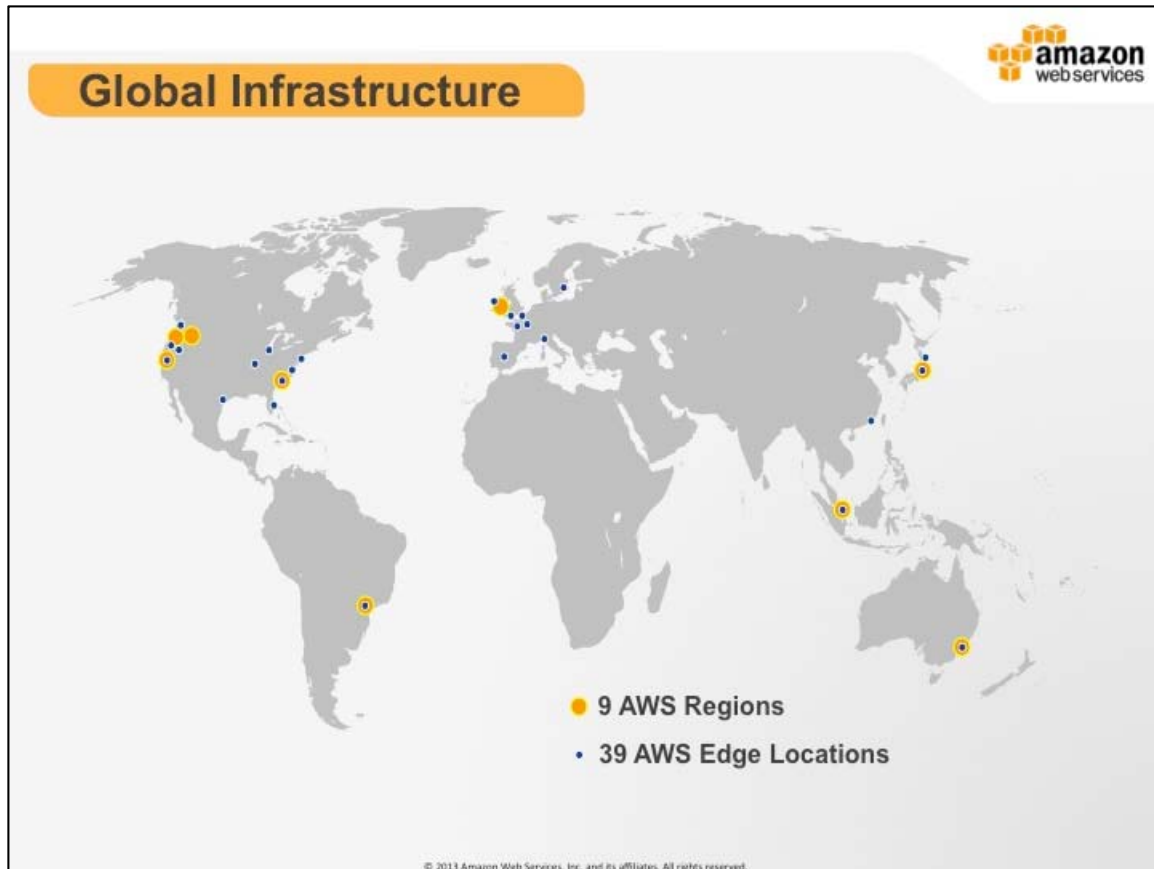
# Global Infrastructure



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## **Global Infrastructure**

Amazon Web Services serves hundreds of thousands of customers in more than 190 countries. We are going to explain more in this section.



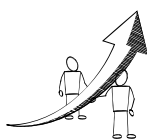
## Global Infrastructure

We are steadily expanding global infrastructure to help our customers achieve lower latency and higher throughput, and to ensure that their data resides only in the Region they specify. As our customers grow their businesses, AWS will continue to provide infrastructure that meets their global requirements.



**Regions:** are used to run applications and workloads to reduce latency to end-users while avoiding the up-front expenses, long-term commitments, and scaling challenges associated with maintaining and operating a global infrastructure.

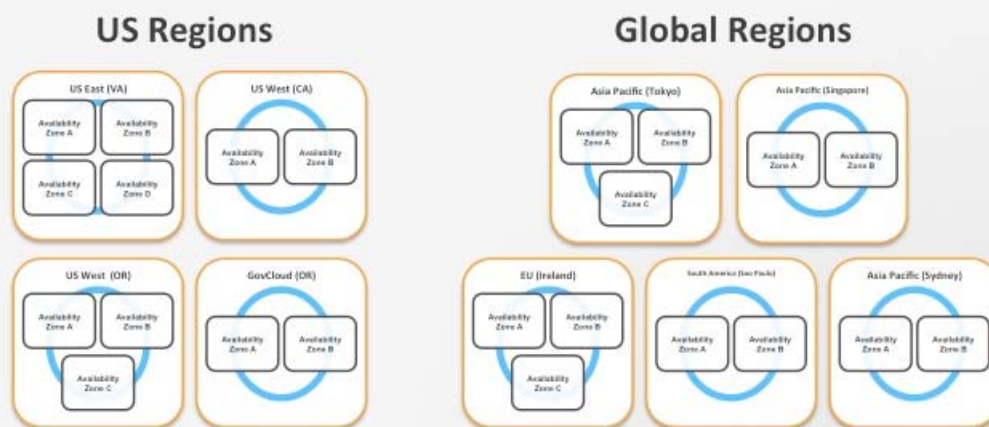
**Edge Locations:** helps lower latency and improves performance for end users. Two of our services run in our edge locations. Those services are Route 53 and CloudFront.



### More Information

Global Infrastructure locations: <http://aws.amazon.com/about-aws/globalinfrastructure/>

## AWS Regions & Availability Zones



Customer Decides Where Applications and Data Reside

*Note: Conceptual drawing only. The number of Availability Zones may vary.*

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### AWS Regions & Availability Zones

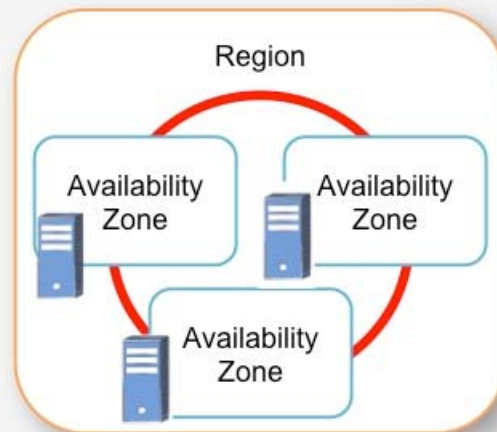
All regions have at least two availability zones. Many of our regions have more than two. Availability zones are given account relative names to help distinguish between AZs. It is important to note that these names differ from account to account.



**Availability Zones:** are physically distinct groups of data centers. A region is made of multiple availability zones so as to allow our customers the ability to spread their computing resource across multiple tier 1 ISPs and power providers. Where natural disasters are a serious consideration we do our best to isolate AZs from each other. For example, where earthquakes are a problem we would not build two AZs on the same fault line.

## Capacity Provisioning

Q: Where should capacity be provisioned?



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Capacity Provisioning

AWS highly recommends provisioning your resources across multiple availability zones. If you have more than one server, it costs nothing extra to run them across more than one AZ and doing so will get you added redundancy. Should a single AZ have a problem, all assets in your second AZ will be unaffected.

## Knowledge Check



Q: What is a good analogy for your AWS Master account?

Q: What are Edge Locations?

Q: AWS Cloud Computing is:

Q: AWS Security is:

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

1. \_\_\_\_\_  
\_\_\_\_\_
2. \_\_\_\_\_  
\_\_\_\_\_
3. \_\_\_\_\_  
\_\_\_\_\_
4. \_\_\_\_\_  
\_\_\_\_\_

# AWS Services



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

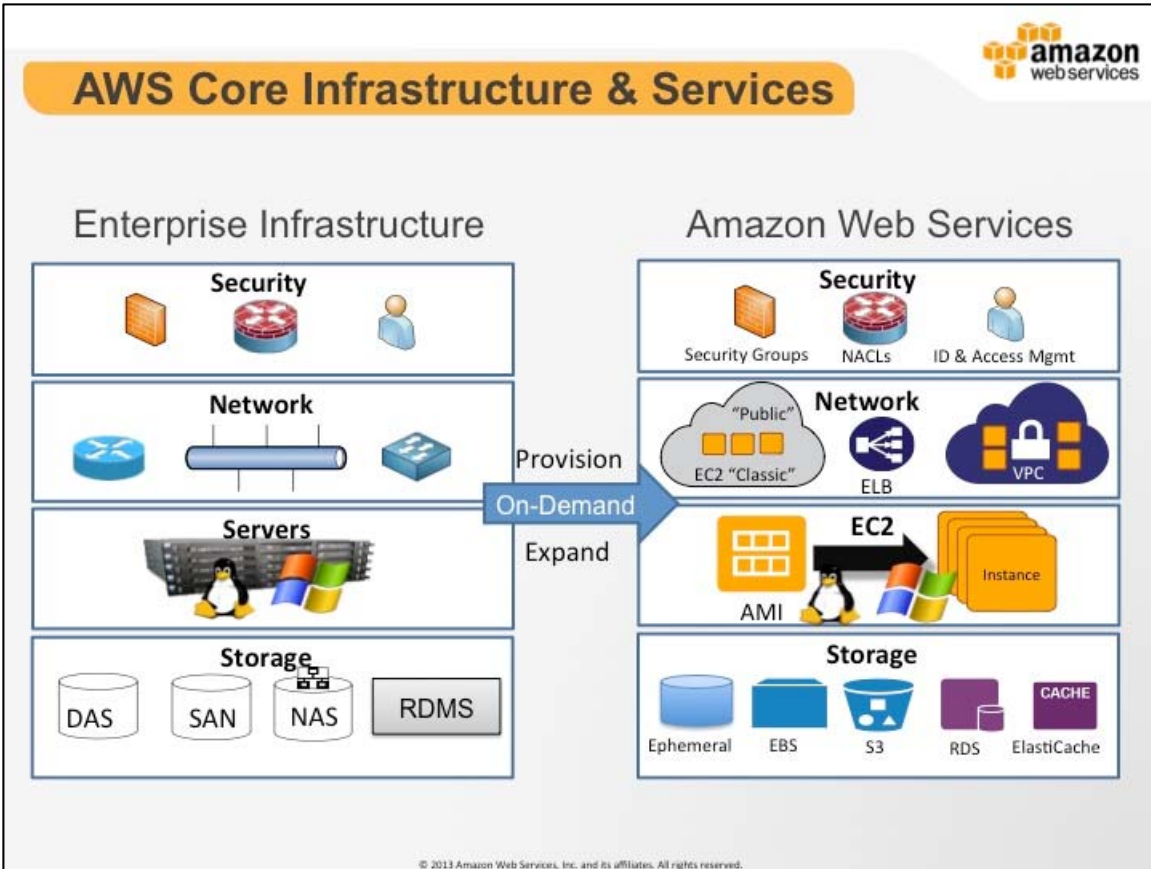
## AWS Services

In this section we will begin covering specific services.



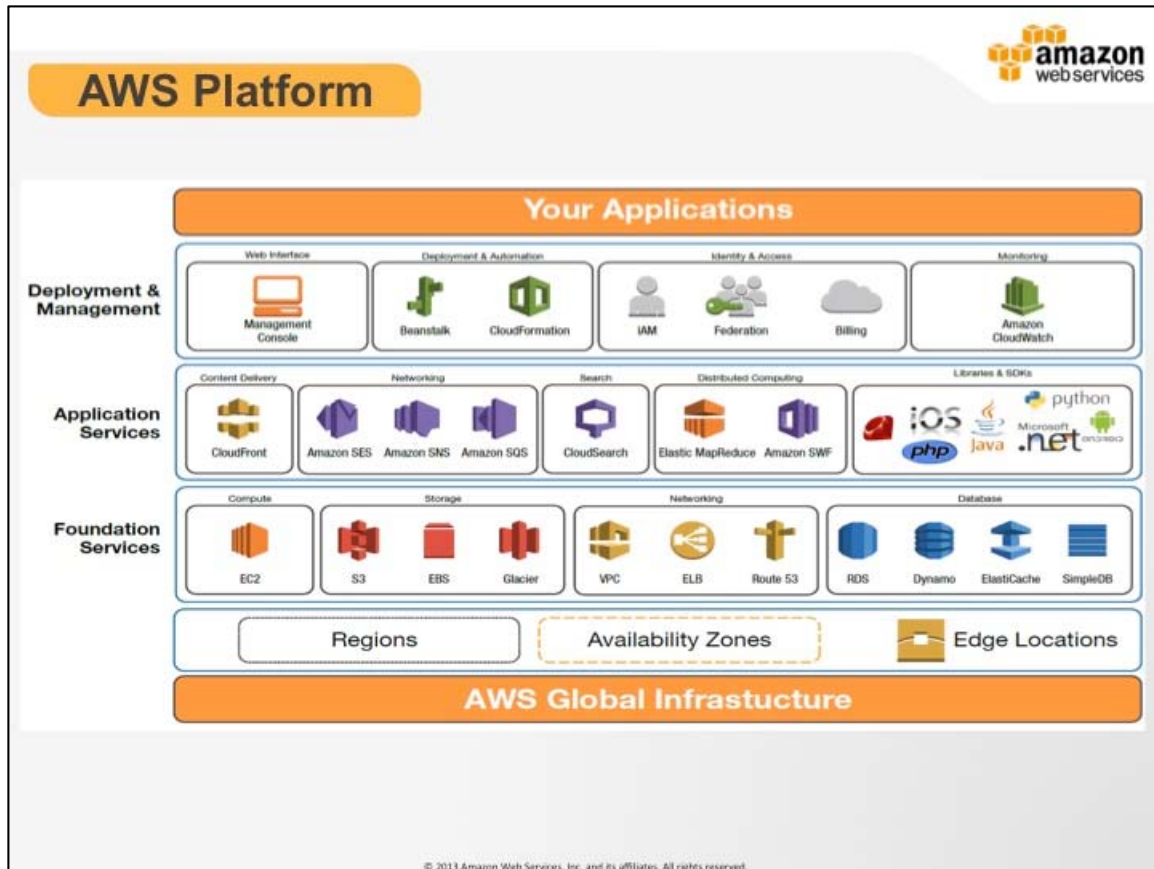
### More Information

AWS products and services: <http://aws.amazon.com/products/>



### AWS Core Infrastructure & Services

Many of our services have analogs in the Traditional IT space and terminology.



## AWS Platform

We will be covering the following deployment management and services in this training:

- CloudFormation
- IAM
- CloudWatch
- EC2
- S3
- EBS
- Glacier
- VPC
- RDS



# Storage Services



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## **Storage Services**

AWS offers many storage services and features that are a significant paradigm shift from the traditional IT storage space. In traditional IT, most storage exists as some form of hard drive space. In AWS, storage services have a varying mix of durability, availability, and cost.

## AWS Storage Products and Services



Amazon Glacier



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS Storage Products and Services**

In this section we are going to cover three of our more popular storage services, specifically, high-level features, costs, and best practices.

## Amazon Simple Storage Service: S3



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### AWS Simple Storage Service: S3

S3 is storage for the Internet. It is designed to make web-scale computing easier for developers. Amazon S3 provides a simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web. It gives any developer access to the same highly scalable, reliable, secure, fast, inexpensive infrastructure that Amazon uses to run its own global network of web sites.



