



# DevOps

## AWS: A Quick Introduction

Trainer: Abhijith V G – AWS, eCommerce, Mobile & DevOps Architect



Certified

Developer - Associate

## Cloud Computing with Amazon Web Services



Amazon Web Services (AWS) is a secure cloud services platform, offering compute power, database storage, content delivery and other functionality to help businesses scale and grow. Millions of users are currently leveraging AWS cloud products and solutions to build sophisticated applications with increased flexibility, scalability and reliability.

# Introduction to AWS

What sets AWS apart? Building and managing cloud since 2006  
90+ services!

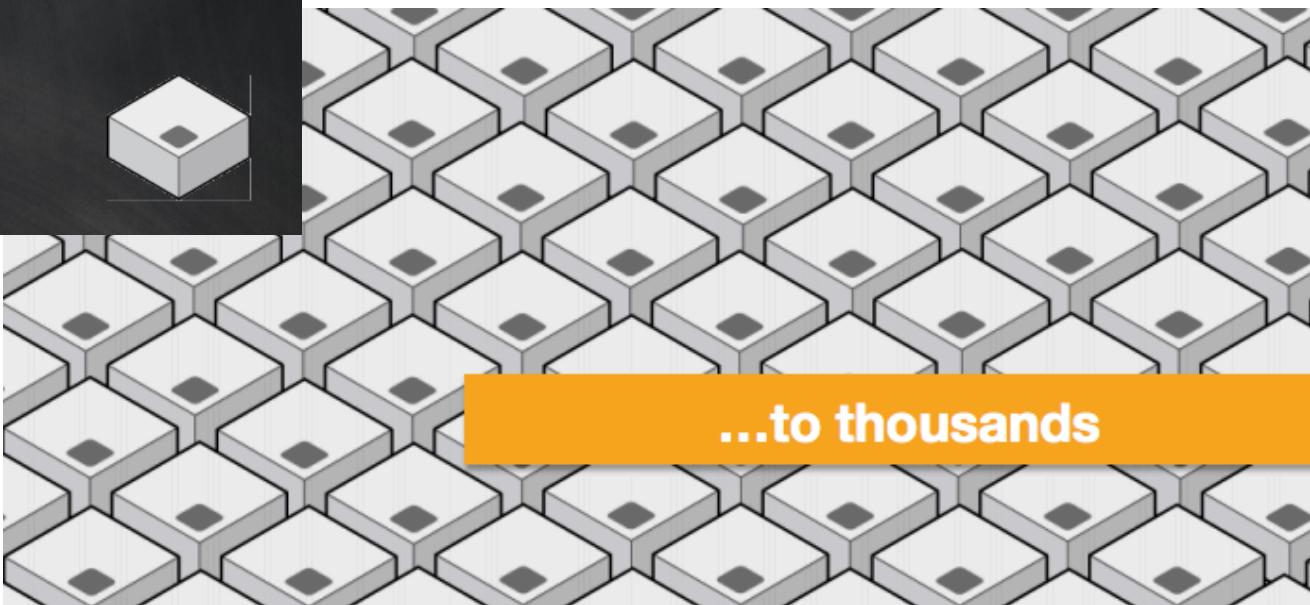


# Elastic Computing

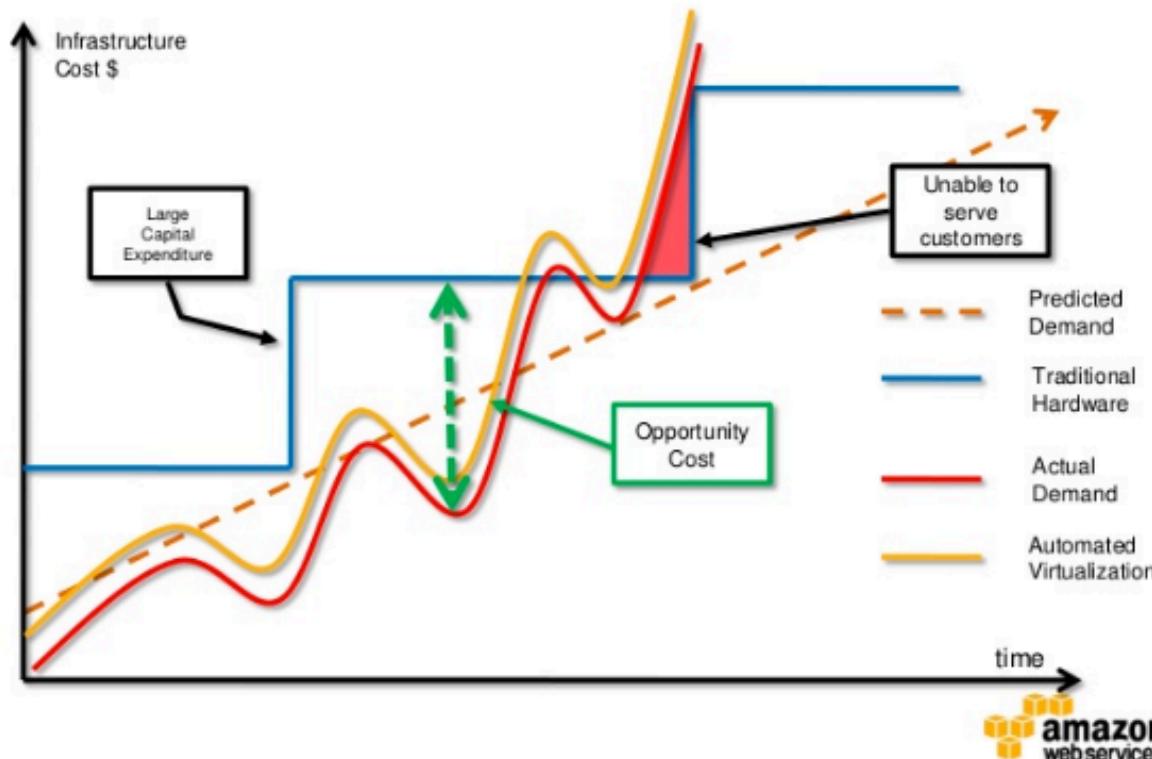
from one compute instance...



...to thousands



## Elastic and Pay-Per-Use Infrastructure



# Introduction to AWS Products: Management Tools

<https://aws.amazon.com/documentation/>

Visit the above link to see all the products/services of AWS

Go through this ever growing list!

<b>Compute</b> Amazon EC2 Amazon EC2 Container Registry Amazon EC2 Container Service Amazon Lightsail Amazon VPC AWS Batch AWS Elastic Beanstalk AWS Lambda Auto Scaling Elastic Load Balancing	<b>Developer Tools</b> AWS CodeStar AWS CodeCommit AWS CodeBuild AWS CodeDeploy AWS CodePipeline AWS X-Ray AWS Tools & SDKs	<b>Internet of Things</b> AWS Greengrass AWS IoT	<b>SDKs &amp; Toolkits</b> AWS SDK for C++ AWS SDK for Go AWS SDK for Java AWS SDK for JavaScript AWS SDK for .NET AWS SDK for PHP AWS SDK for Python (Beta) AWS SDK for Ruby AWS Toolkit for Eclipse AWS Toolkit for Visual Studio AWS Tools for Visual Studio Services
<b>Storage</b> Amazon S3 Amazon EBS Amazon EFS Amazon Glacier AWS Snowball AWS Storage Gateway	<b>Management Tools</b> Amazon CloudWatch AWS CloudFormation AWS CloudTrail AWS Config AWS OpsWorks AWS Service Catalog Trusted Advisor AWS Health AWS Management Console AWS Command Line Interface AWS Tools for Windows PowerShell	<b>Game Development</b> Amazon Lumberyard (beta) Amazon GameLift	<b>Mobile Services</b> AWS Mobile Hub Amazon API Gateway Amazon Cognito AWS Device Farm Amazon Mobile Analytics Amazon Pinpoint AWS Mobile SDK for Android AWS Mobile SDK for iOS AWS Mobile SDK for Unity AWS Mobile SDK for Xamarin Amazon SNS
<b>Database</b> Amazon RDS Amazon DynamoDB Amazon ElastiCache Amazon Redshift	<b>Security, Identity, &amp; Compliance</b> Identity & Access Management AWS Artifact AWS Certificate Manager AWS CloudHSM AWS Directory Service Amazon Inspector AWS KMS Amazon Macie AWS Organizations AWS Shield AWS WAF	<b>Application Services</b> Amazon API Gateway Amazon Elastic Transcoder Amazon SWF AWS Step Functions	<b>Additional Software &amp; Services</b> AWS Billing and Cost Management AWS Marketplace AWS Support Alexa Top Sites Alexa Web Information Service Amazon Silk AWS GovCloud (US)
<b>Networking &amp; Content Delivery</b> Amazon VPC Amazon CloudFront AWS Direct Connect Elastic Load Balancing Amazon Route 53		<b>Messaging</b> Amazon SNS Amazon SES Amazon SQS	<b>General Reference</b> Regions and Endpoints Security Credentials ARNs & Service Namespaces Service Limits AWS Glossary
<b>Migration</b> AWS Application Discovery Service AWS Database Migration Service AWS Import/Export AWS Migration Hub AWS Server Migration Service	<b>Analytics</b> Amazon Athena Amazon CloudSearch AWS Data Pipeline	<b>Business Productivity</b> Amazon WorkDocs Amazon WorkMail Amazon Chime	<b>AWS Management Console</b> Resource Groups Tag Editor Resource Groups Tagging
			<b>Resources</b> AWS Quick Starts AWS Whitepapers AWS Training & Certification