**Amazon Simple Storage Service (S3):** is storage for the Internet. It is designed to make web-scale computing easier for developers.



#### More Information

- Learn more about S3: <http://aws.amazon.com/s3/>

## S3: Pricing

- Bandwidth
  - All data into AWS is free
  - First GB out each month is free
  - Additional data out starts at \$.12 per GB/month
- Used Capacity
  - \$.09 per GB / Month, pricing for US-East
  - Review Pricing Calculator

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **S3: Pricing**

S3 pricing is based on capacity and bandwidth actually used. Since S3 is an Internet scale service that runs natively across an entire region, it can handle significant request throughput and bandwidth output. All bandwidth into S3 is free, but we charge a rate on bandwidth out.

Most importantly, since S3 can handle any amount of data it is important to distinguish that you only pay for the amount of space you use. Space is priced in a prorated GB per month.

## S3: Best Practice

- 100 “Bucket” Limit
- Able to store unlimited number of Objects in a Bucket
- Objects from 1B-5 TB; no bucket size limit
- Highly available storage for the Internet (99.99%)
- HTTP/S endpoint to store and retrieve any amount of data, at any time, from anywhere on the web
- Highly scalable, reliable, fast, and inexpensive
- Over 1 trillion objects stored
- Peak requests 835,000+ per second

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **S3: Best Practices**

Here are some important best practices to take into consideration when utilizing Amazon S3.

Bucket names directly translate into publicly addressable DNS names. As such they are global to a region and must be unique. Because of this, there is a hard limit of 100 buckets per account. This is to avoid bucket name squatting. It also means that you should take care when deciding what requires a bucket. Some customer chose to use one bucket per application. Others choose to use one bucket per customer.

## S3 & Glacier

Q: Does Amazon S3 provide capabilities for archiving objects to lower cost storage options?

Q: How can I store my data using the Amazon Glacier option?

Q: Can I use the Amazon S3 APIs or Management Console to list objects that I've archived to Amazon Glacier?

Q: Can I use Amazon Glacier APIs to access objects that I've archived to Amazon Glacier?

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

1. \_\_\_\_\_  
\_\_\_\_\_
2. \_\_\_\_\_  
\_\_\_\_\_
3. \_\_\_\_\_  
\_\_\_\_\_
4. \_\_\_\_\_  
\_\_\_\_\_



## Amazon Glacier

Amazon Glacier is an extremely low-cost storage service that provides secure and durable storage for data archiving and backup. Amazon Glacier is optimized for data that is infrequently accessed and for which retrieval times of several hours are suitable. You store data in Amazon Glacier as archives. An archive can represent a single file or you may choose to combine several files to be uploaded as a single archive. Retrieving archives from Amazon Glacier requires the initiation of a job. Jobs typically complete in 3 to 5 hours.



**Amazon Glacier:** is an extremely low-cost storage service that provides secure and durable storage for data archiving and backup.



### More Information

- Glacier: <http://aws.amazon.com/glacier/>

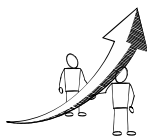
## Glacier: Pricing

- Request Pricing Online
  - Pick the Region
- Data Transfer OUT From Amazon Glacier to Internet
- A low-cost storage service for data archiving and backup
- \$0.01 per GB / Month, pricing for US-East
  - Review Pricing Calculator

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Glacier: Pricing

Glacier is designed with the expectation that retrievals are infrequent and unusual, and data will be stored for extended periods of time. You can retrieve up to 5% of your average monthly storage (pro-rated daily) for free each month. If you choose to retrieve more than this amount of data in a month, you are charged a retrieval fee starting at \$0.01 per gigabyte. Learn more. In addition, there is a pro-rated charge of \$0.03 per gigabyte for items deleted prior to 90 days.



#### More Information

- Glacier Pricing: <http://aws.amazon.com/glacier/pricing/>



## Glacier: Best Practice

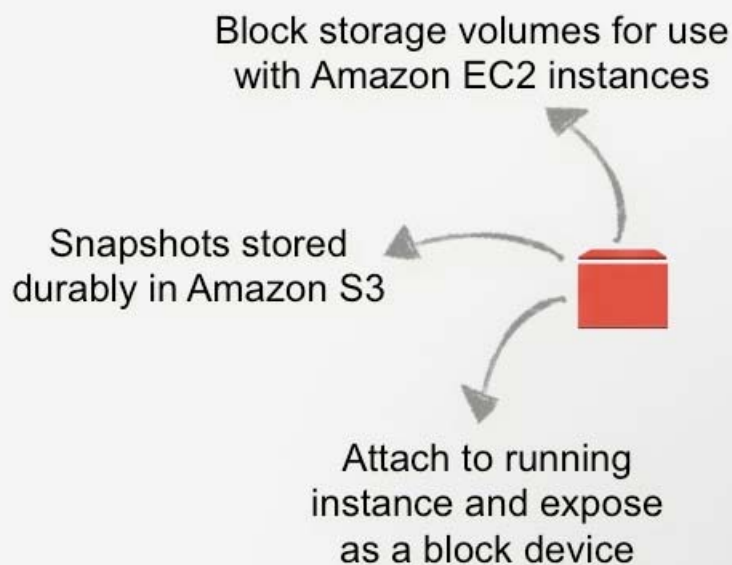
- Optimized for data that is infrequently accessed
- Retrieval times measured in hours not days or weeks
- Annual durability of 99.999999999% for an archive
- AES 256 data at rest encryption
- Data stored as archives within a vault. Vaults are located within a specific AWS region

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **Glacier: Best Practice**

Amazon Glacier is an extremely low-cost storage service that provides secure and durable storage for data archiving and backup. In order to keep costs low, Amazon Glacier is optimized for data that is infrequently accessed and for which retrieval times of several hours are suitable. With Amazon Glacier, customers can reliably store large or small amounts of data for as little as \$0.01 per gigabyte per month, a significant savings compared to on-premises solutions.

## Amazon Elastic Block Store: EBS



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS Elastic Block Store (EBS)**

EBS provides block level storage volumes for use with Amazon EC2 instances. Amazon EBS volumes are off-instance storage that persists independently from the life of an instance. Amazon EBS provides highly available, highly reliable, predictable storage volumes that can be attached to a running Amazon EC2 instance and exposed as a device within the instance. Amazon EBS is particularly suited for applications that require a database, file system, or access to raw block level storage. We will talk more about EC2 later.

## EBS: Pricing

- Pay for what you Provision (pricing for US-East)
  - \$0.10 per GB/month for standard EBS volume
  - \$0.10 per 1 million I/O requests
  - \$0.125 per GB-month for PIOPS EBS volume
  - Review Pricing Calculator



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### EBS: Pricing

Volume storage for Standard volumes is charged by the amount you provision in GB per month until you release it. Volume I/O for Standard volumes is charged by the number of requests you make to your volume. Programs like IOSTAT can be used to measure the exact I/O usage of your system at any time. However, due to varying levels of caching of applications and operating systems, you may see a lower number of I/O requests on your bill for Standard volumes than is seen by your application unless you sync all of your I/Os to disk.

As with Standard volumes, volume storage for Provisioned IOPS volumes is charged by the amount you provision in GB per month. With Provisioned IOPS volumes, you are also charged by the amount you provision in IOPS (input/output operations per second) X the percentage of days you provision for the month. For example, if you provision a volume with 1000 IOPS, and keep this volume for 15 days in a 30 day month, then in the Virginia Region, you would be charged \$50 for the IOPS that you provision ( $\$0.10 \text{ per provisioned IOPS-Month} * 1000 \text{ IOPS Provisioned} * 15 \text{ days}/30$ ).

Snapshot storage is based on the amount of space your data consumes in Amazon S3. Because data is compressed before being saved to Amazon S3, and Amazon EBS does not save empty blocks, it is likely that the size of a

snapshot will be considerably less than the size of your volume. For the first snapshot of a volume, Amazon EBS will save a full copy of your data to Amazon S3. However for each incremental snapshot, only the part of your Amazon EBS volume that has been changed will be saved to Amazon S3.

## EBS: Best Practice

### EBS Volumes are in a Single Availability Zone



Note: Conceptual diagram only

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### EBS : Best Practices

Amazon EBS volumes are created in a particular Availability Zone and can be from 1 GB to 1 TB in size. Once a volume is created, it can be attached to any Amazon EC2 instance in the same Availability Zone. Once attached, it will appear as a mounted device similar to any hard drive or other block device. At that point, the instance can interact with the volume just as it would with a local drive, formatting it with a file system or installing applications on it directly.

A volume can only be attached to one instance at a time, but many volumes can be attached to a single instance. This means that you can attach multiple volumes and stripe your data across them for increased I/O and throughput performance. This is particularly helpful for database style applications that frequently encounter many random reads and writes across the dataset. If an instance fails or is detached from an Amazon EBS volume, the volume can be attached to any other instance in that Availability Zone.

Amazon EBS volumes can also be used as boot partitions for Amazon EC2 instances, which allows you to increase the size of your boot partition up to 1 TB, preserve your boot partition data beyond the life of your instance, and bundle your AMI in one-click. You can also stop and restart instances that boot from Amazon EBS volumes while preserving state, with very fast start-up times.

## S3 & EBS

	EBS	Amazon S3
Paradigm	File system	Object store
Performance	very fast	fast
Redundancy	Within data center	Across data centers
Security	Visible only to your EC2	Public Key / Private Key
Pricing	\$0.10/GB/month <u>allocated</u>	\$0.125/GB/month <u>stored</u>
Access from the Net?	No(2)	Yes(1)
Typical use case	It's a disk drive	Write once, read many

- (1) Only with proper credentials, unless ACLs are world readable  
 (2) Accessible from Net if mounted to server and set up as FTP, etc.

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### S3 & EBS

The most significant differences between S3 and EBS are nature in which they are written to and accessed. EBS volumes are simply network attached hard drives that can be written to or read from at a block level.

S3 is an object level storage medium. This means that you must write whole objects at a time. If you change one small part of a file, you must still rewrite the entire file in order to commit the change to S3. This can be very expensive if you have frequent writes to the same object. S3 is optimized for write one read many use cases.

Lastly, the other major difference is in how the services are priced. With S3 you pay for what you use, and with EBS you pay for what you provision.



## Snapshots

Amazon EBS provides the ability to back up point-in-time snapshots of your data to Amazon S3 for durable recovery. Amazon EBS snapshots are incremental backups, meaning that only the blocks on the device that have changed since your last snapshot will be saved. If you have a device with 100 GBs of data, but only 5 GBs of data has changed since your last snapshot, only the 5 additional GBs of snapshot data will be stored back to Amazon S3. Even though the snapshots are saved incrementally, when you delete a snapshot, only the data not needed for any other snapshot is removed. So regardless of which prior snapshots have been deleted, all active snapshots will contain all the information needed to restore the volume. In addition, the time to restore the volume is the same for all snapshots, offering the restore time of full backups with the space savings of incremental.

New volumes created from existing Amazon S3 snapshots load lazily in the background. This means that once a volume is created from a snapshot, there is no need to wait for all of the data to transfer from Amazon S3 to your Amazon EBS volume before your attached instance can start accessing the volume and all of its data. If your instance accesses a piece of data which hasn't yet been loaded, the volume will immediately download the requested data from Amazon S3, and then will continue loading the rest of the volume's data in the background.

# S3 Exercise



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## S3 Exercise #1

1. Create a bucket in S3.
2. Add an object to the bucket.
3. View the object.
4. Move the object.
5. Delete the object and the bucket in S3.



## Getting Started with S3



### Exercise #1

- Create a Bucket
- Add an Object to a Bucket
- View an Object
- Move an Object
- Delete an Object and Bucket

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.



### More Information

There are step-by-step directions at the end of this guide.

# Compute Services

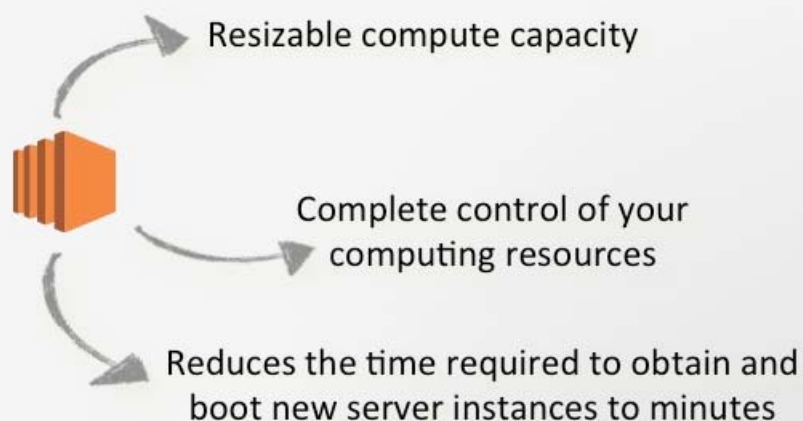


© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## **Computer Services**

AWS Compute Services are at the core of all the products and services that make up the Amazon Cloud.

## Amazon Elastic Compute Cloud: EC2



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Amazon Elastic Compute Cloud: EC2

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers. Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use. Amazon EC2 provides developers the tools to build failure resilient applications and isolate themselves from common failure scenarios.



**Tags:** AWS resources can be tagged to simplify management. Tags are key, value pair that you define. E.g. you can define NAME=PRODSERVER1 for an EC2 instance or DEPARTMENT=FINANCE to later run a report against DEPARTMENT tags for each resource to be used for cost allocation to different departments.



**Amazon Machine Image (AMI):** is an abbreviation for Amazon Machine Image. An Amazon Machine Image (AMI) is an encrypted machine image stored in Amazon S3. It contains all the information necessary to boot instances of your software. Instance is a result of running a system. After you launch an Amazon Machine Image (AMI), the resulting running system is referred to as an instance.

**Compute:** The compute feature allows you to take advantage of thousand of networked servers for virtually unlimited compute power.