# Introduction to AWS Products: Compute

<https://aws.amazon.com/documentation/>

## Amazon EC2

Amazon EC2 Container Registry

Amazon EC2 Container Service

Amazon Lightsail

Amazon VPC

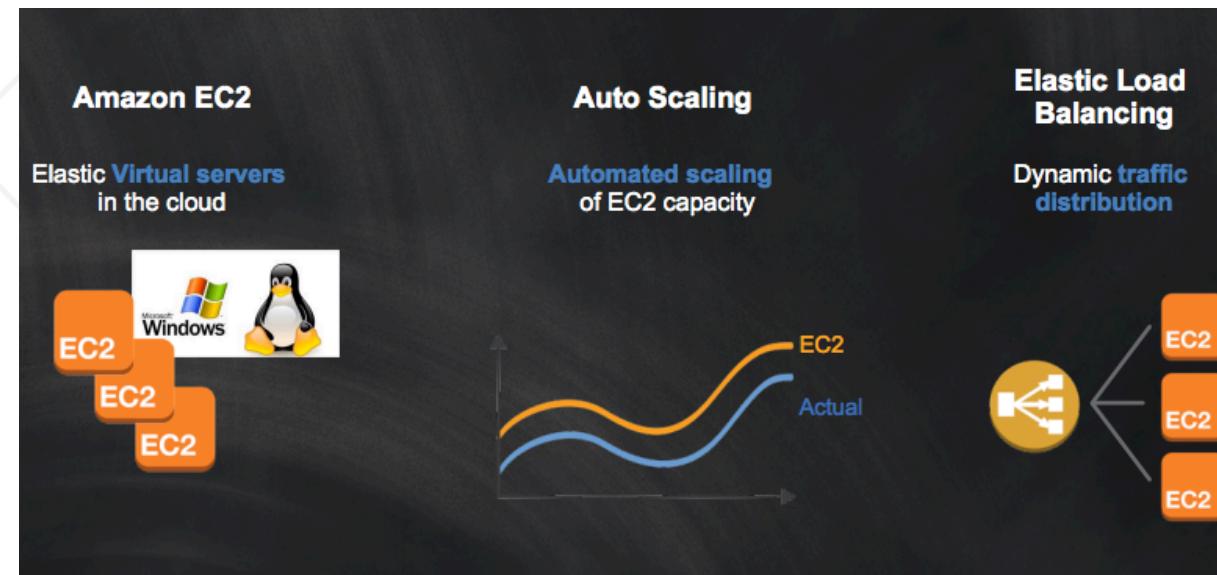
AWS Batch

AWS Elastic Beanstalk(PaaS)

AWS Lambda

Auto Scaling

Elastic Load Balancing



# Introduction to AWS Products: Storage

<https://aws.amazon.com/documentation/>

Amazon S3

Amazon EBS

Amazon EFS

Amazon Glacier

AWS Snowball

AWS Storage Gateway



# Introduction to AWS Products: Database

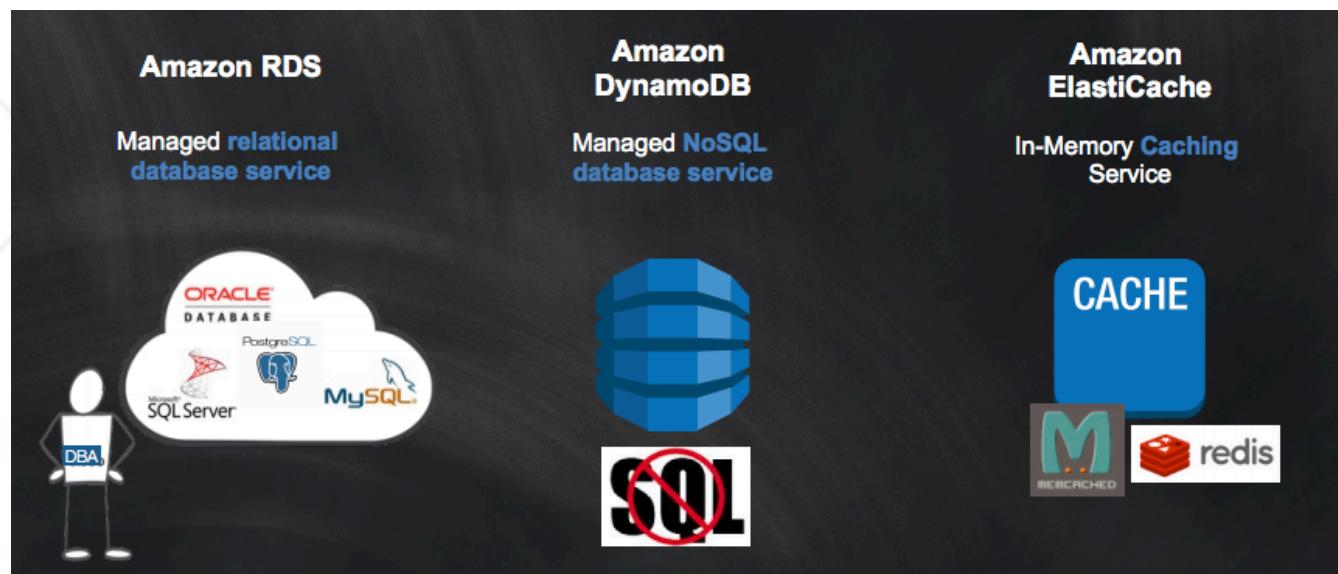
<https://aws.amazon.com/documentation/>

Amazon RDS

Amazon DynamoDB

Amazon ElastiCache

Amazon Redshift



# Introduction to AWS Products: Networking & Content Delivery

<https://aws.amazon.com/documentation/>

Amazon VPC

Amazon CloudFront (CDN)

AWS Direct Connect

Elastic Load Balancing

Amazon Route 53 (DNS)

# Introduction to AWS Products: Management Tools

<https://aws.amazon.com/documentation/>

Amazon CloudWatch

AWS CloudFormation (IaaS)

AWS CloudTrail

AWS Config

AWS OpsWorks

AWS Service Catalog

Trusted Advisor

AWS Health

AWS Management Console

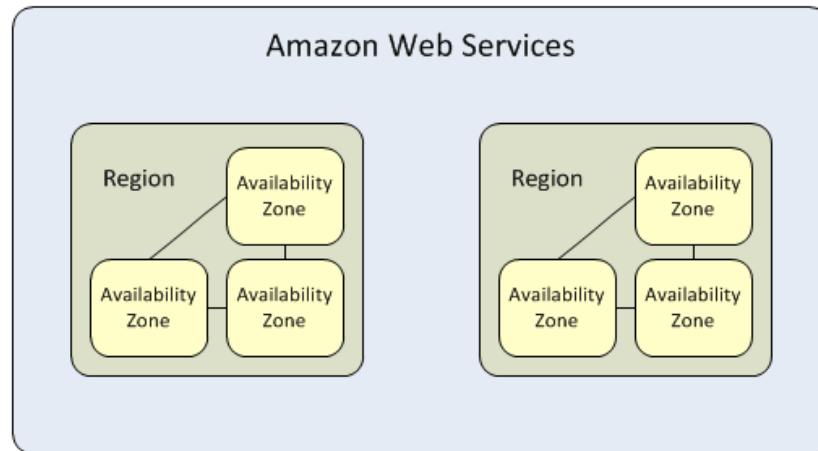
AWS Command Line Interface

AWS Tools for Windows PowerShell

# AWS Regions and Availability Zones

Amazon EC2 is hosted in multiple locations world-wide. These locations are composed of **Regions** and **Availability Zones**. Each region is a separate geographic area. Each region has multiple, **isolated** locations known as Availability Zones. Resources **aren't replicated** across regions unless you do so specifically.

Each region is completely independent. Each Availability Zone is isolated, but the Availability Zones in a region are connected through low-latency links. The following diagram illustrates the relationship between regions and Availability Zones.

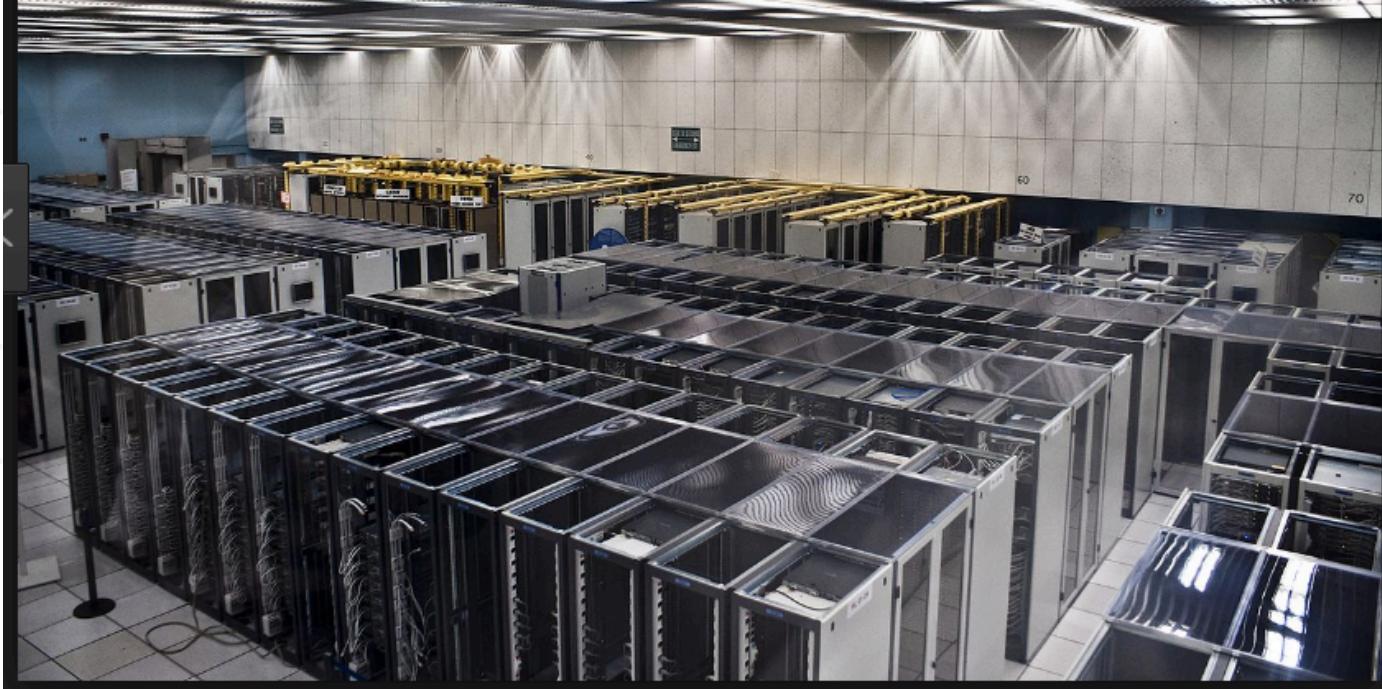


# AWS Regions and Availability Zones

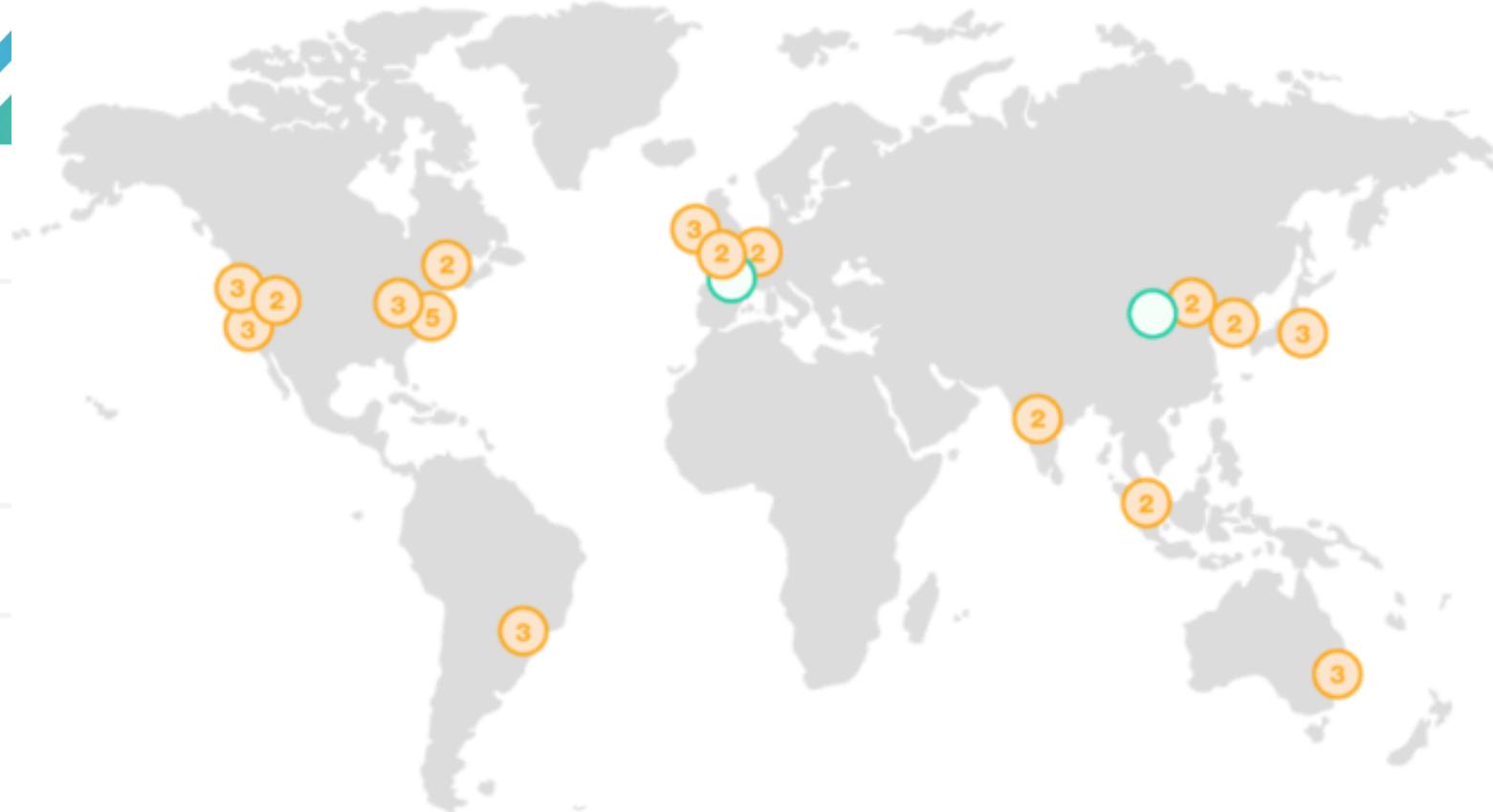
Some AZs have as many as 6 DCs (Data Centers)

All regions have a minimum of 2 or more EC2 AZs

A Single DC typically over 50,000 servers & often over 80,000



# AWS Regions and Availability Zones



# AWS Regions and Availability Zones



## Region & Number of Availability Zones

<b>US East</b>	<b>China</b>
N. Virginia (6), Ohio (3)	Beijing (2), Ningxia (2)
<b>US West</b>	<b>Europe</b>
N. California (3), Oregon (3)	Frankfurt (3), Ireland (3), London (3), Paris (3)
<b>Asia Pacific</b>	<b>South America</b>
Mumbai (2), Seoul (2), Singapore (2), Sydney (3), Tokyo (3)	São Paulo (3)
<b>Canada</b>	<b>AWS GovCloud (US-West) (2)</b>
Central (2)	



## New Region (coming soon)

Bahrain

Hong Kong SAR, China

Sweden

AWS GovCloud (US-East)

# AWS Regions and Availability Zones (Resource Locations)

<http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/resources.html>

Amazon EC2 resources are either global, tied to a region, or tied to an Availability Zone. For more information, see the above Resource location link.

## Resource Locations

Some resources can be used in all regions (global), and some resources are specific to the region or Availability Zone in which they reside.

Resource	Type	Description
AWS account	Global	You can use the same AWS account in all regions.
Key pairs	Global or Regional	<p>The key pairs that you create using Amazon EC2 are tied to the region where you created them. You can create your own RSA key pair and upload it to the region in which you want to use it; therefore, you can make your key pair globally available by uploading it to each region.</p> <p>For more information, see <a href="#">Amazon EC2 Key Pairs</a>.</p>
Amazon EC2 resource identifiers	Regional	Each resource identifier, such as an AMI ID, instance ID, EBS volume ID, or EBS snapshot ID, is tied to its region and can be used only in the region where you created the resource.

# AWS Regions and Availability Zones - Regions

Each Amazon EC2 region is designed to be completely isolated from the other Amazon EC2 regions. This achieves the greatest possible fault tolerance and stability. When you view your resources, you'll only see the resources tied to the region you've specified. When you launch an instance, you must select an AMI (Amazon Machine Image) that's in the same region, there is a charge for data transfer between regions.

## To find your regions and Availability Zones using the console

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/>
2. From the navigation bar, view the options in the region selector.



US East (N. Virginia)

**US East (Ohio)**

US West (N. California)

US West (Oregon)

Canada (Central)

EU (Ireland)

EU (Frankfurt)

EU (London)

Asia Pacific (Singapore)

Asia Pacific (Sydney)

Asia Pacific (Seoul)

Asia Pacific (Tokyo)

Asia Pacific (Mumbai)

South America (São Paulo)

# AWS Regions and Availability Zones - Availability Zones

When you launch an instance, you can select an Availability Zone or let Amazon choose one for you. If you distribute your instances across multiple Availability Zones and one instance fails, **you can design** your application so that an instance in another Availability Zone can handle requests. (aka Fault Tolerance)

An Availability Zone is represented by a region code followed by a letter identifier; for example, us-east-1a.