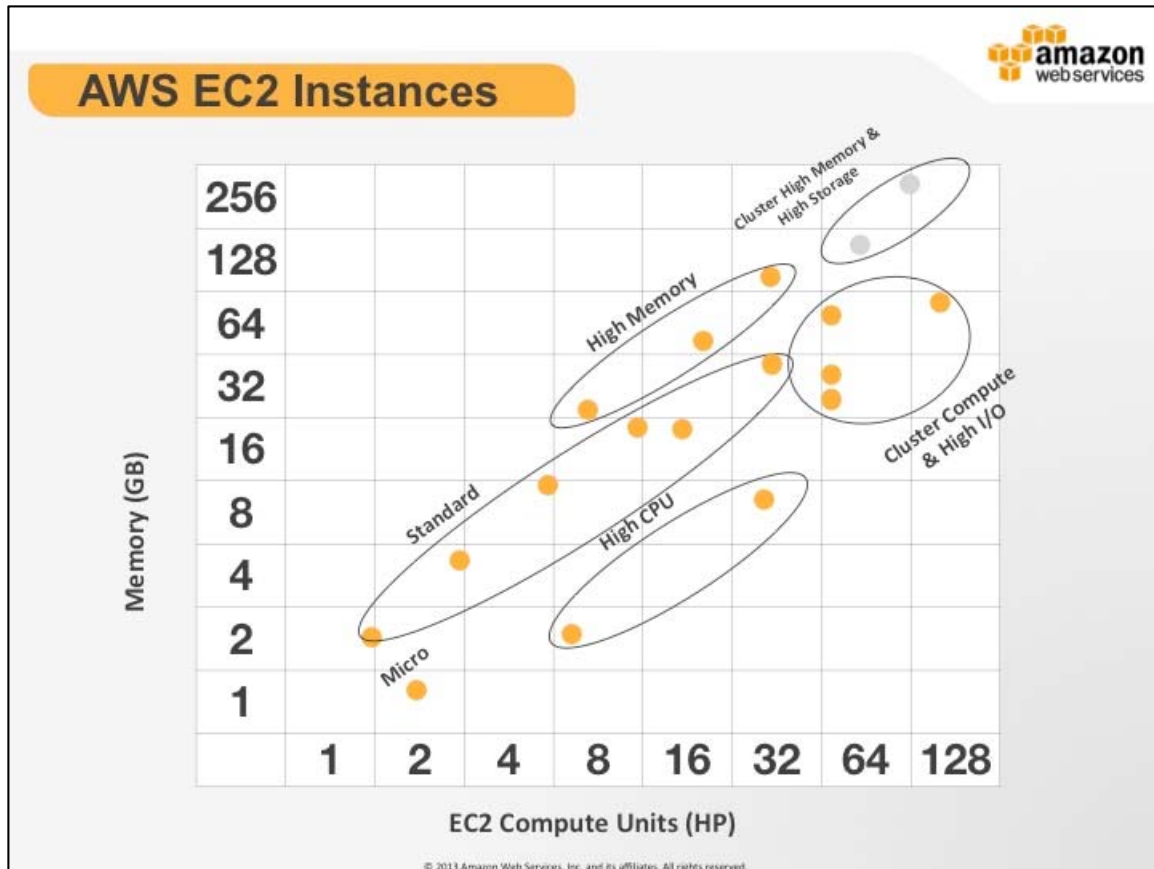
## EC2: Best Practice

- Resizable compute capacity in 19 instance types
- Reduces the time required to obtain and boot new server instances to minutes or seconds
- Scale capacity as your computing requirements change
- Pay only for capacity that you actually use
- Choose Linux or Windows
- Deploy across Regions and Availability Zones for reliability

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **EC2: Best Practices**

Amazon EC2 presents a true virtual computing environment, allowing you to use web service interfaces to launch instances with a variety of operating systems, load them with your custom application environment, manage your network's access permissions, and run your image using as many or few systems as you desire.



## AWS EC2 Instances

AWS EC2 provides the flexibility to choose from a number of different instance types to meet your computing needs. Each instance provides a predictable amount of dedicated compute capacity and is charged per instance-hour consumed.

First generation (M1) Standard instances provide customers with a balanced set of resources and a low cost platform that is well suited for a wide variety of applications.

Second generation (M3) Standard instances provide a balanced set of resources and a higher level of processing performance compared to M1 instances. These are ideal for applications that require higher absolute CPU and memory performance. Encoding and high traffic content management systems are examples of applications that will benefit from the performance of M3 instances.

High-Memory Instances offer large memory sizes for high throughput applications, including database and memory caching applications. High-CPU Instances have proportionally more CPU resources than memory (RAM) and are well suited for compute-intensive applications. There are also various high storage and cluster computer instance types available.

## EC2: Instances

### On-Demand Instances

Unix/Linux instances start at \$0.08/hour USD in the US East Region

Pay as you go for compute power

Pay only for what you use, no up-front commitments or long-term contracts

### Reserved Instances

1- or 3-year terms

Light/Medium/Heavy

Pay low up-front fee, receive significant hourly discount

Helps ensure compute capacity is available when needed

### Spot Instances

Bid on unused EC2 capacity

Spot Price based on supply/demand, determined automatically

Spot Price below bid, instances start  
Spot Price above bid, instances stop

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## EC2: Instances

There are a variety of billing constructs that you can use to make sure your application runs in the most cost effective way possible.

### AWS Free Usage Tier:

To help new AWS customers get started in the cloud

New AWS customers will be able to run a free Amazon EC2 Micro Instance and a free Amazon RDS Micro Instance for a year, while also leveraging a free usage tier for Amazon S3, Amazon Elastic Block Store, Amazon Elastic Load Balancing, and AWS data transfer.

AWS's free usage tier can be used for anything you want to run in the cloud: launch new applications, test existing applications in the cloud, or simply gain hands-on experience with AWS



### More Information

For more information on the AWS Free Usage Tier:

<http://aws.amazon.com/free/>

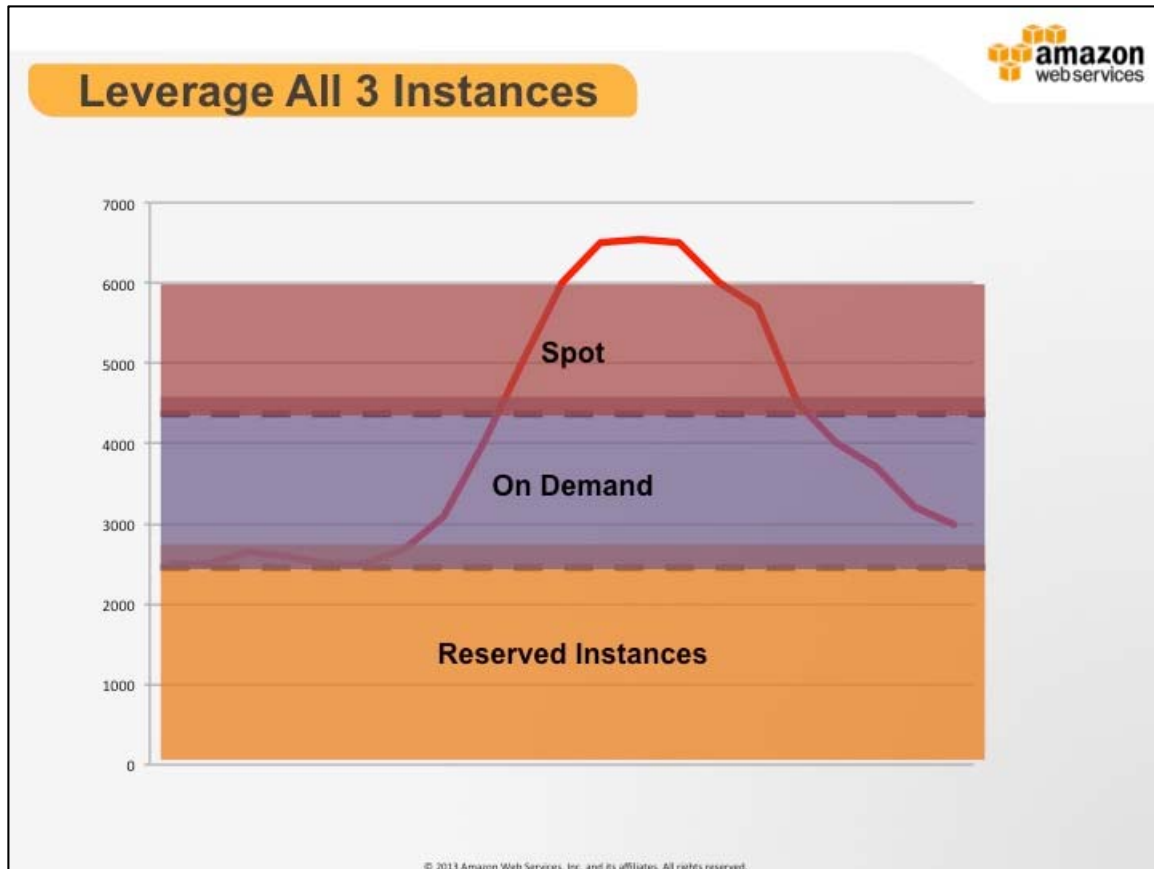


## Reserved Instances

- Pay up-front fee to receive significant hourly discount
- Cost / Predictability
- Get priority compute capacity when needed

Use Cases: Applications with steady state or predictable usage; Applications that require reserved capacity, including disaster recovery; Users are able to make upfront payments to reduce overall computing





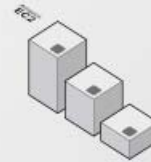
### Leverage all Three Models

In this example, we have a variable usage curve. Computer resource below the 3000 mark is an example the applications steady state. For this level of computing resource the customer might consider utilizing Reserved Instance to reduce their expected costs because they know they will use at least that many resources.

For variable elevated usage periods the customer would then perhaps make use of On-Demand or Spot market instances to supplement their always-on fleet.

## Using EC2

- Create or select a pre-configured Amazon Machine Image (AMI)
- Configure security and network access
- Choose instance type(s)
- Determine what to run
- Pay only for the resources used



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Using EC2

1. Select a pre-configured, Amazon Machine Image (AMI) to get up and running immediately. Or create an AMI containing your applications, libraries, data, and associated configuration settings. Amazon EC2 provides templates known as Amazon Machine Images (AMIs) that contains a software configuration (for example, an operating system, an application server, and applications.) You use these templates to launch an instance, which is a copy of the AMI running as a virtual server in the cloud.
2. You can launch different types of instances from a single AMI. An instance type essentially determines the hardware of the host computer used for your instance. Each instance type offers different compute and memory capabilities. Select an instance type based on the amount of memory and computing power that you need for the application or software that you plan to run on the instance. You can launch multiple instances from an AMI.
3. Your instance keeps running until you stop or terminate it, or until it fails. If an instance fails, you can launch a new one from the AMI.
4. Configure security and network access on your Amazon EC2 instance.

5. Choose which instance type(s) you want, then start, terminate, and monitor as many instances of your AMI as needed, using the web service APIs or the variety of management tools provided.
6. Determine whether you want to run in multiple locations, utilize static IP endpoints, or attach persistent block storage to your instances.
7. Pay only for the resources that you actually consume, like instance-hours or data transfer.

## Amazon Machine Images: AMIs

- Building blocks of EC2 instances
- An AMI is like a template of a computer's root volume
- Can be public or private
- Create hardened or gold “Images” of your EC2 infrastructure

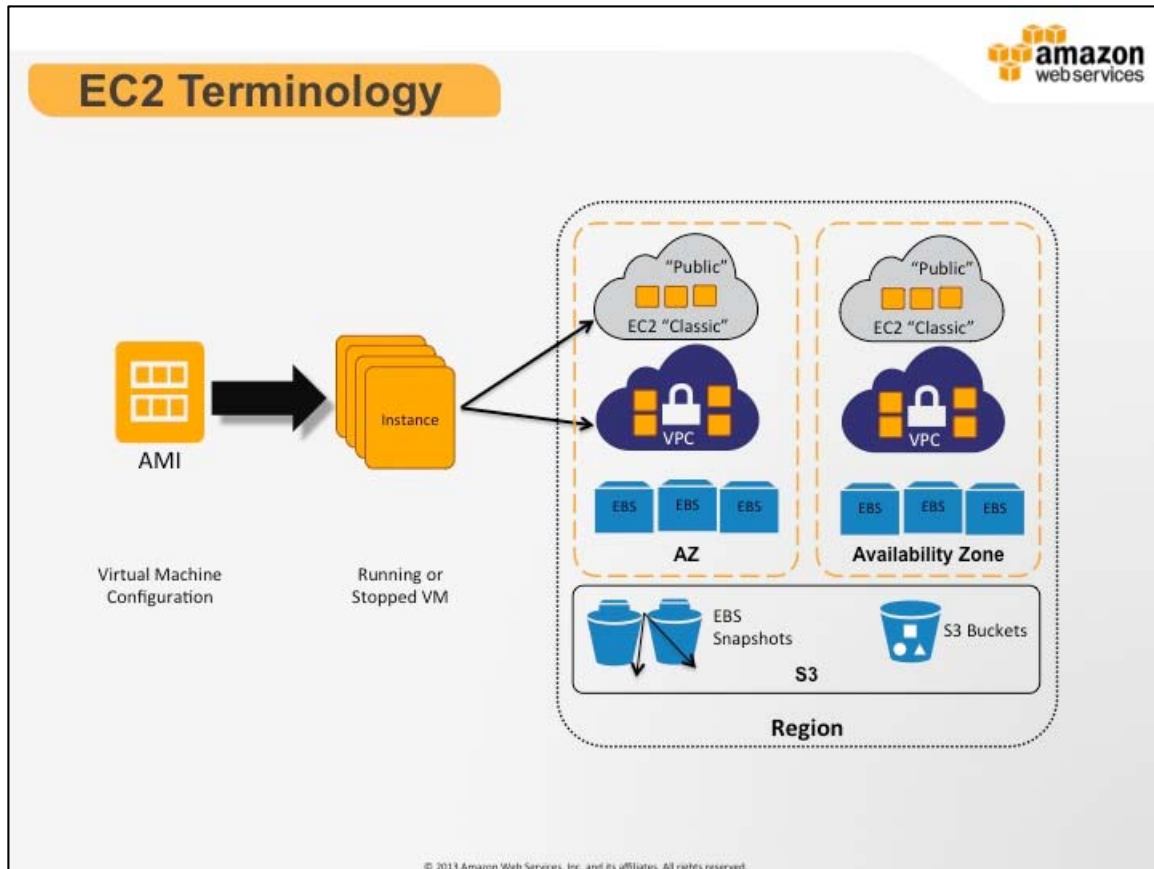


AMI

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **Amazon Machine Images (AMI)**

Amazon Machine Images are the basic building block for launching virtual machine in EC2. They can be either S3 or EBS backed and they contain the operating system and various other metadata that is required for initial boot.



## EC2 Terminology

When launching an EC2 instance, you will initially begin by selecting an AMI and an Instance type (optionally an availability zone). You will then have to select various configuration options for your EC2 instance such as the networking typology (EC2 Classic or VPC), what sort of storage devices you would like attached (number and sizes of EBS volumes, ephemeral drives), and whether or not you will be creating the storage devices from any S3 snapshots.

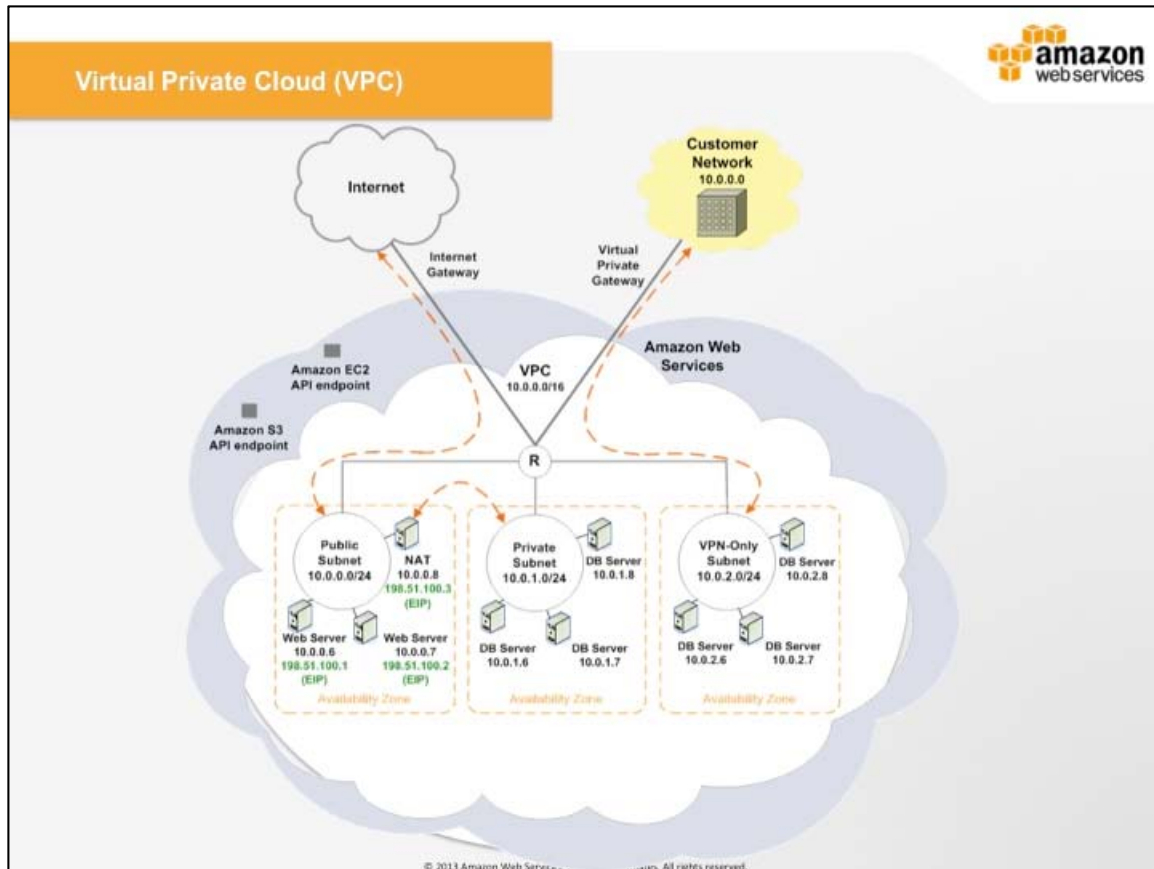
## EC2 Security Groups

Public EC2	Virtual Private Cloud
Inbound Only	Inbound and Outbound
TCP, UDP, ICMP only	Any Internet Protocol
Assigned at launch	Assigned at launch or when stopped
Modify anytime	Modify anytime

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### AWS EC2 Security Groups

Security Groups and network ACLs allow you to control inbound and outbound network access to and from your instances. If you do not have a default VPC you must create a VPC and launch instances into that VPC to leverage advanced networking features such as private subnets, outbound security group filtering, network ACLs, Dedicated Instances, and VPN connections.