## To find your regions and Availability Zones using the console

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/>
2. From the navigation bar, view the options in the region selector.



US East (N. Virginia)

**US East (Ohio)**

US West (N. California)

US West (Oregon)

Canada (Central)

EU (Ireland)

EU (Frankfurt)

EU (London)

Asia Pacific (Singapore)

Asia Pacific (Sydney)

Asia Pacific (Seoul)

Asia Pacific (Tokyo)

Asia Pacific (Mumbai)

South America (São Paulo)

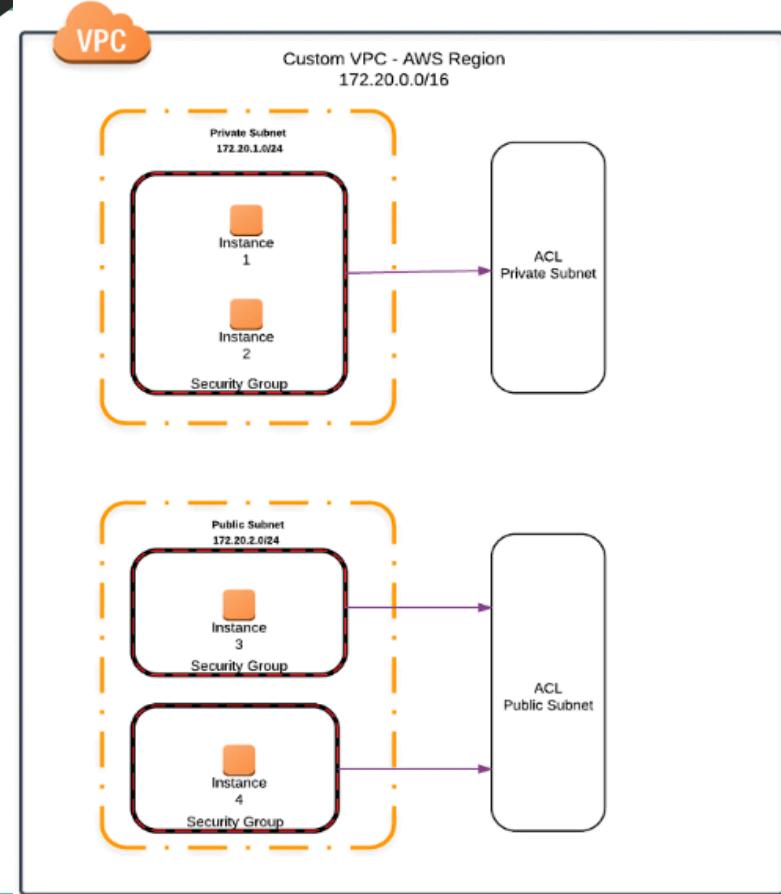


# VPC: What is VPC?

Amazon VPC is the **networking layer** for Amazon EC2.

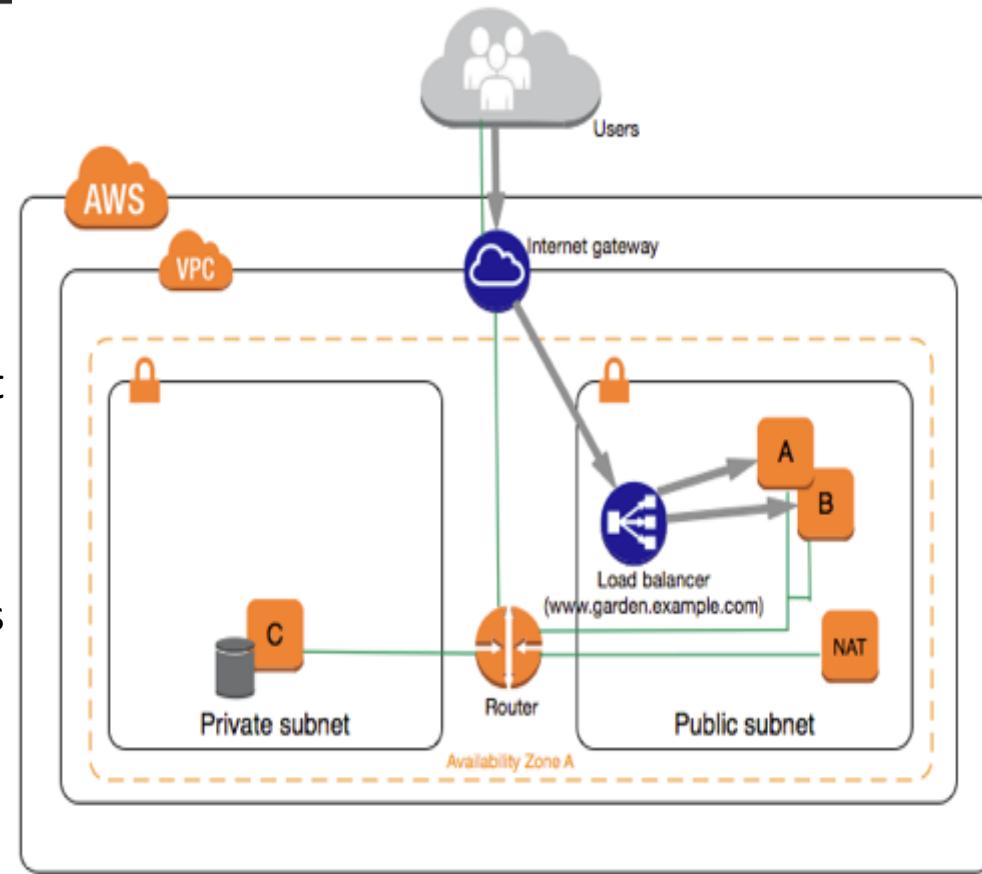
Amazon Virtual Private Cloud (Amazon VPC) enables you to launch Amazon Web Services (AWS) **resources** into a **virtual network** that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center/office, with the benefits of using the scalable infrastructure of AWS.

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a **logically isolated sections** of the Amazon Web Services (AWS) cloud.



# VPC: Allows Public and Private Subnets

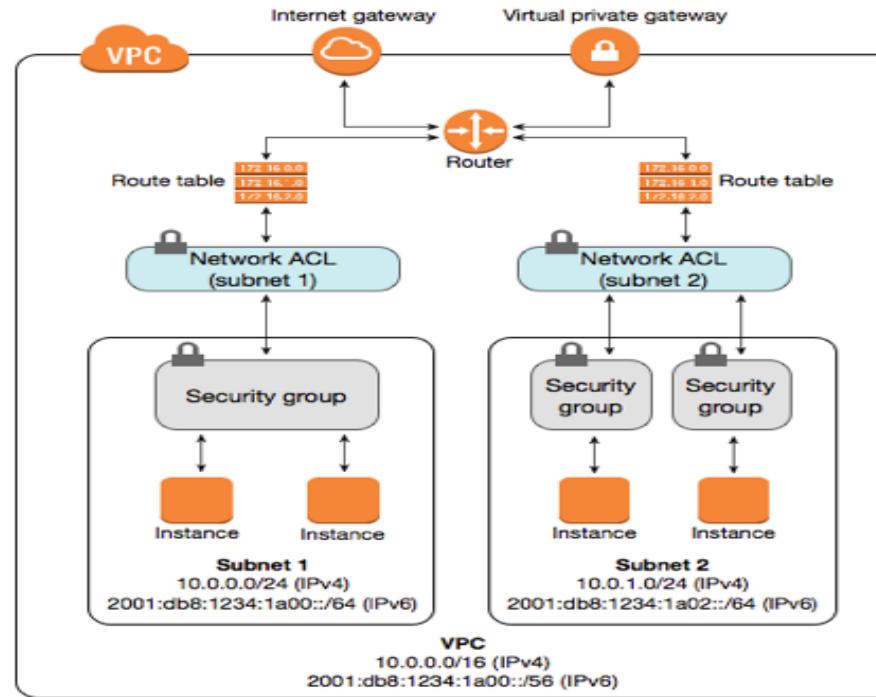
You can easily customize the network configuration for your Amazon Virtual Private Cloud. For example, you can create a public-facing subnet for your webservers that has access to the Internet, and place your backend systems such as databases or application servers in a private-facing subnet with no 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.



## VPC Peering (to interconnect many VPCs)

# VPC: Security (SG and N.ACL)

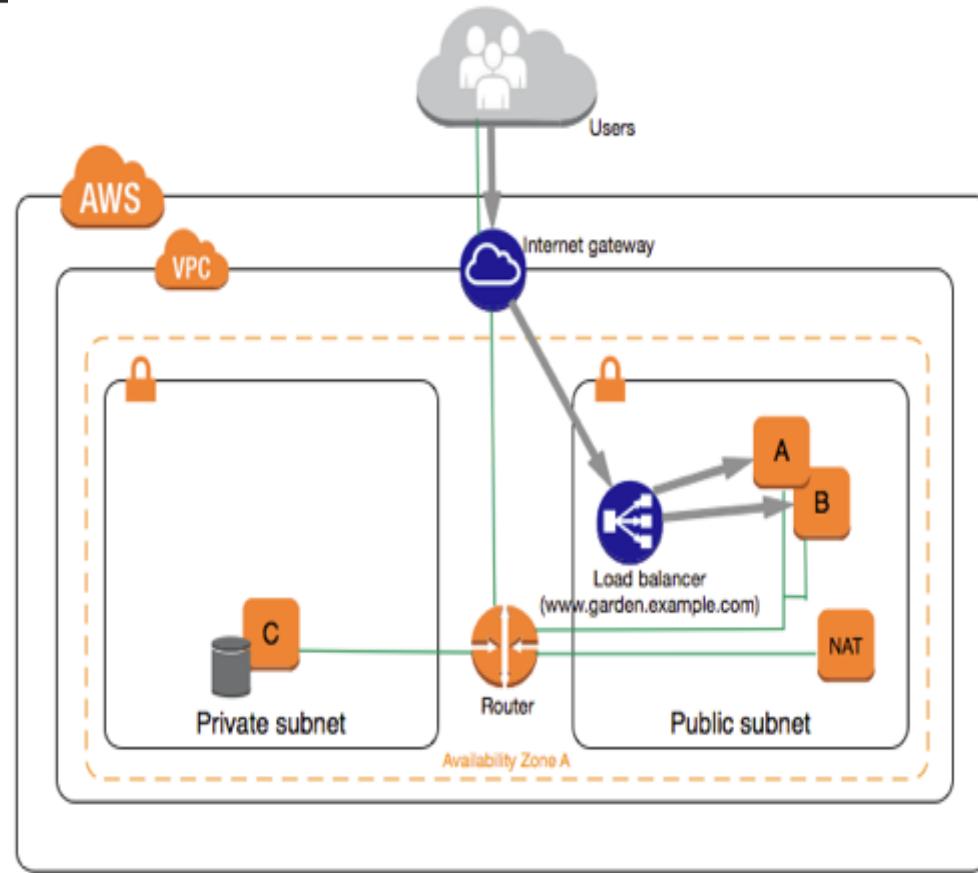
To protect the AWS resources in each subnet, you can use multiple layers of security, including security groups and network access control lists (ACL).



# VPC: Subnets

A subnet is a range of IP addresses in your VPC. You can launch AWS resources into a subnet that you select within the VPC.

Use a **public** subnet for resources that must be connected to the Internet, and a **private** subnet for resources that won't be connected to the Internet.



# Signing Up for AWS

## Creating an account

In order to use AWS for the first time, you will need to create an account. In order to create and run instances as described in this tutorial, you will need to associate a credit card with that account for billing purposes.

<https://console.aws.amazon.com>

AWS log in:

**Sign In or Create an AWS Account**

You may sign in using your existing Amazon.com account or you can create a new account by selecting "I am a new user."

My e-mail address is:

I am a new user.

I am a returning user  
and my password is:

**Sign in using our secure server** 

[Forgot your password?](#)

[Has your e-mail address changed?](#)

**New AWS Accounts Include:**

**12 months of access to the AWS Free Tier**

Amazon EC2: 750 hrs/month of Windows and Linux t2.micro instance usage  
Amazon S3: 5GBs of Storage  
Amazon RDS: 750 hrs/month of Micro DB Instance usage  
Amazon DynamoDB: 25 GB of storage, up to 200 million requests/month

**AWS Basic Support Features**

Customer Service: 24x7x365  
Support Forums  
Documentation, White Papers, and Best Practice Guides

Visit [aws.amazon.com/free](http://aws.amazon.com/free) for full offer terms.

## AWS Free Usage Tier

You can try some AWS services free of charge, within certain usage **limits**. AWS calls this the AWS Free Tier. The free tier is designed to give you hands-on experience with a range of AWS services at no charge.

When you create an AWS account, you are automatically signed up for the free tier for 12 months. Your free tier eligibility expires at the end of the 12-month period.

When your free tier expires, AWS starts charging the regular rates for any AWS services and resources that you are using.

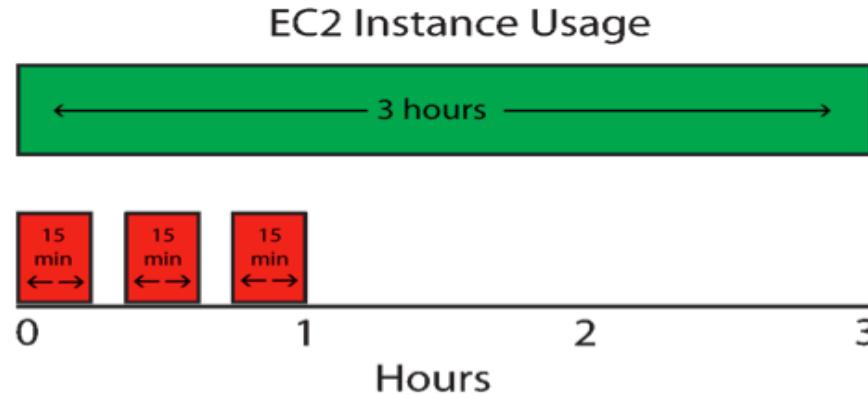
To help you stay within the limits, you can track your free tier usage and **set a billing alarm** to notify you if you start incurring charges.

# AWS Free Usage Tier

<http://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/free-tier-limits.html>

For example, you can use one Linux instance continuously for a month, or 10 Linux instances for 75 hours a month.

In some cases, leaving your resources running maximizes your free tier benefits. For example, if you run an Amazon EC2 instance for only a portion of an hour, AWS counts that as an entire hour. Therefore, if you stop and start an Amazon EC2 instance three times in a single hour, you use up three hours of your monthly allotment.



# AWS Free Usage Tier Eligibility

## Amazon Machine Images (AMI)

When you start an Amazon EC2 instance, you must select an Amazon Machine Image (AMI) that is eligible for the free tier. Because of licensing restrictions, some AMIs are not eligible for the free tier.

<https://console.aws.amazon.com/billing/home> (Track usage)

# AWS Free Usage Tier – Create a Billing Alarm

- 1) Enable billing alerts <https://console.aws.amazon.com/billing/home#/>

On the navigation pane, choose Preferences.

Select the **Receive Billing Alerts check box** (you cannot undo this).

Choose Save preferences.

- 2) Create a billing alarm

Open the CloudWatch console at <https://console.aws.amazon.com/cloudwatch/>.

(Remember the region on the navigation bar, billing data is stored in this region)

On the navigation pane, under Alarms, choose Billing.

Choose **Create Alarm**.

If you want the alarm to trigger as soon as you go over the free tier, set When my total AWS charges for the month exceed to \$.01. Otherwise, set it to the amount you want to trigger the alarm, and you will be notified when you go over that amount.

(Click the link in the confirmation email to confirm)

# EC2 - Launching your first AWS instance

The screenshot shows the AWS EC2 Dashboard. On the left sidebar, there's a navigation menu with sections like EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES (with sub-options Instances, Spot Requests, Reserved Instances, Dedicated Hosts), IMAGES (with sub-options AMIs, Bundle Tasks), ELASTIC BLOCK STORE (with sub-options Volumes, Snapshots), and NETWORK & SECURITY (with sub-options Security Groups, Elastic IPs, Placement Groups). The main content area has a heading 'Resources' and a message stating 'You are using the following Amazon EC2 resources in the Asia Pacific (Mumbai) region:' followed by a table of resource counts. Below this is a callout box with text about Amazon Lightsail. The right side features 'Account Attributes' with sections for Supported Platforms (VPC), Default VPC (vpc-10a96a79), Resource ID length management, Additional Information (Getting Started Guide, Documentation, All EC2 Resources, Forums, Pricing, Contact Us), and AWS Marketplace.

**Resources**

You are using the following Amazon EC2 resources in the Asia Pacific (Mumbai) region:

0 Running Instances	0 Elastic IPs
0 Dedicated Hosts	0 Snapshots
0 Volumes	0 Load Balancers
0 Key Pairs	1 Security Groups
0 Placement Groups	

Just need a simple virtual private server? Get everything you need to jumpstart your project - compute, storage, and networking – for a low, predictable price. Try [Amazon Lightsail for free](#).

**Create Instance**

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

**Launch Instance**

Note: Your instances will launch in the Asia Pacific (Mumbai) region

**Account Attributes**

Supported Platforms  
VPC  
Default VPC  
vpc-10a96a79  
Resource ID length management

**Additional Information**

Getting Started Guide  
Documentation  
All EC2 Resources  
Forums  
Pricing  
Contact Us

**AWS Marketplace**

Find free software trial products in the AWS

Feedback

English (US)

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

[Privacy Policy](#) [Terms of Use](#)

# EC2 - Launching your first AWS instance (EC2 - Choosing the AMI)

The screenshot shows the AWS EC2 console interface for launching a new instance. The top navigation bar includes the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, a bell icon for notifications, the user name 'Sreepakash. N', location 'Mumbai', and 'Support' dropdown.