## Virtual Private Cloud: VPC

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the Amazon Web Services (AWS) Cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

You can easily customize the network configuration for your Amazon VPC. For example, you can create a public-facing subnet for your web servers that has access to the Internet, and place your backend systems such as databases or application servers in a private-facing subnet with no Internet access. You can leverage multiple layers of security, including security groups and network access control lists, to help control access to Amazon EC2 instances in each subnet.

Additionally, you can create a Hardware Virtual Private Network (VPN) connection between your corporate datacenter and your VPC and leverage the AWS cloud as an extension of your corporate datacenter.

## Infrastructure and Applications

- Oracle
- SAP
- Microsoft
- AMI Market Place

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Infrastructure and Applications

Amazon and Oracle have worked together to offer customers convenient options for deploying enterprise applications on the cloud. Customers can not only build enterprise-grade solutions hosted by Amazon Web Services (AWS) using database and middleware software by Oracle, but they can also launch entire enterprise software stacks from Oracle on EC2.

New and existing SAP customers can deploy their SAP solutions on SAP certified Amazon EC2 instances in production environments knowing that SAP and AWS have tested the performance of the underlying AWS resources, verified their performance, and certified them against the same standards that apply to servers and virtual platforms.

Amazon Web Services (AWS) provides infrastructure services that allow customers to easily run Microsoft Windows Server applications in the cloud, without the cost and complexity of having to purchase or manage servers or data centers. AWS provides pre-configured virtual machines, which enable customers to start running fully supported Windows Server virtual machine instances in minutes. Customers may also rely on the global infrastructure of AWS to power everything from custom .NET applications to enterprise deployments of Microsoft Exchange Server, SQL Server or SharePoint Server.



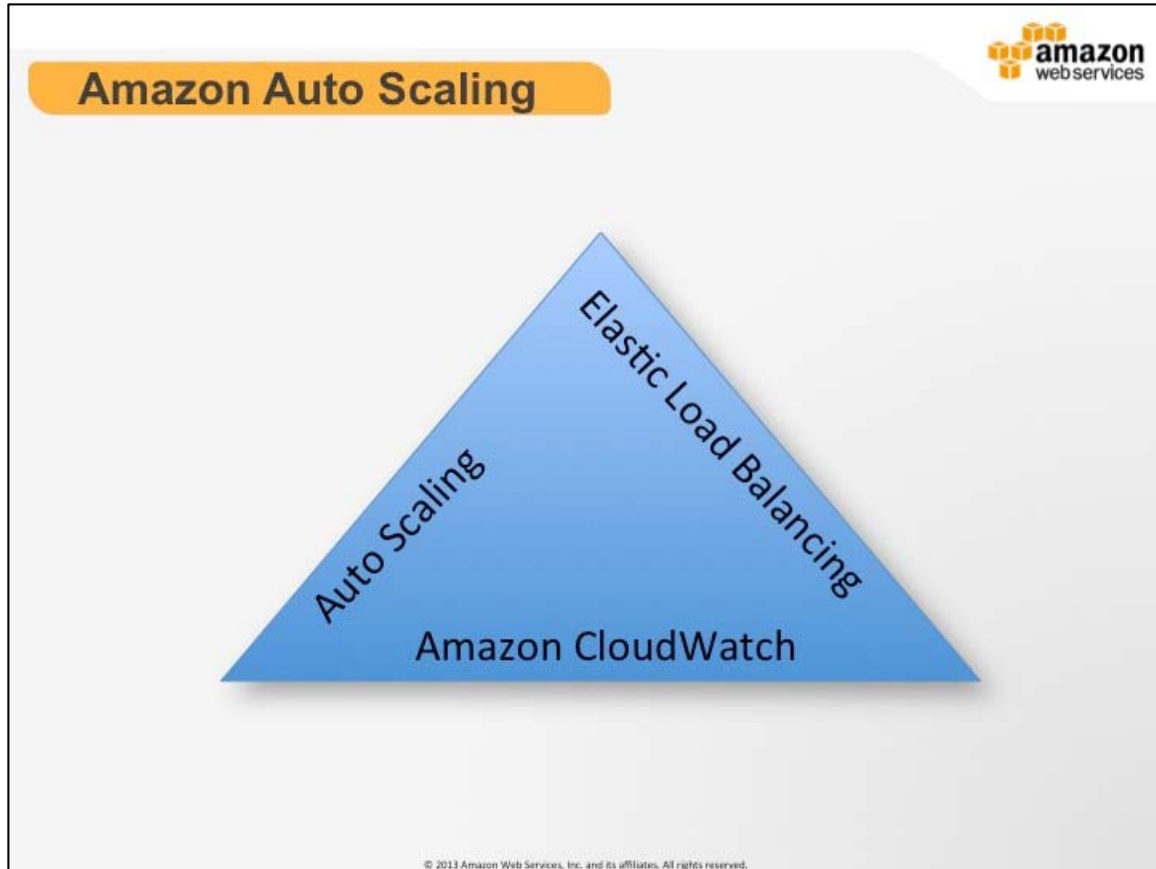
## Elasticity & Scalability

- Amazon Auto Scaling allows you to automatically horizontally scale to accommodate load
- You define policies to scale out (add instances) or scale in (remove instances); AWS takes care of the rest
- Policies are triggered by Amazon CloudWatch alarms

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

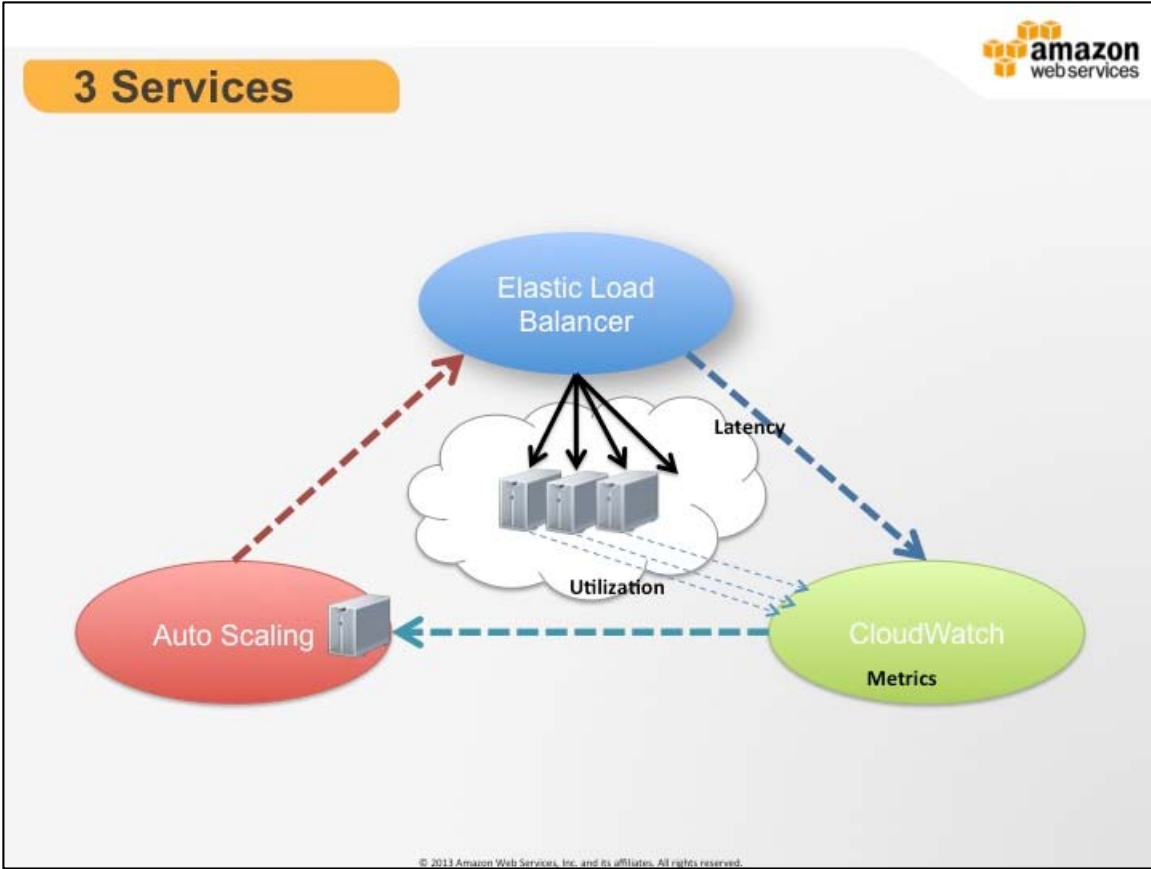
### **Elastic & Scalability**

Auto Scaling allows you to scale your Amazon EC2 capacity up or down automatically according to conditions you define. With Auto Scaling, you can ensure that the number of Amazon EC2 instances you're using increases seamlessly during demand spikes to maintain performance, and decreases automatically during demand lulls to minimize costs. Auto Scaling is particularly well suited for applications that experience hourly, daily, or weekly variability in usage. Auto Scaling is enabled by Amazon CloudWatch and available at no additional charge beyond Amazon CloudWatch fees.



### **Amazon Auto Scaling**

Auto Scaling works as a trinity of services working in concert. Elastic Load Balancers and EC2 instances feed metrics to CloudWatch. Auto Scaling defines a group with Launch Configurations and Auto Scaling Policies. Cloud Watch Alarms execute Auto Scaling policies to affect the size of your fleet.

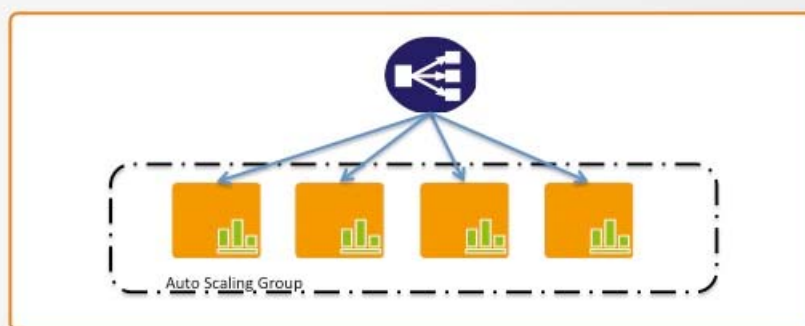


### 3 Services

All of these services work well individually, but together they become more powerful and increase the control and flexibility our customers demand.

## AWS Auto Scaling

- Elastic Capacity
- Ease of Use
- Cost Savings
- Geographic
- Geographic
- Easier Maintenance
- Scheduled Actions



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Amazon Auto Scaling

Let's say that you want to make sure that the number of healthy Amazon EC2 instances behind an Elastic Load Balancer is never fewer than two. You can use Auto Scaling to set this condition, and when Auto Scaling detects that this condition has been met, it automatically adds the requisite amount of Amazon EC2 instances to your Auto Scaling Group. Or, if you want to make sure that you add Amazon EC2 instances when latency of any one of your Amazon EC2 instances exceeds 4 seconds over any 15 minute period, you can set that condition, and Auto Scaling will take the appropriate action on your Amazon EC2 instances — even when running behind an Elastic Load Balancer. Auto Scaling works equally well for scaling Amazon EC2 instances whether you're using Elastic Load Balancing or not.

## How Can I Obtain Info about Myself from a Running Instance?

<http://169.254.169.254/latest/meta-data> contains a wealth of info

- 📁 ami-id
- 📁 ami-launch-index
- 📁 ami-manifest-path
- 📁 block-device-mapping/
- 📁 hostname
- 📁 instance-action
- 📁 instance-id
- 📁 Instance-type
- 📁 kernel-id
- 📁 local-hostname
- 📁 local-ipv4
- 📁 mac
- 📁 network/
- 📁 placement/
- 📁 profile
- 📁 public-hostname
- 📁 public-ipv4
- 📁 public-keys/
- 📁 reservation-id

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Metadata Service

Q: How can I obtain information about myself from a running instance?

# Elastic Load Balancing Exercise



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## Elastic Load Balancing Exercise #2

1. Security groups
2. Launch an instance.
3. Instance bootstrapping.
4. Machine images and elastic IPs.

## Working with EC2



### Exercise #2

- Security groups
- Instance launch
- Instance bootstrapping
- Machine images, and
- Elastic IPs

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.



### More Information

There are step-by-step directions at the end of this guide.

## Knowledge Check



Q: What is the AWS Instance that let's one pay a low, one-time fee and in turn receive a significant discount on the hourly charge for that instance?

Q: What kind of content can be stored in the cloud with Amazon S3?

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

1. \_\_\_\_\_  
\_\_\_\_\_
2. \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_



# Programmable Infrastructure

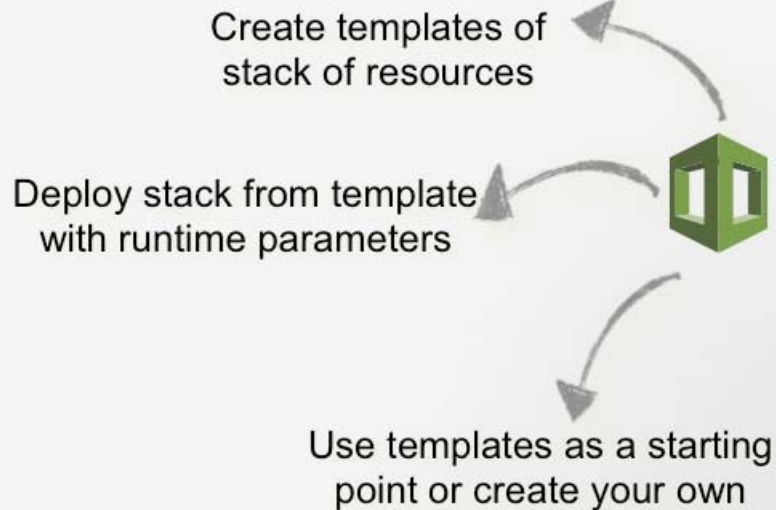


© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## **Programmable Infrastructure**

When you bring together all the tools and benefits of the Amazon Cloud, new strategies, paradigms, and processes become possible that simply were not possible with Traditional IT. One of the most significant among these is the idea of programmable infrastructure.