The main content area displays a step-by-step wizard:

1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review

**Step 1: Choose an Amazon Machine Image (AMI)**

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

**Quick Start**

- My AMIs
- AWS Marketplace
- Community AMIs

Free tier only (i)

**1 to 29 of 29 AMIs**

Image	Name	Description	Root device type	Virtualization type	Action
	<b>Amazon Linux AMI 2017.03.1 (HVM), SSD Volume Type - ami-d7abd1b8</b>	The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.	ebs	hvm	<b>Select</b>
	<b>Red Hat Enterprise Linux 7.4 (HVM), SSD Volume Type - ami-e41b618b</b>	Red Hat Enterprise Linux version 7.4 (HVM), EBS General Purpose (SSD) Volume Type	ebs	hvm	<b>Select</b>
	<b>SUSE Linux Enterprise Server 12 SP3 (HVM), SSD Volume Type - ami-e310578c</b>	SUSE Linux Enterprise Server 12 Service Pack 3 (HVM), EBS General Purpose (SSD) Volume Type. Public Cloud, Advanced Systems Management, Web and Scripting, and Legacy modules enabled.	ebs	hvm	<b>Select</b>
	<b>Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-099fe766</b>	Ubuntu Server 16.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical ( <a href="http://www.ubuntu.com/cloud/services">http://www.ubuntu.com/cloud/services</a> ).	ebs	hvm	<b>Select</b>

# EC2 - Launching your first AWS instance

The screenshot shows the AWS EC2 Instance Creation Wizard at Step 2: Choose an Instance Type. The top navigation bar includes the AWS logo, Services, Resource Groups, a user profile for Sreepakash.N, and options for Mumbai and Support.

The main content area has a title "Step 2: Choose an Instance Type" and a descriptive paragraph about Amazon EC2 instance types. Below this is a table for filtering and viewing instance types.

**Filter by:** All instance types ▾ Current generation ▾ Show/Hide Columns

**Currently selected:** t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes
<input type="checkbox"/>	General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
<input type="checkbox"/>	General purpose	m4.large	2	8	EBS only	Yes	Moderate	Yes

Buttons at the bottom include: Cancel, Previous, Review and Launch, and Next: Configure Instance Details.

# EC2 - Launching your first AWS instance

AWS Services Resource Groups Sreepakash. N Mumbai Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

## Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances 1 [Launch into Auto Scaling Group](#)

Purchasing option  Request Spot instances

Network vpc-10a96a79 (default) [Create new VPC](#)

Subnet No preference (default subnet in any Availability Zone) [Create new subnet](#)

Auto-assign Public IP Use subnet setting (Enable)

IAM role None [Create new IAM role](#)

Shutdown behavior Stop

Enable termination protection  Protect against accidental termination

Monitoring  Enable CloudWatch detailed monitoring  
Additional charges apply.

Tenancy Shared - Run a shared hardware instance  
Additional charges will apply for dedicated tenancy.

Advanced Details

Cancel Previous [Review and Launch](#) Next: Add Storage

# EC2 - Launching your first AWS instance



Services ▾

Resource Groups ▾



Sreepakash. N ▾

Mumbai ▾

Support ▾

1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review

## Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-0187a530d8c47f3f4	8	General Purpose SSD	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#)[Previous](#)[Review and Launch](#)[Next: Add Tags](#)

# EC2 - Launching your first AWS instance

The screenshot shows the AWS EC2 instance creation wizard at Step 5: Add Tags. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, a bell icon for notifications, user name Sreeprakash.N, location Mumbai, and Support dropdown. Below the navigation is a horizontal progress bar with steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags (which is highlighted in orange), 6. Configure Security Group, and 7. Review.

## Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.  
A copy of a tag can be applied to volumes, instances or both.  
Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(127 characters maximum)	Value	(255 characters maximum)	Instances	Volumes
<i>This resource currently has no tags</i>					

Choose the Add tag button or [click to add a Name tag](#).  
Make sure your [IAM policy](#) includes permissions to create tags.

**Add Tag** (Up to 50 tags maximum)

At the bottom are navigation buttons: Cancel, Previous, **Review and Launch** (which is highlighted in blue), and Next: Configure Security Group.

# EC2 - Launching your first AWS instance



Services ▾

Resource Groups ▾



Sreepakash. N ▾

Mumbai ▾

Support ▾

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

## Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(127 characters maximum)	Value	(255 characters maximum)	Instances	Volumes
Name		Sree-Demo1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Add another tag (Up to 50 tags maximum)

Cancel

Previous

Review and Launch

Next: Configure Security Group

# EC2 - Launching your first AWS instance (Security groups)

Screenshot of the AWS EC2 instance creation wizard, Step 6: Configure Security Group.

The page shows the configuration for a new security group named "launch-wizard-1". A warning message at the bottom advises against using a source of "0.0.0.0/0" and recommends setting rules for known IP addresses only.

**Assign a security group:**  Create a new security group  
 Select an existing security group

**Security group name:** launch-wizard-1

**Description:** launch-wizard-1 created 2017-09-30T15:29:27.069+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom	0.0.0.0/0
e.g. SSH for Admin Desktop <span style="float: right;">X</span>				

**Add Rule**

**Warning**  
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel Previous Review and Launch

# EC2 - Launching your first AWS instance (Security groups)

The screenshot shows the AWS EC2 instance creation wizard at Step 6: Configure Security Group. The navigation bar includes links for Choose AMI, Choose Instance Type, Configure Instance, Add Storage, Add Tags, Configure Security Group (which is highlighted), and Review.

**Step 6: Configure Security Group**

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group:  Create a new security group  
 Select an existing security group

Security group name: Sree-Demo-SG-All-IP-Open

Description: Sree Demo SG All IP Open

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom	0.0.0.0/0
SSH Demo All IP Open				

**Add Rule**

**Warning**  
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel Previous Review and Launch

# EC2 - Launching your first AWS instance (Security groups)

The screenshot shows the AWS EC2 Instance Launch Wizard at Step 7: Review Instance Launch. The top navigation bar includes the AWS logo, Services, Resource Groups, a bell icon, Sreeprakash. N, Mumbai, and Support.

The progress bar at the top indicates steps 1 through 7: Choose AMI, Choose Instance Type, Configure Instance, Add Storage, Add Tags, Configure Security Group, and Review. Step 7 is highlighted.

**Step 7: Review Instance Launch**

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

**Important:** Improve your instances' security. Your security group, **Sree-Demo-SG-All-IP-Open**, is open to the world.

Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only.

You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

**AMI Details** (Edit AMI)

**Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-099fe766**  
Free tier eligible

Ubuntu Server 16.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).  
Root Device Type: ebs Virtualization type: hvm

**Instance Type** (Edit instance type)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

**Security Groups** (Edit security groups)

Security group name: Sree-Demo-SG-All-IP-Open

Launch Cancel Previous

# EC2 - Launching your first AWS instance (Key Pairs)

Select an existing key pair or create a new key pair X

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Choose an existing key pair  
[Create a new key pair](#)  
[Proceed without a key pair](#)

**No key pairs found**

You don't have any key pairs. Please create a new key pair by selecting the [Create a new key pair](#) option above to continue.

[Cancel](#) [Launch Instances](#)

# EC2 - Launching your first AWS instance (Key Pairs)

## Select an existing key pair or create a new key pair



A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

▼

**Key pair name**

SreeDemoKey



You have to download the **private key file (\*.pem file)** before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

# EC2 - Launching your first AWS instance (Key Pairs)



SreeDemoKey.pem.txt

```
-----BEGIN RSA PRIVATE KEY-----  
MIIEpAIBAAKCAQEApAReRoEAZN7mJFg6UGH/0Z2Jwk7baG7deLfBwoKEzf3i  
4qM+aciTm7l9dY/5g06CwnFtMY2SmfwKwrT0jAAUUhii6XK8Cq9tvUi4X4T  
2De9GJjx+BaS6PYq/koxBzBgB32+mVILW9qliFJL/+fLT010/01KopGQSNP  
2Er33PKQf6lKEBz8eZBexUqasxpN1f+FSsR9HtTMKD03Mpg4g/rwu/jwC5  
9/xHQhxZBWAVnfKQid8QGgL0aH3DWUgCfV2ES/XIHV7/0Xa1SVJhjwIDAQA  
/j05AL2eCeYIPci2n+5rNC9Cm42amNTMvpFrDn+oPD100N1mfDSCbuqQ80/  
iQXMYj2fLdcyJ8hRoZCd0Cmz0PJpyLc0xls3fZVPNUZRN3crvfAupAeQbngeYvZewEdskghajSb2Mjz804A4SvW80b7uYVgthubu5Ze+uX2f3QVx3WApD5  
McG4gEyuW0udZg0lIHGprnHDjQ5wZhtYB0aaSFTn+sXICWuqzJFbRPkrQWx  
3v69yfmttx71AbJv9gwaNU4XAwEcgYEAA/04T+eDnV2kutNvdz7kwnjP9USg  
Ogw1jr60juw0N2RKy3ABI5u6YdrVTDSMyJUdaq7XEg/DMUucs6L4CCKWUKq  
izWlDvX1TmEQd7Y18LsvSrkzMGAUFkh4nvjJtrWvxUJT4HCeFEECgYEApfhI  
6+TwY6VHYr614YFx0jvAVEd4yQ/STaKyQfJq39Y6zvS7pBP+u4x1DY22lFM  
M66PHejkKkmsemb91i38rWmsNQFmnYBSiVMYbGK3NmZ+0S9LdhSJgq4V9x  
gYEAgR6C2diWB85JcezUxK67WPegQQBLilaX85yZdV6SaWEzaiYsnBH1DWe  
/06AeR15MDBW02qnZHICQZPFb4/Wh9uTuynto0IsFHUXBANeStWJsgRIwZg  
SHnEj1nx4ncG+30tKhCq0kEcgYBX26GnW0FV56wedzV3twb5QFo0q5o0TsI  
Yu0eZMhJ7T2gee7EBVurA8H0oHDjFvvGVrwhLDpDogj7gXUnvn+wF6C1Inj  
VSirev1jdBH7bx02h/DWTmZt/x+yQkBERf08QLsZ7tKjGwKBgQD8dQboXm0  
RqCg2KdyJ6MRKoC0Twvqt/lD6+53Do0/gVPnVkJA8jY/596mbrHmMVlC7Iy  
AFnlmTAK9Gm1Kb6rgFiEti+Za8AkxoE96r9Jgvkcbj921wuEXoh+ffnCvzQ  
-----END RSA PRIVATE KEY-----
```