## AWS CloudFormation



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### AWS CloudFormation

AWS CloudFormation gives developers and systems administrators an easy way to create a collection of related AWS resources and provision them in an orderly and predictable fashion. The AWS CloudFormation samples package contains a collection of templates that illustrate various usage cases. Stacks can be created from the templates via the AWS Management Console, through the AWS CloudFormation command line tools or via the AWS CloudFormation APIs. (Demo CloudFormation in the AWS Management Console.)



#### More Information

Sample templates for CloudFormation:

<http://aws.amazon.com/cloudformation/aws-cloudformation-templates/>

## CloudFormation: Best Practice

- Programmable Infrastructure
- Check your templates into your favorite source control
- Enable New Change Management Controls and Operations Practices

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS CloudFormation: Best Practices**

Where previously, engineers would have diagrams of network layouts, and documents specifying firewall rules, with AWS all of that can be programmed into code and then checked into your favorite source control tool. This enables clear versioning and change tracking.

## Deployment & Management

- Templates are simple JSON formatted text files
- CloudFormer supports generating templates from running environments

```
"Resources" : {  
  "Ec2Instance" : {  
    "Type" : "AWS::EC2::Instance",  
    "Properties" : {  
      "SecurityGroups" : [ { "Ref" : "InstanceSecurityGroup" } ],  
      "ImageId" : { "Fn::FindInMap" : [ "RegionMap", { "Ref" : "AWS::Region" }, "AMI" ] },  
      "Tags" : [{  
        "Key" : "MyTag",  
        "Value" : "TagValue"  
      }]  
    }  
  },  
}
```

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Deployment & Management

AWS CloudFormation gives developers and systems administrators an easy way to create a collection of related AWS resources and provision them in an orderly and predictable fashion.

## AWS: API

Enables your applications running on Amazon EC2 to make requests to AWS services **without** you having to copy credentials onto every instance your application is deployed to.

- Temporary (STS) credentials pushed securely to Amazon EC2 instance via metadata
- Automatically rotated every minute (configurable)
- AWS SDKs do the work for you

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS Web Service API**

AWS has created an API that is wrapped into software development kits that come in a variety of languages and structures. All the SDKs are based on the same API and expose native interfaces that allow you to leverage the expertise you already have.

## AWS: API

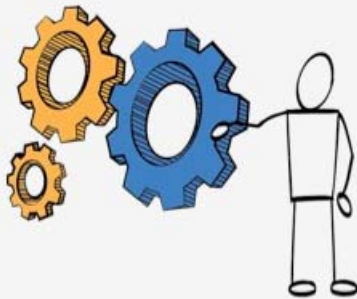
- AWS has wrappers for:
  - Android
  - iOS
  - Java
  - .NET
  - Node.js
  - Python
  - PHP
  - Ruby

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS Web Service API**

Most of the time, whatever API wrapper you use will dictate the interface.

# Managed Services



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## Managed Services

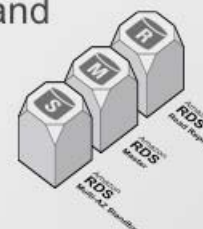
AWS offers many services that shift burden of undifferentiated heavy lifting off the customer by leveraging Amazon engineering and automation expertise.

For example, Amazon S3 is automatically run across every availability zone in the region. We do that for you so that you don't have to worry about it. As a result you get eleven nines of durability.

Although we have many managed services we are only going to dive into RDS today.

## Amazon Relational Database Service: RDS

- Fully-managed, tuned MySQL, Oracle 11g, or MS SQL databases
- Cost-efficient and resizable capacity
- Manages time-consuming database admin tasks
- Code, applications, and tools you already use today work seamlessly
- Automatically patches the database software and backs up your database
- Flexible Licensing: BYOL or License Include



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

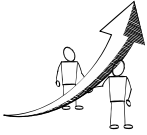
### Amazon Relational Database Service: RDS

Amazon RDS is a web service that makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while managing time-consuming database administration tasks, freeing you up to focus on your applications and business. Amazon RDS gives you access to the full capabilities of a familiar MySQL database. This means the code, applications, and tools you already use today with your existing MySQL databases work seamlessly with Amazon RDS. Amazon RDS automatically patches the database software and backs up your database, storing the backups for a user-defined retention period. You also benefit from the flexibility of being able to scale the compute resources or storage capacity associated with your relational database instance via a single API call.





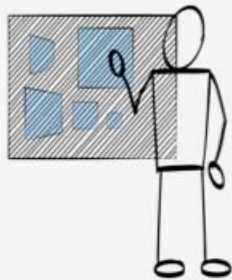
**Amazon Relational Database Service (RDS):** is a web service that makes it easy to set up, operate, and scale a relational database in the cloud.



#### **More Information**

- RDS: <http://aws.amazon.com/rds/>

# Database Services Demo

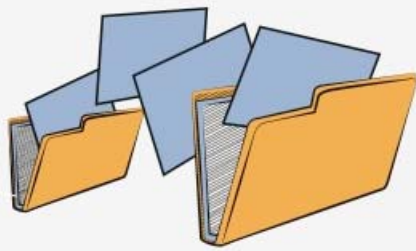


© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## Database Services Demo

1. Launch Multi-AZ cloud formation script from the cloud formation console (Word Press or Drupal)
2. Test drive the console
3. Answer questions
4. Be sure to hit the launch RDS dialog, talk about the different options

# Resources



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## **Resources**

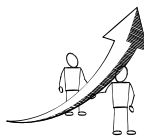
This section is a brief look at some of the resources available to you.

## AWS Online Resources



- Service Documentation
- Articles and Tutorials
- White Papers
- SDK & Sample Code Libraries
- Solutions Case Studies
- Marketing Overview Materials
- Videos & Webinars
- AWS Blog

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.



### More Information

- Service Documentation <http://aws.amazon.com/documentation>
- The Articles and Tutorials section features in-depth documents designed to give practical help to developers working with AWS. They have been created by members of the AWS developer community or the Amazon Team and give structured examples, analysis, tips, tricks and guidelines based on real usage of AWS services.
- White Papers <http://aws.amazon.com/whitepapers>: There are whitepapers authored by Amazon Team or AWS Community features a comprehensive list of technical AWS whitepapers, covering topics such as architecture, security and economics. This page will be regularly updated with new and updated whitepapers. Check back regularly or subscribe to the RSS feed for new content.
- Solutions Case Studies <http://aws.amazon.com/solutions/case-studies>: The Sample Code & Libraries Catalog provides a listing of code, SDKs, sample applications, and other tools available to for use by the AWS developer community.

- Marketing Overview Materials <http://aws.amazon.com>: AWS Case Studies and customer success stories.
- Videos & Webinars <http://www.youtube.com/AmazonWebServices>: View previously recorded webinars and videos about products, architecture, security and more. Check back regularly for new content from AWS, our customers and partners. There are also AWS Videos on YouTube.
- AWS Blog <http://aws.typepad.com/>

## AWS User Groups



AWS User Groups allow existing and new AWS users to join a growing, dynamic community, and interact with other developers to answer questions, share ideas, and learn about new services and best practices.

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **AWS User Groups**

With the rapid growth of AWS among individual developers, start-ups, and enterprises around the world, people are asking for AWS user groups. User groups allow existing and new AWS users to join a growing, dynamic community, and interact with other users to answer questions, share ideas, and learn about new services and best practices.



#### **More Information**

- If you are a passionate AWS user and are interested in joining or starting your own AWS User Group, see the list of existing groups online or review the tips on starting your own group:

<http://aws.amazon.com/usergroups/>

## Summary



- AWS History
- Cloud Computing
- Elastic Capacity
- Security and IAM
- Global Infrastructure
- AWS Services
- Additional Resources

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **Summary**

In summary here is what we covered in today's training.

## Next Steps



- AWS Courses
  - Architecting on AWS
  - Advanced System Operations
- AWS Certification
  - AWS Business Professional Courses
  - AWS Technical Professional Courses

© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### Next Steps

Your learning doesn't stop here. Continue your AWS education with additional courses or start the path to being certified.



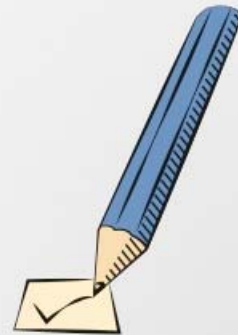
#### More Information

- AWS Training: <http://aws.amazon.com/aws-training/>



## Thank You

Your feedback is important to us!!!  
Please take a moment to fill out our evaluation so we can  
continue to enhance this training.



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

### **Thank You**

This concludes our training. Please take a moment to fill out our evaluation form so we can continue to enhance and revise our training to meet your needs. Your feedback is important to us and we really appreciate your comments.



For our exercises we will be using qwikLAB. qwikLAB will provide you an IAM User and Login Profile with which to access the AWS Management Console.

1. Navigate to <https://aws.qwiklab.com>
2. Register a new qwikLAB account if you don't have an existing qwikLAB account.
3. Login.

**Note:** If you forgot your password, retrieve it using the mechanism provided to send a reset link to your email.

4. Navigate to the class title "AWS Essentials" by clicking on it's title under "My Classes"
5. Click the button "Start Lab" next to the class lab title and number.

**Note:** Wait a moment after clicking "Start Lab" while the lab is starting and until you see the running lab details.

6. Click the 'Download PEM' button and save it to your Downloads directory or folder.
7. Select the contents of the 'Password' field and click the button "Enter AWS Console".
8. Enter 'awsstudent' into 'User Name' and paste the password you copied into 'Password' and click on the button "Sign in using our secure server".

Congratulations! You now have access to an AWS account and are logged into the AWS Management Console.

# S3 Exercise



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## S3 Exercise #1

### Getting Started with Amazon S3

1. Create a bucket in S3.
2. Add an object to the bucket.
3. View the object.
4. Move the object.
5. Delete the object and the bucket in S3.

### Create a Bucket in S3

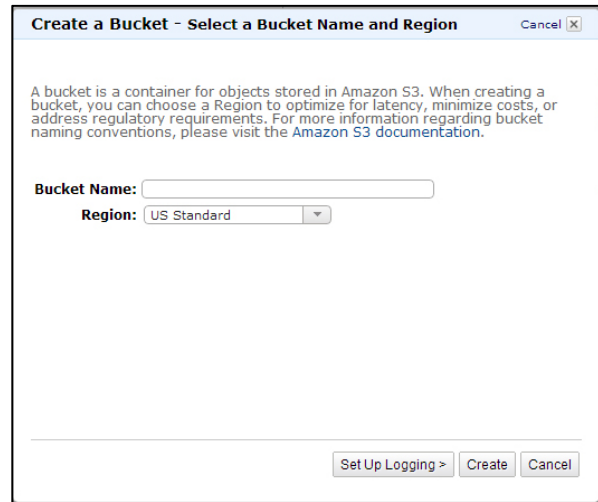
Every object in Amazon S3 is stored in a bucket. Before you can store data in Amazon S3 you must create a bucket.

**Note:** You are not charged for creating a bucket; you are only charged for storing objects in the bucket and for transferring objects in and out of the bucket.

1. Sign into the AWS Management Console and open the Amazon S3 console at <https://console.aws.amazon.com/s3>.

2. Click **Create Bucket**. The **Create a Bucket** dialog box appears.

3. Enter a bucket name in the **Bucket Name** field. The bucket name you choose must be unique across all existing bucket names in Amazon S3. One way to do that is to prefix your bucket names with your company's name.



Bucket names must comply with the following requirements.

Bucket names:

- Can contain lowercase letters, numbers, periods (.) and dashes (-)
- Must start with a number or letter
- Must be between 3 and 255 characters long
- Must not be formatted as an IP address (e.g., 265.255.5.4)

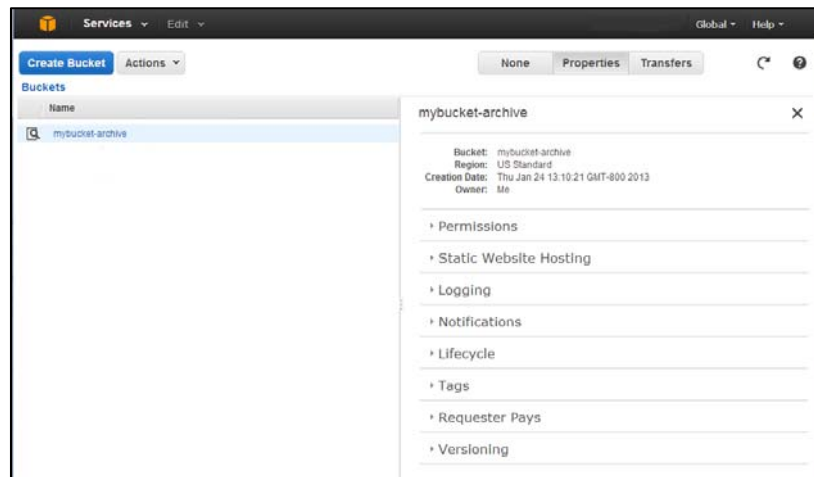
**Note:** There might be additional restrictions on bucket names based on the region your bucket is in or how you intend to access the object. Once you create a bucket, you cannot change its name. In addition, the bucket name is visible in the URL that points to the objects stored in the bucket. Make sure the bucket name you choose is appropriate.

4. In the **Region** drop-down list box, select a region.



Ask the Instructor what region to select.

5. Click **Create**. When Amazon S3 successfully creates your bucket, the console displays your empty bucket in the **Buckets** panel.



Great Job: You've created a bucket in Amazon S3!!!

### Add and Object to a Bucket:

Now that you've created a bucket, you're ready to add an object to it. An object can be any kind of file: a text file, a photo, a video and so forth. When you add a file to Amazon S3, you have the option of including metadata with the file and setting permissions to control access to the file.

6. In the Amazon S3 console click the bucket you want to upload an object into

and then click

**Upload** in the

**Objects and**

**Folders** panel.

The **Upload -**

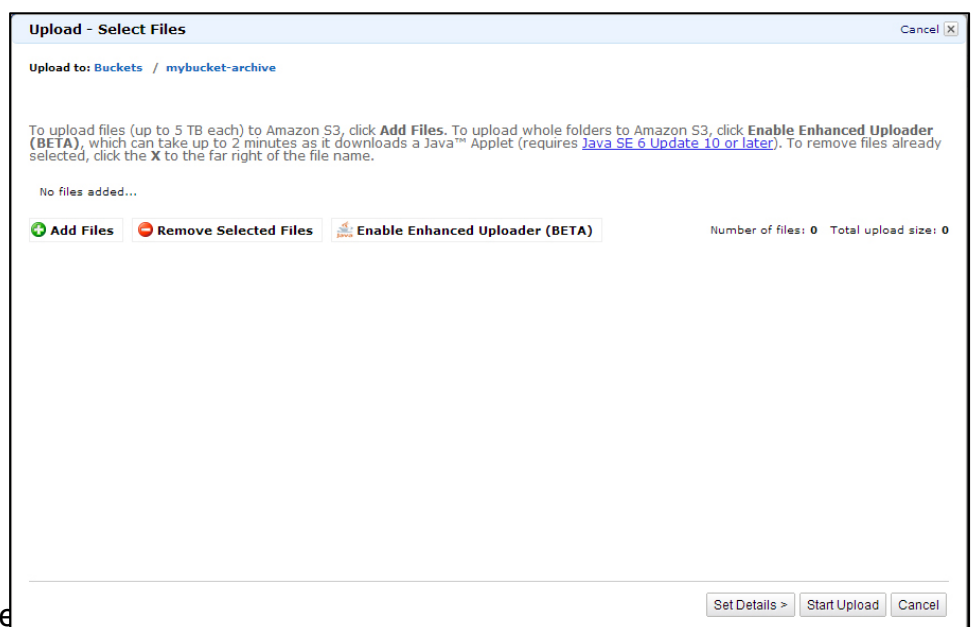
**Select Files**

wizard opens

(appearance may

differ slightly in

different



AWS Essentials Study

browsers):

7. If you want to upload a folder you must click **Enable Enhanced Uploader** for the Java applet. After you download the Java applet, the **Enable Enhanced Uploader** link disappears from the wizard. You only need to do this once per console session and you can transfer entire folders.

**Note:** If you are behind any corporate firewall you will need to install your corporate supported proxy client for the Java applet to work.

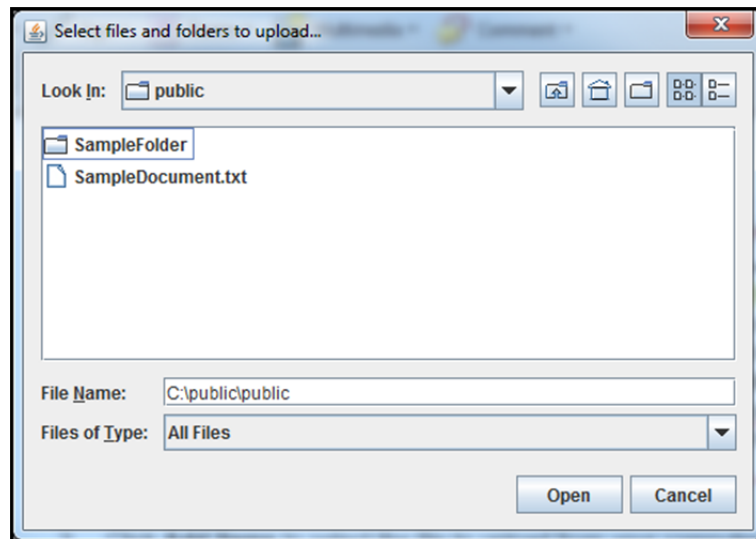
8. Click **Add Files** to select the file to upload from your computer. A file selection dialog box opens.



If you enabled advanced uploader in step 2, you see a Java file selection dialog box. If not, you see an operating system specific dialog box.

9. The following image shows a sample Java file selection dialog box.

10. Select a sample graphic that came with your computer to upload and click **Open**. The **Upload - Select Files** wizard shows the files and folders you've selected to upload.



11. Click **Start Upload**. You can watch the progress of the upload using the **Transfer** panel. The **Transfer** panel appears on the bottom of the screen as soon as you begin the upload.



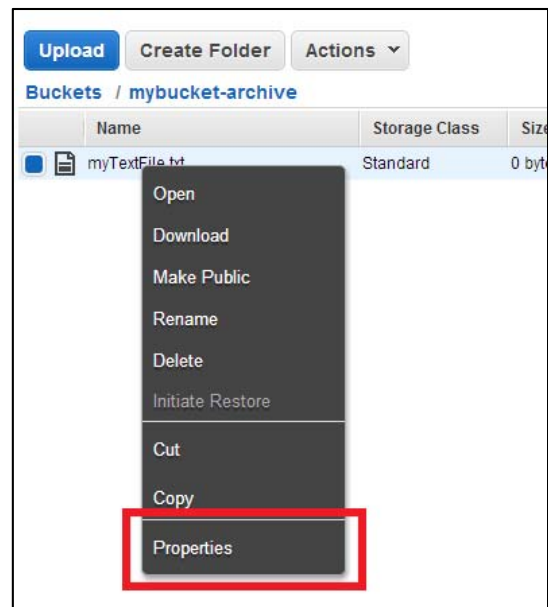
If you want to toggle between hiding and viewing the **Transfer** panel, click the **Transfers** button in the top right of the **Objects and Folders** panel. After the object uploads successfully to Amazon S3, it appears in the object listing.

Great Job: You've added a file to your bucket!!!

### View an Object:

12. Now that you've added an object to a bucket, you can open and view it in a browser. In the Amazon S3 console, right-click the objects you want to open.
13. Click **Properties** to browse the URL for the item you added.

**Note:** By default your Amazon S3 buckets and objects are private. To view object using a URL, for example, <https://s3.amazonaws.com/Bucket/Object> the object must be publicly readable. Otherwise, you will need to create signed URL that includes a signature with authentication information. You can optionally save the object locally.



Great Job: You've opened your object!!!

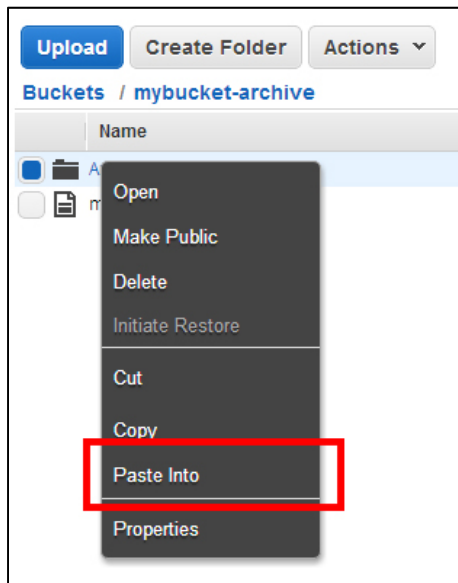
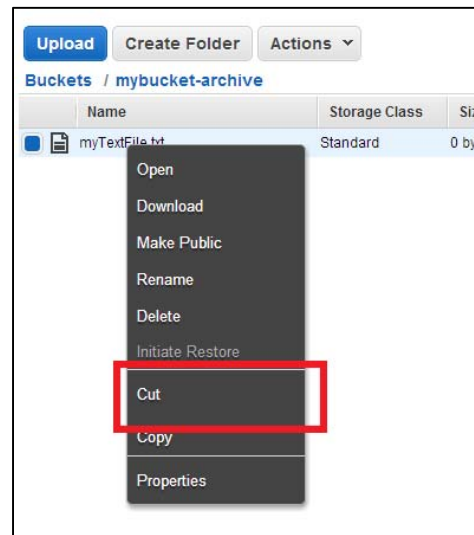
### Move an Object:

Now that you've added an object to a bucket and viewed it, you might like to move the object to a different bucket or folder.

14. In the [Amazon S3 console](#), right-click the object you want to move. **Tip** You can use the **SHIFT** and **CRTL** keys to select multiple objects and perform the same action on them simultaneously.

15. Click **Cut**.

16. Navigate to the bucket (and folder) you want to move the object to, and right-click the folder or bucket you want to move the object to.



17. Click **Paste Into**. Amazon S3 moves your files to the new location.



You can monitor the progress of the move on the **Transfers** panel. To hide or show the **Transfer** panel, click the **Transfers** button at the top right of the console page.



Great Job: Your file has been moved!!!

### Delete an Object and Bucket:

You've viewed the object. Now, you can delete it and the bucket it's in.

If you no longer need to store the objects you uploaded and moved while going through this guide, you should delete them so you do not incur further charges on those objects.

18. In the [Amazon S3 console](#), right-click on the object you want to delete. A dialog box shows the actions you can take on the selected object(s).



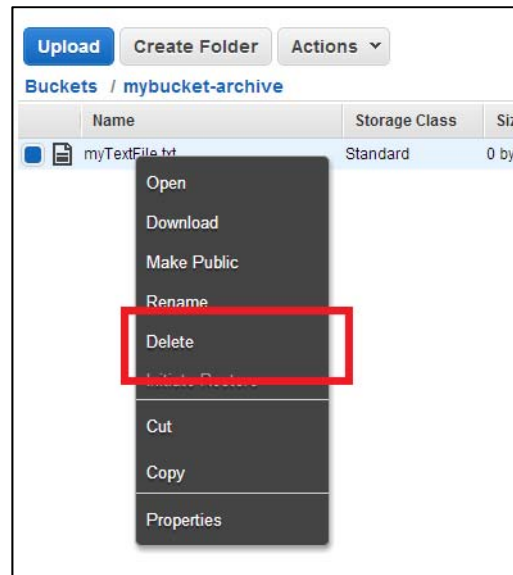
You can use the **SHIFT** and **CRTL** keys to select multiple objects and perform the same action on them simultaneously.

19. Click **Delete**.

20. Confirm the deletion when the console prompts you to.

21. Right-click the bucket you want to delete. A dialog box shows the actions you can take on the selected bucket.

22. Click **Delete**. Confirm the deletion when the console prompts you to.



To delete a bucket, you must first delete all of the objects in it. If you haven't deleted all of the objects in your bucket, do that now.

Great Job: Your bucket is deleted!!!

# Elastic Load Balancing Exercise



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.

## Elastic Load Balancing Exercise #2

1. Create an ELB across several EC2 instances
2. Deploy a simple application on EC2
3. Then distribute load by viewing the application

This lab will walk you through the process of creating an Elastic Load Balancer (ELB) to load balance traffic across several EC2 instances in a single Availability Zone. You will deploy a simple application on EC2 instances over which you will distribute load by viewing the application in your browser.

During this lab you will achieve the following:

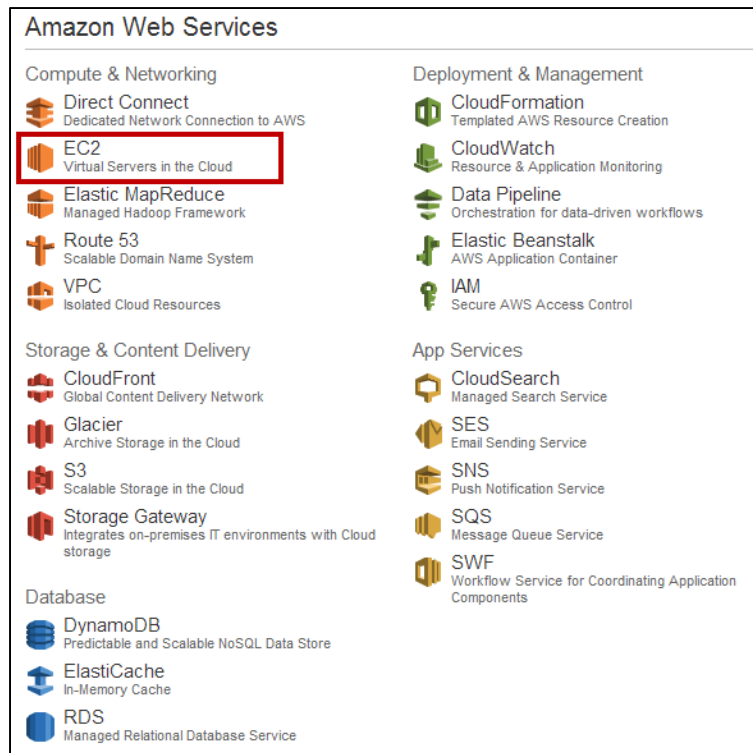
- Launch a multiple server web server farm on EC2 using bootstrapping techniques to configure a Linux server with Apache, PHP and a simple PHP application downloaded from S3
- Create an Elastic Load Balancer (ELB) using the console to front your created EC2 instances

- Configure the ELB to front the web servers and distributed load
- Explore Cloud Watch metrics for the ELB and also the instances behind the ELB.

## Launch Web Servers

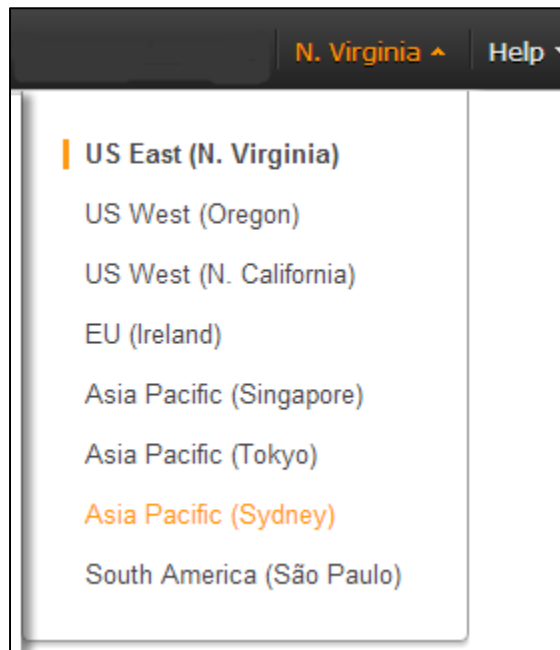
In this lab we will launch a 2 server Amazon Linux farm with an Apache PHP web server and basic application installed on initialization. This lab introduces you to a very basic example of bootstrapping your instances using the meta-data service to get you thinking about more complicated patterns that you might want to implement to configure instances as they are started.