# EC2 - Launching your first AWS instance



Services ▾

Resource Groups ▾



Sreepakash. N ▾

Mumbai ▾

Support ▾

## Launch Status

✓ Your instances are now launching

The following instance launches have been initiated: [i-089a65dc9d73481e](#) [View launch log](#)

ℹ️ Get notified of estimated charges

Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

### How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can [connect](#) to them from the Instances screen. [Find out](#) how to connect to your instances.

▼ Here are some helpful resources to get you started

- [How to connect to your Linux instance](#)
- [Learn about AWS Free Usage Tier](#)
- [Amazon EC2: User Guide](#)
- [Amazon EC2: Discussion Forum](#)

While your instances are launching you can also

[Create status check alarms](#) to be notified when these instances fail status checks. (Additional charges may apply)

[Create and attach additional EBS volumes](#) (Additional charges may apply)

[Manage security groups](#)

# EC2 - Launching your first AWS instance

The screenshot shows the AWS EC2 Instances page. On the left, a sidebar menu lists various EC2 services: EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Spot Requests, Reserved Instances, Dedicated Hosts, AMIs, Bundle Tasks, Volumes, Snapshots, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancers, and Target Groups. The main content area displays a table of instances. A modal window is open for the instance 'Sree-Demo1' (i-089a65dcb9d73481e). The modal shows the instance's details, including its Public DNS (ec2-35-154-32-193.ap-south-1.compute.amazonaws.com), Instance ID (i-089a65dcb9d73481e), Instance Type (t2.micro), State (running), and Availability Zone (ap-south-1a). It also shows security group information (Sree-Demo-SG-All-IP-Open) and network details (IPv4 Public IP: 35.154.32.193, IPv6 IPs: -, Private DNS: ip-172-31-17-65.ap-south-1.compute.internal, Private IP: 172.31.17.65, Secondary private IPs: -, VPC ID: vpc-10a96a79, Subnet ID: subnet-302fe359).

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status
Sree-Demo1	i-089a65dcb9d73481e	t2.micro	ap-south-1a	running	Initializing	None

**Instance: i-089a65dcb9d73481e (Sree-Demo1) Public DNS: ec2-35-154-32-193.ap-south-1.compute.amazonaws.com**

Description	Status Checks	Monitoring	Tags
Instance ID i-089a65dcb9d73481e			Public DNS (IPv4) ec2-35-154-32-193.ap-south-1.compute.amazonaws.com
Instance state running			IPv4 Public IP 35.154.32.193
Instance type t2.micro			IPv6 IPs -
Elastic IPs			Private DNS ip-172-31-17-65.ap-south-1.compute.internal
Availability zone ap-south-1a			Private IPs 172.31.17.65
Security groups Sree-Demo-SG-All-IP-Open, view inbound rules			Secondary private IPs -
Scheduled events No scheduled events			VPC ID vpc-10a96a79
AMI ID ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20170721 (ami-009f6766)			Subnet ID subnet-302fe359

# EC2 - Launching your first AWS instance

The screenshot shows the AWS EC2 Instances page. On the left, a sidebar menu includes options like EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Instances, Spot Requests, Reserved Instances, Dedicated Hosts, Images (AMIs), Bundle Tasks, Elastic Block Store (Volumes, Snapshots), Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces), and Load Balancing (Load Balancers, Target Groups). The main content area displays a table of instances. A context menu is open over the first instance, listing options: Connect, Get Windows Password, Launch More Like This, Instance State, Instance Settings, Image, Networking, and CloudWatch Monitoring. The instance details show: Instance ID i-089a65dcb9d73481e (Sree-Demo1), Public DNS ec2-35-154-32-193.ap-south-1.compute.amazonaws.com. The instance state is running, instance type is t2.micro, and it is in the ap-south-1a availability zone. The AMI ID is ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20170721. Other details include security groups (Sree-Demo-SG-All-IP-Open), no scheduled events, and VPC ID vpc-10a96a79.

Instance ID	Public DNS (IPv4)
i-089a65dcb9d73481e	ec2-35-154-32-193.ap-south-1.compute.amazonaws.com
Instance state	running
Instance type	t2.micro
Elastic IPs	
Availability zone	ap-south-1a
Security groups	Sree-Demo-SG-All-IP-Open . view inbound rules
Scheduled events	No scheduled events
AMI ID	ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20170721
Public DNS (IPv4)	ec2-35-154-32-193.ap-south-1.compute.amazonaws.com
IPv4 Public IP	35.154.32.193
IPv6 IPs	-
Private DNS	ip-172-31-17-65.ap-south-1.compute.internal
Private IPs	172.31.17.65
Secondary private IPs	
VPC ID	vpc-10a96a79
Subnet ID	subnet-302fe359

# EC2 - Launching your first AWS instance

## Connect To Your Instance

x

I would like to connect with

A standalone SSH client

A Java SSH Client directly from my browser (Java required)

To access your instance:

1. Open an SSH client. (find out how to [connect using PuTTY](#))
2. Locate your private key file (`SreeDemoKey.pem`). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:

**chmod 400 SreeDemoKey.pem**

4. Connect to your instance using its Public DNS:

**ec2-35-154-32-193.ap-south-1.compute.amazonaws.com**

Example:

**ssh -i "SreeDemoKey.pem" ubuntu@ec2-35-154-32-193.ap-south-1.compute.amazonaws.com**

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

**Close**

# EC2 - Launching your first AWS instance

```
Sree>ls -lt | grep SreeDemoKey
-rw-r--r--@ 1 sree staff 1696 Sep 30 15:31 SreeDemoKey.pem.txt
Sree>
```

# EC2 - Launching your first AWS instance

```
Sree>ssh -i "SreeDemoKey.pem.txt" ubuntu@ec2-35-154-32-193.ap-south-1.compute.amazonaws.com
The authenticity of host 'ec2-35-154-32-193.ap-south-1.compute.amazonaws.com (35.154.32.193)' can't be established.
ECDSA key fingerprint is SHA256:5xV/UE415r+Ws9otAmEV8C7J581TRHeySa201gMkrGI.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-35-154-32-193.ap-south-1.compute.amazonaws.com,35.154.32.193' (ECDSA) to the list of known hosts.
oooooooooooooooooooooooooooooooooooooooooooo
@      WARNING: UNPROTECTED PRIVATE KEY FILE!      @
oooooooooooooooooooooooooooooooooooooooooooo
Permissions 0644 for 'SreeDemoKey.pem.txt' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "SreeDemoKey.pem.txt": bad permissions
Permission denied (publickey).
Sree>
```

# EC2 - Launching your first AWS instance

```
[Sree>chmod 400 SreeDemoKey.pem.txt
```

# EC2 - Launching your first AWS instance

```
Sree>chmod 400 SreeDemoKey.pem.txt
Sree>ssh -i "SreeDemoKey.pem.txt" ubuntu@ec2-35-154-32-193.ap-south-1.compute.amazonaws.com
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-1022-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
  http://www.ubuntu.com/business/services/cloud

0 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-17-65:~$
```

# EC2 - Launching your first AWS instance (Terminating)

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Spot Requests, Reserved Instances, Dedicated Hosts, Images (AMIs), Bundle Tasks, Elastic Block Store (Volumes, Snapshots), and Network & Security. The main area displays a table of instances. One instance, "Sree-Demo1" (i-089a65dcb9d73481e), is selected. An "Actions" dropdown menu is open over this instance, showing options: Connect, Get Windows Password, Launch More Like This, Instance State, Instance Settings, Image, Networking, CloudWatch Monitoring, and Terminate. The "Instance State" option is expanded, showing Start, Stop, Reboot, and Terminate. Below the table, detailed information for the selected instance is shown: Instance ID i-089a65dcb9d73481e, Public DNS ec2-35-154-32-193.ap-south-1.compute.amazonaws.com, Instance state running, Instance type t2.micro, Public DNS (IPv4) ec2-35-154-32-193.ap-south-1.compute.amazonaws.com, IPv4 Public IP 35.154.32.193, and IPv6 IPs -.