1. To start your instances navigate to the EC2 page in the AWS Console.

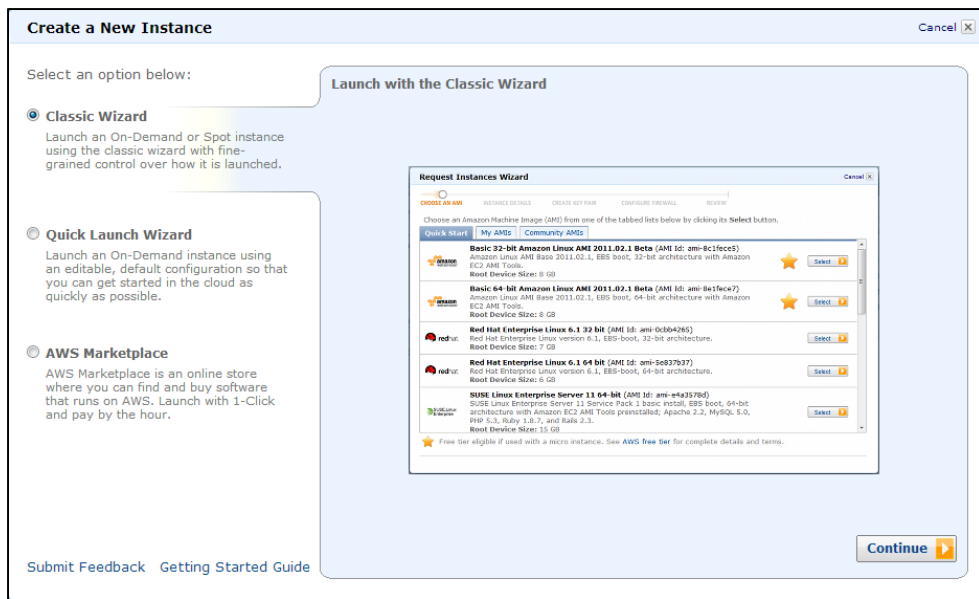




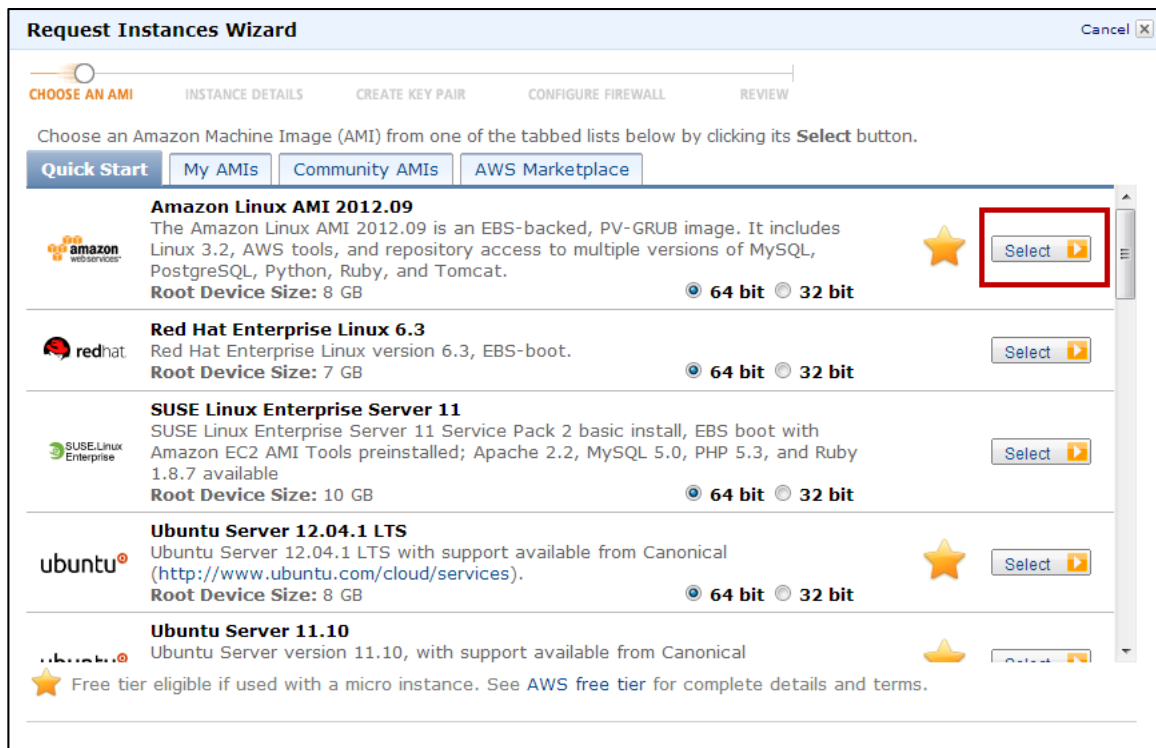
Please ensure your region is set to US East (N. Virginia). In the upper right corner of the page, you can choose the region using the selector in the console shown below:



2. Now click on **Launch Instance**.
3. Next select **Launch Classic Wizard** and click **Continue**. It is possible to start your instances using the QuickLaunch wizard, but for the purposes of this lab we want to see all the settings step by step, which is the way the Classic Wizard captures instance information.



- Now select the **Basic 64-bit Amazon Linux AMI**. This is a machine image from which our instances will be created. In this case this is an Amazon maintained Linux distribution with access to all the software repositories we require to install software for this lab (such as Apache and PHP).



5. We want to start more than one instance for this lab, so change the number of instances to **2** and click **Continue**.

The screenshot shows the 'Request Instances Wizard' window. The progress bar indicates the current step is 'INSTANCE DETAILS'. Below the progress bar, there is a text prompt: 'Provide the details for your instance(s). You may also decide whether you want to launch your instances as "on-demand" or "spot" instances.' The 'Number of Instances' field contains the value '2'. The 'Instance Type' dropdown menu is set to 'T1 Micro (t1.micro, 613 MiB)'. The 'Launch as an EBS-Optimized instance (additional charges apply):' checkbox is unchecked. Under the 'Launch into:' section, the 'EC2' radio button is selected, and the 'Availability Zone' dropdown is set to 'No Preference'. The 'Request Spot Instances' radio button is unselected.

6. In the next screen we will use the **User Data** field to bootstrap our instance, running a custom script to install the necessary packages (Apache and PHP) and sample code (PHP scripts) that we will use in this lab. User data provides a mechanism to pass information to the Amazon metadata service, which instances request information from at launch time. One property of the metadata service is that shell scripts passed in will be executed. In our case we will bootstrap using the script shown in the box below.

7. To do this, copy & paste the following initialization script (you can use Shift-Enter to get to a new line in the text box) into the **User Data** field and click **Continue**:

```
#!/bin/sh
curl -L http://bootstrapping-assets.s3.amazonaws.com/bootstrap-elb.sh | sh
```

8. As an FYI - the shell script at the url above downloads from S3 and then executes the following command:

```
yum -y install httpd php
chkconfig httpd on
/etc/init.d/httpd start
```

```
cd /var/www/html
wget http://bootstrapping-
assets.s3.amazonaws.com/examplefiles-elb.zip
unzip examplefiles-elb.zip
```

This downloads and installs various components, starts them, and installs our sample application.

The screenshot shows the 'Request Instances Wizard' window, specifically the 'INSTANCE DETAILS' step. The wizard has five steps: 'CHOOSE AN AMI', 'INSTANCE DETAILS', 'CREATE KEY PAIR', 'CONFIGURE FIREWALL', and 'REVIEW'. The 'INSTANCE DETAILS' step is currently active. The 'Number of Instances' is set to 2, and the 'Availability Zone' is 'No Preference'. Under 'Advanced Instance Options', the 'Kernel ID' and 'RAM Disk ID' are both set to 'Use Default'. The 'Monitoring' section has a checkbox for 'Enable CloudWatch detailed monitoring for this instance' which is unchecked. The 'User Data' section has a radio button for 'as text' selected, and the text area contains the command: `#!/bin/sh  
curl -L http://bootstrapping-assets.s3.amazonaws.com/bootstrap-elb.sh | sh`. There are also options for '(Use shift+enter to insert a newline)' and 'base64 encoded'. The 'Termination Protection' section has a checkbox for 'Prevention against accidental termination' which is unchecked. The 'Shutdown Behavior' is set to 'Stop'. The 'IAM Role' is set to 'None'.

9. Next you can **click continue** on this screen, but you will notice that should you require, you are able to edit the size of the root volume, plus add additional disks to the instance at boot time.

**Request Instances Wizard** Cancel

CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW

**Number of Instances:** 2  
**Availability Zone:** No Preference

**Storage Device Configuration**

Your instance will be launched with the following storage device settings. Edit these settings to add EBS volumes, instance store volumes, or edit the settings of the root volume.

Type	Device	Snapshot ID	Size	Volume Type	IOPS	Delete on Termination
Root	/dev/sda1	snap-96f7f4e2	8	standard		true

**0 EBS Volumes**  Edit

< Back
Continue

10. Give your new web servers a nice name like **Essentials Lab Servers** and click **Continue**.

**Request Instances Wizard** Cancel

CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW

Add tags to your instance to simplify the administration of your EC2 infrastructure. A form of metadata, tags consist of a case-sensitive key/value pair, are stored in the cloud and are private to your account. You can create user-friendly names that help you organize, search, and browse your resources. For example, you could define a tag with key = Name and value = Webserver. You can add up to 10 unique keys to each instance along with an optional value for each key. For more information, go to [Using Tags](#) in the *EC2 User Guide*.

Key (127 characters maximum)	Value (255 characters maximum)	Remove
Name	Essentials Lab Servers	

[Add another Tag.](#) (Maximum of 10)



11. For this lab, we will reuse the key pair we created in the earlier lab. You also have the option of creating a new key pair.

**Request Instances Wizard** Cancel X

CHOOSE AN AMI INSTANCE DETAILS **CREATE KEY PAIR** CONFIGURE FIREWALL REVIEW

Public/private key pairs allow you to securely connect to your instance after it launches. For Windows Server instances, a Key Pair is required to set and deliver a secure encrypted password. For Linux server instances, a key pair allows you to SSH into your instance.  
To create a key pair, enter a name and click **Create & Download Your Key Pair**. You will be prompted to save the private key to your computer. Note: You only need to generate a key pair once - not each time you want to deploy an Amazon EC2 instance.

**Choose from your existing Key Pairs**

Your existing Key Pairs\*: EssentialsLab

Create a new Key Pair

Proceed without a Key Pair

12. Create a security group, which will be your firewall rules. As we are building out a Web server, we can name this “Essentials Lab Web Tier”, and shown below you need to open ports 22 and 80. You need to add a rule for both ports. This enables SSH and HTTP traffic.

**Request Instances Wizard** Cancel X

CHOOSE AN AMI INSTANCE DETAILS CREATE KEY PAIR **CONFIGURE FIREWALL** REVIEW

Security groups determine whether a network port is open or blocked on your instances. You may use an existing security group, or we can help you create a new security group to allow access to your instances using the suggested ports below. Add additional ports now or update your security group anytime using the Security Groups page.

Choose one or more of your existing Security Groups

**Create a new Security Group**

**Group Name** Essentials Lab Web Tier

**Group Description** SSH and HTTP Access

**Inbound Rules**

Create a new rule: Custom TCP rule

Port range: (e.g., 80 or 49152-65535)

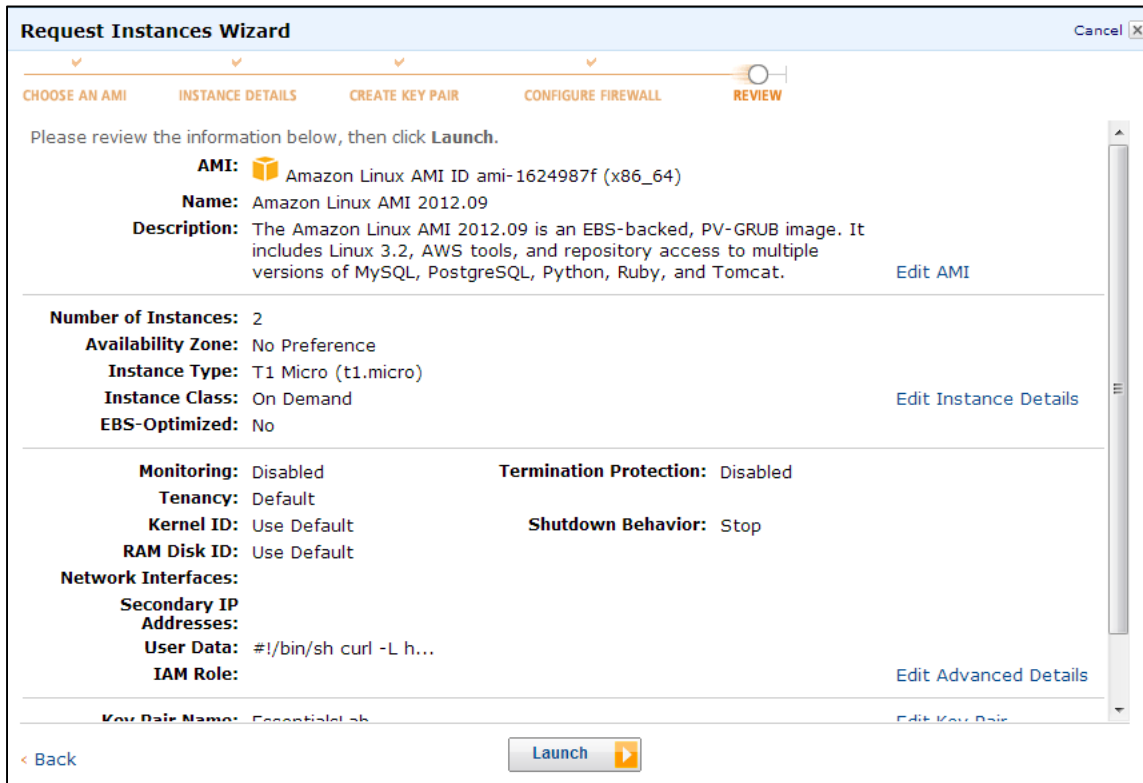
Source: 0.0.0.0/0 (e.g., 192.168.2.0/24, sg-47ad482e, or 1234567890/default)

**Add Rule**

TCP	Port (Service)	Source	Action
	22 (SSH)	0.0.0.0/0	Delete
	80 (HTTP)	0.0.0.0/0	Delete

< Back Continue >

13. Review your choices, and then click **Launch**. Your instances will now start.



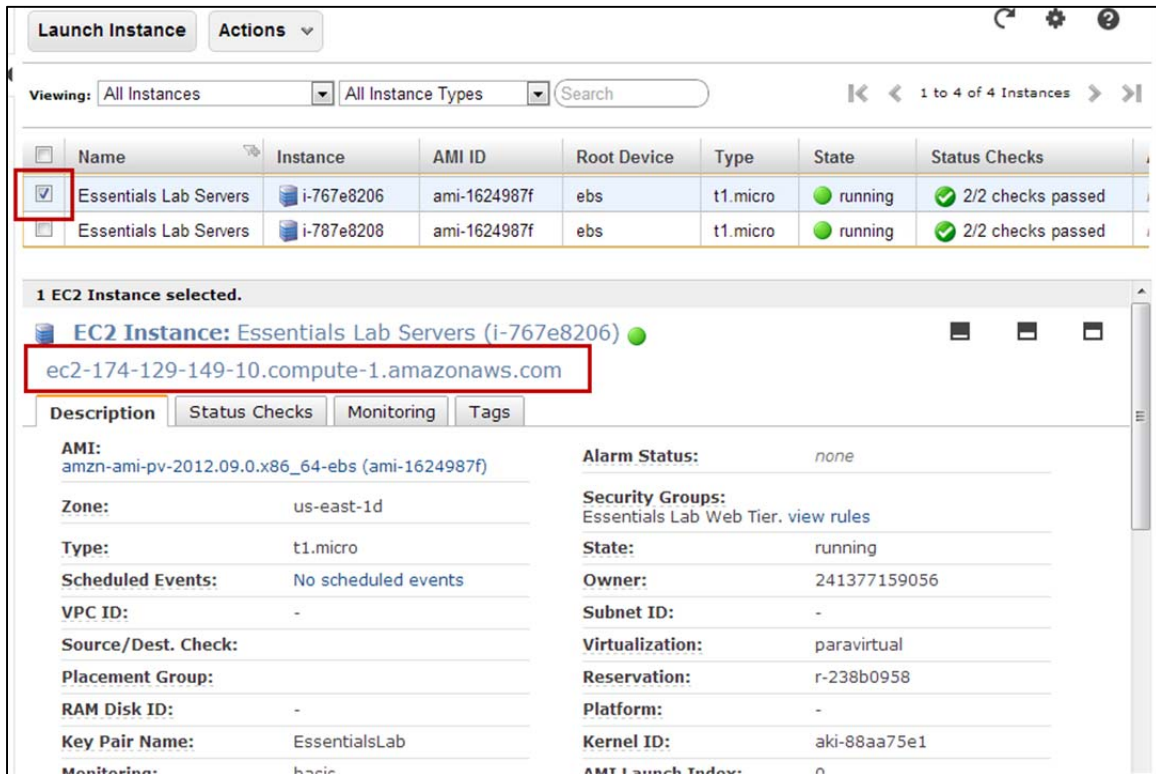
Great Job!!!

### Independently Connect to Each Web Server

14. First check the instances we started have finished their creation cycle by monitoring them to make certain they are running. You'll notice that the instances will be in a 'running' state with '2/2 checks passed'.

	Essentials Lab Servers	i-767e8206	ami-1624987f	ebs	t1.micro	running	2/2 checks passed
	Essentials Lab Servers	i-787e8208	ami-1624987f	ebs	t1.micro	running	2/2 checks passed

15. Now we can grab the public DNS entry allocated to each server so that we can use this to hit the server in our web browser. Click on the first Web Server, locate the server's DNS name, **select and copy** the server's name.



16. Paste the DNS name of each instance into another browser window or tab and you should see something like the following images:

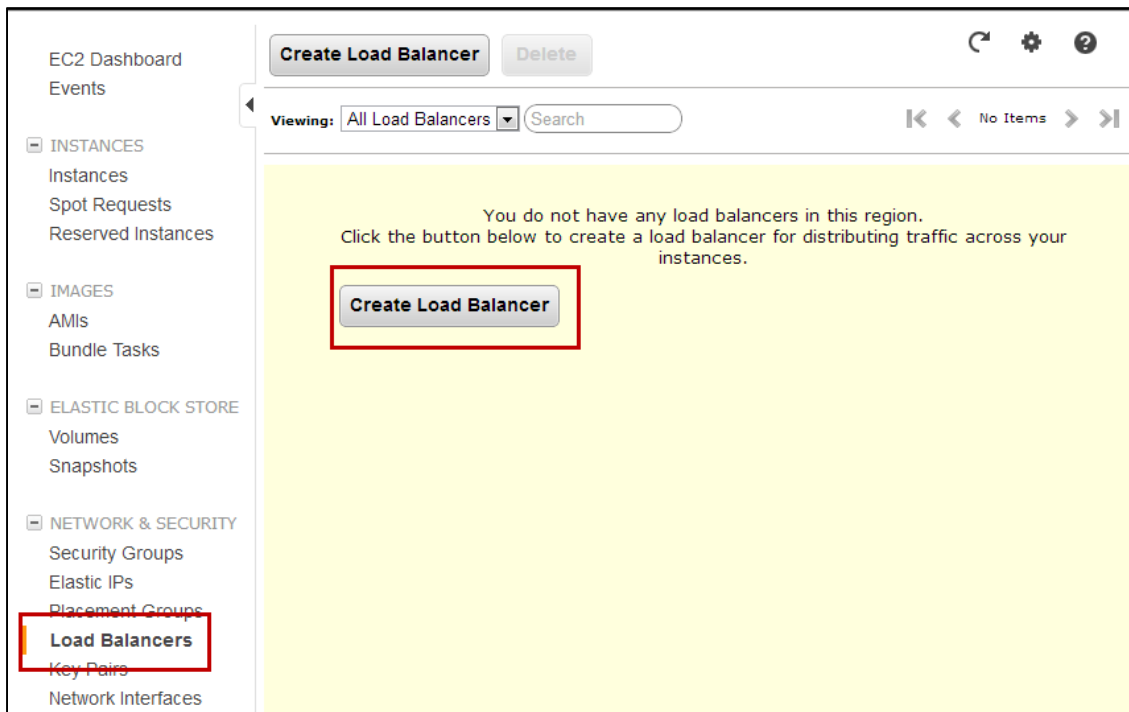


17. This is the web page returned by the PHP script that was installed when the instance when it started. It is a simple script that interrogates the metadata service and returns the instance ID and where it is running. This will be information that will help you see which instance you are hitting when we put an Elastic Load Balancer in front of them.

### Create an Elastic Load Balancer (ELB)

You now have two web servers, but you need a load balancer in front of these servers to give your users a single location for accessing both servers and to balance user requests across your simple web server farm.

18. In the **EC2 console**, click on the **Load Balancers** link, and click on **Create Load Balancer** button.



19. For this lab we will be creating a simple HTTP load balancer, so give your ELB a new name like **LabELB**, accept the default listener, and click **Continue**.

**Create a New Load Balancer** Cancel

---

DEFINE LOAD BALANCER   
 CONFIGURE HEALTH CHECK   
 ADD EC2 INSTANCES   
 REVIEW

This wizard will walk you through setting up a new load balancer. Begin by giving your new load balancer a unique name so that you can identify it from other load balancers you might create. You will also need to configure ports and protocols for your load balancer. Traffic from your clients can be routed from any load balancer port to any port on your EC2 instances. By default, we've configured your load balancer with a standard web server on port 80.

**Load Balancer Name:**

**Create LB inside:** EC2

**Create an internal load balancer:**   
(what's this?)

**Listener Configuration:**

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port	Actions
HTTP	80	HTTP	80	<input type="button" value="Remove"/>
<input type="text" value="HTTP"/>	<input type="text"/>	<input type="text" value="HTTP"/>	<input type="text"/>	<input type="button" value="Save"/>

20. On the next screen change **Ping Path** to **/ (delete index.html)** and change the **Healthy Threshold** to **3**. The ping path is the location on our web servers the ELB will check is returning a healthy response to keep instances in service or not. In our example / will return the default page – our PHP generated page seen earlier. The Healthy Threshold is the number of successful checks the ELB expects to see in a row before bringing an instance into service behind the ELB. We are lowering this to speed things up for our lab.

**Create a New Load Balancer** Cancel X

✔ DEFINE LOAD BALANCER   
 ○ CONFIGURE HEALTH CHECK   
 ADD EC2 INSTANCES   
 REVIEW

Your load balancer will automatically perform health checks on your EC2 instances and only route traffic to instances that pass the health check. If an instance fails the health check, it is automatically removed from the load balancer. Customize the health check to meet your specific needs.

**Configuration Options:**

Ping Protocol:

Ping Port:

Ping Path:

**Advanced Options:**

Response Timeout:  Seconds

Health Check Interval:  Minutes

Unhealthy Threshold:

Healthy Threshold:

Time to wait when receiving a response from the health check (2 sec - 60 sec).  
Amount of time between health checks (0.1 min - 5 min)  
Number of consecutive health check failures before declaring an EC2 instance unhealthy.  
Number of consecutive health check successes before declaring an EC2 instance healthy.

< Back                      Continue ▶

21. Click continue to accept the advanced options. Note that these options can be changed in the future, and configure how the ELB Health Check will be performed including the health check protocol, port, and path as well as the health check interval, timeout, and health thresholds.
22. Select your Web Servers to add them to your ELB and click **Continue**.

**Create a New Load Balancer** Cancel X

DEFINE LOAD BALANCER
CONFIGURE HEALTH CHECK
ADD EC2 INSTANCES
REVIEW

The table below lists all your running EC2 Instances that are not already behind another load balancer or part of an auto-scaling capacity group. Check the boxes in the Select column to add those instances to this load balancer.

**Manually Add Instances to Load Balancer:**

Select	Instance	Name	State	Security Groups
<input checked="" type="checkbox"/>	i-767e8206	Essentials Lab Servers	<span style="color: green;">●</span> running	Essentials Lab Web Tier
<input checked="" type="checkbox"/>	i-787e8208	Essentials Lab Servers	<span style="color: green;">●</span> running	Essentials Lab Web Tier

select all | select none

**Availability Zone Distribution:**

2 instances in us-east-1d  
0 instances in us-east-1c

< Back Continue ▶

23. Review your ELB settings and click **Create** (followed by **Close**).

24. AWS is now creating your ELB. It will take a couple of minutes to spin up your load balancers, attach your web servers, and pass the health checks.



**Click** on your load balancer, select the **Instances tab**, and wait until the instances status changes from Out of Service to **In Service**. Also note that the overall 'Healthy?' column turns from N0 to **Yes**. Your ELB is ready when this happens.



EC2 Dashboard  
Events

INSTANCES  
Instances  
Spot Requests  
Reserved Instances

IMAGES  
AMIs  
Bundle Tasks

ELASTIC BLOCK STORE  
Volumes  
Snapshots

NETWORK & SECURITY  
Security Groups  
Elastic IPs  
Placement Groups  
**Load Balancers**  
Key Pairs  
Network Interfaces

Create Load Balancer Delete

Viewing: All Load Balancers Search

Load Balancer Name	DNS Name	Port Configuration	Availability Zones
ELBLab	ELBLab-1563725764.us-east-1.elb.amazonaws.com	80 (HTTP) forwarding to 80 (HTTP)	us-east-1d

1 Load Balancer selected

Load Balancer: ELBLab

Description Instances Health Check Security Listeners

Instances

Instance	Name	Availability Zone	Status	Actions
i-767e8206	Essentials Lab Servers	us-east-1d	In Service	Remove from Load Balancer
i-787e8208	Essentials Lab Servers	us-east-1d	In Service	Remove from Load Balancer

Availability Zones

Availability Zone	Instance Count	Healthy?	Actions
us-east-1d	2	Yes	-

25. Once your ELB is healthy, click on the **Description tab**, select and copy the ELB's DNS name.

Load Balancer Name	DNS Name	Port Configuration
ELBLab	ELBLab-1563725764.us-east-1.elb.amazonaws.com	80 (HTTP) forwarding to 80 (HTTP)

1 Load Balancer selected

Load Balancer: ELBLab

Description Instances Health Check Security Listeners

**DNS Name:**  
ELBLab-1563725764.us-east-1.elb.amazonaws.com (A Record)  
ipv6.ELBLab-1563725764.us-east-1.elb.amazonaws.com (AAAA Record)  
dualstack.ELBLab-1563725764.us-east-1.elb.amazonaws.com (A or AAAA Record)

Note: Because the set of IP addresses associated with a LoadBalancer can change over time, you should never create an "A" record with any specific IP address. If you want to use a friendly



ELBs work across availability zones and they also scale elastically as demand dictates. They therefore do not have IP addresses but rather a URL to hit.

26. **Open** the **ELB URL** in another browser tab. Hit the browser refresh button a few times and you should cycle through your web servers such as the following images.



The URL in these browsers is the ELB address, not the instance addresses.

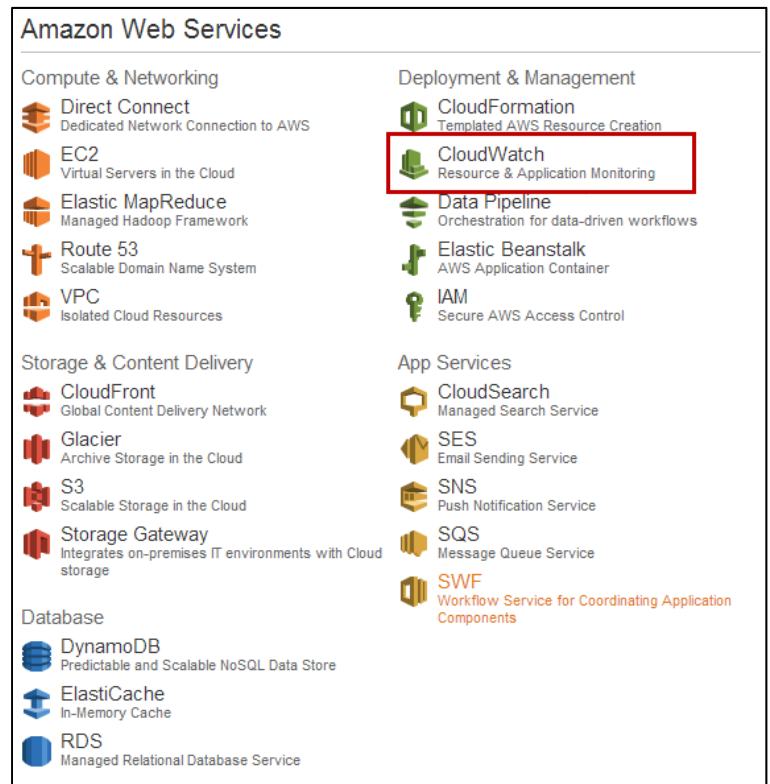


Great Job you created an ELB!!!

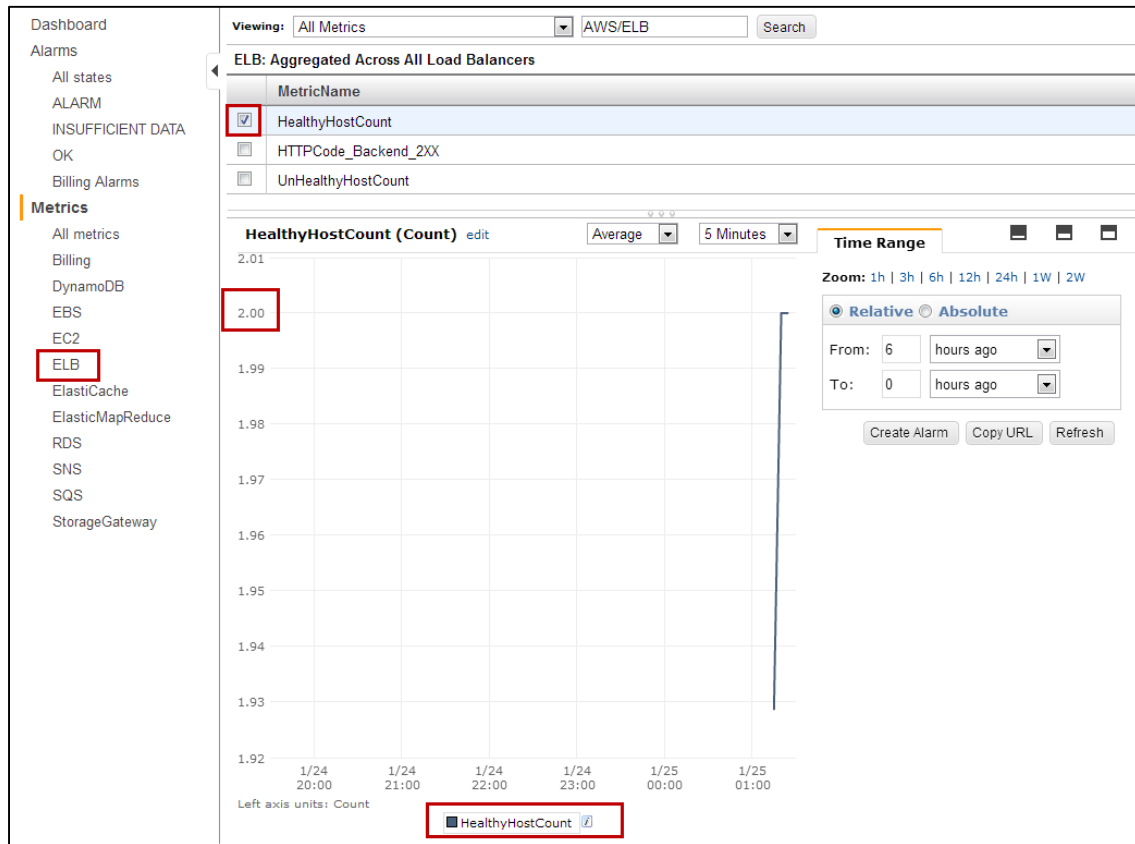
### View ELB CloudWatch Metrics

ELB automatically reports load balancer metrics to CloudWatch. You can view these metrics by clicking on the CloudWatch tab in the console.

In CloudWatch, click on the ELB link on the left, and select the metric you would like to view. ELB reports request latency, request count, healthy & unhealthy host counts, and a number of additional metrics. The metrics are reported as they are



encountered and can take several minutes to show up in CloudWatch. The following screenshot shows CloudWatch graphing the HealthyHostCount, which transitioned from zero healthy hosts to two shortly after the ELB was created for this lab.



Great Job!!! You have now been able to create an Elastic Load Balancer backed by multiple EC2 servers. In normal operation we would advise that these servers be located in separate availability zones to enable your application to be fault tolerant. You have also briefly seen how you can monitor the ELB metrics in Cloud Watch.

THIS PAGE INTENTIONALLY LEFT BLANK.



© 2013 Amazon Web Services, Inc. and its affiliates. All rights reserved.