Availability Zone	Instance State	Status Checks	Alarm Status
	running	2/2 checks ...	None

Description	Status Checks	Monitoring	Tags
Instance ID i-089a65dcb9d73481e	Public DNS (IPv4) ec2-35-154-32-193.ap-south-1.compute.amazonaws.com	IPv4 Public IP 35.154.32.193	IPv6 IPs -
Instance state running	Instance type t2.micro		

## Terminate Instances



### Warning

On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.

Are you sure you want to terminate these instances?

i-089a65dcb9d73481e (Sree-Demo1, ec2-35-154-32-193.ap-south-1.compute.amazonaws.com)

Cancel

Yes, Terminate

# EC2 - Launching your first AWS instance

The screenshot shows the AWS EC2 Instances page. At the top, there are three buttons: 'Launch Instance' (blue), 'Connect' (grey), and 'Actions' (grey dropdown). Below these are search and filter fields. The main table displays one instance:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm S
Sree-Demo1	i-089a65dcb9d73481e	t2.micro	ap-south-1a	shutting-do...	None	

# EC2 - Launching your first AWS instance

```
ubuntu@ip-172-31-17-65:~$ Connection to ec2-35-154-32-193.ap-south-1.compute.amazonaws.com closed by remote host.  
Connection to ec2-35-154-32-193.ap-south-1.compute.amazonaws.com closed.  
Sree>
```

# EC2 - Launching your first AWS instance

## IMPORTANT POINTS TO NOTE:

1. Remember the region and A-Z you are launching the instance into
2. Save the Key in a secure location (600 or 400 permissions)
3. Terminate the instance immediately after use

## S3: What is S3?

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

Amazon S3 has a simple web services interface that you can use 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, fast, inexpensive data** storage infrastructure that Amazon uses to run its own global network of web sites.

To upload your data (photos, videos, documents etc.), you first **create a bucket** in one of the **AWS Regions**. You can then upload **any number** of objects to the bucket.

Buckets and objects are resources, and Amazon S3 provides APIs for you to manage them. For example, you can create a bucket and upload objects using the Amazon S3 API. You can also use the Amazon S3 console to perform these operations. The console internally uses the Amazon S3 APIs to send requests to Amazon S3.

## S3: Advantages/Features

**Create Buckets** – Create and name a bucket that stores data. Buckets are the fundamental container in Amazon S3 for data storage.

**Store data in Buckets** – Store an **infinite** amount of data in a bucket. Upload as many objects as you like into an Amazon S3 bucket. Each object can contain up to **5 TB** of data. Each object is stored and retrieved using a unique developer-assigned key.

**Download data** – Download your data or enable others to do so. Download your data any time you like or allow others to do the same.

**Permissions** – Grant or deny access to others who want to upload or download data into your Amazon S3 bucket. Grant upload and download permissions to three types of users. Authentication mechanisms can help keep data secure from unauthorized access.

**Standard interfaces** – Use standards-based REST and SOAP interfaces designed to work with any Internet-development toolkit.

# Simple Notification Service (SNS): What is it?

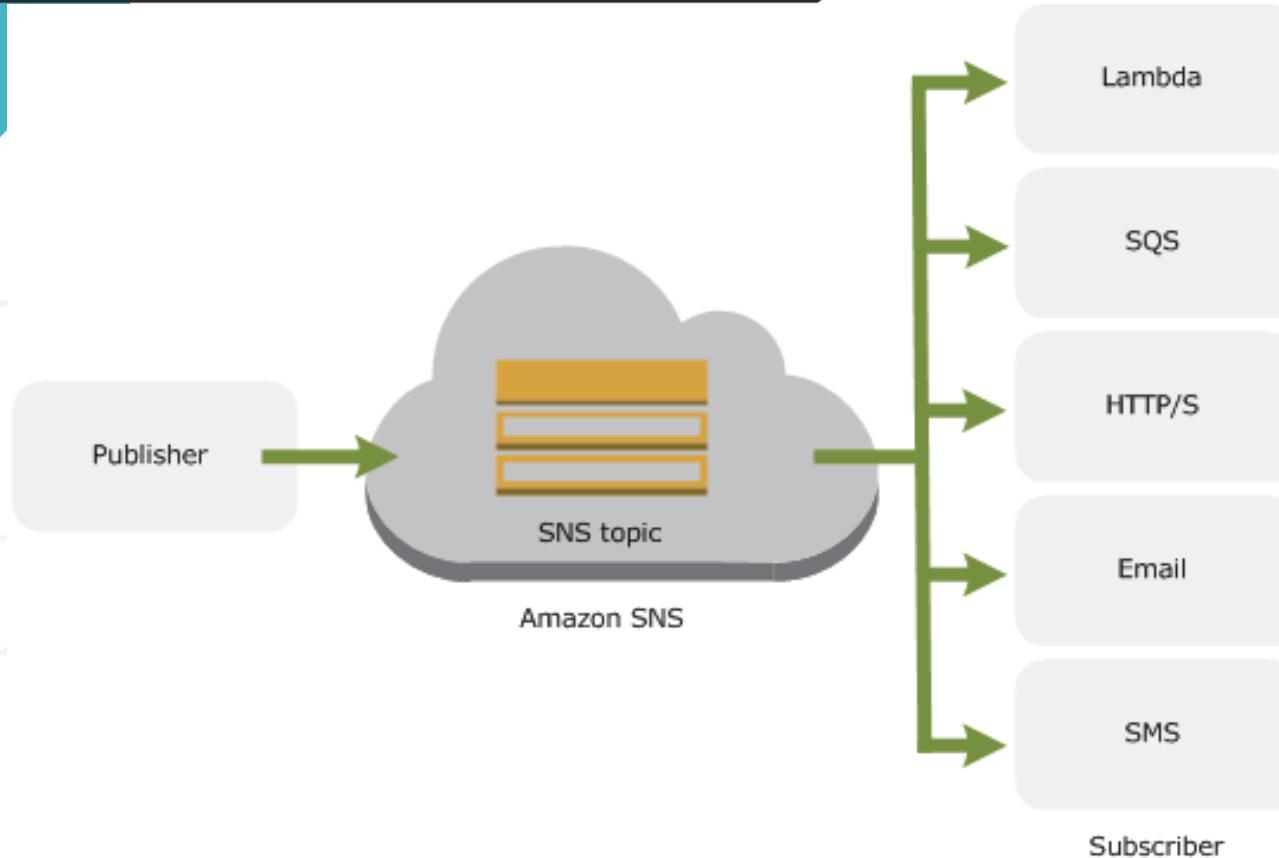
Amazon Simple Notification Service (Amazon SNS) is a web service that coordinates and manages the delivery or sending of messages to **subscribing** endpoints or clients.

In Amazon SNS, there are two types of clients—**publishers** and **subscribers**—also referred to as **producers** and **consumers**.

**Publishers** communicate **asynchronously** with subscribers by producing and sending a message to a topic, which is a logical access point and communication channel.

**Subscribers** (i.e., web servers, email addresses, SMS, Amazon SQS queues, AWS Lambda functions) consume or receive the message or notification over one of the supported protocols (i.e., Amazon SQS, HTTP/S, email, SMS, Lambda) when they are subscribed to the topic.

# Simple Notification Service (SNS): What is it?



## SNS: Creating a topic

A topic is a **communication channel** to send messages and subscribe to notifications. It provides an **access point** for publishers and subscribers to communicate with each other.

**To create a topic:** In the Amazon SNS console, click Create topic.

The Create topic dialog box appears. In the Topic name box, type a topic name.

Click Create topic. The new topic appears in the Topics page.

Select the new topic and then click the topic ARN.

The Topic Details page appears.

## CloudWatch: What is it?

Amazon **CloudWatch** provides a reliable, scalable, and flexible **monitoring** solution that you can start using within minutes. You no longer need to set up, manage, and scale your own monitoring systems and infrastructure.

Use CloudWatch to monitor your AWS resources and the applications you run on AWS in **real time**.

Use **CloudWatch Events** to send system events from AWS resources to AWS Lambda functions, Amazon SNS topics, streams in Amazon Kinesis, and other target types.

Use **CloudWatch Logs** to monitor, store, and access your log files from Amazon EC2 instances, AWS CloudTrail, or other sources.

## CloudWatch: Monitoring other AWS services

You can monitor the CPU usage and disk reads and writes of your Amazon EC2 instances and then use this data to determine whether you should launch additional instances to handle increased load (auto scaling).

You can also use this data to stop under-used instances to save money.

In addition to monitoring the **built-in metrics** that come with AWS, you can monitor **your own custom metrics**.

With CloudWatch, you gain system-wide visibility into resource utilization, application performance, and operational health.

# CloudFormation: What is it?

Model and *provision* all your cloud infrastructure resources.

AWS CloudFormation provides a common language for you to describe and provision all the infrastructure resources in your cloud environment. You use a simple text file to model and provision, in an automated and secure manner, all the resources needed for your applications across all regions and accounts.

AWS CloudFormation is available at no additional charge, and you pay only for the AWS resources needed to run your applications.

## CloudFormation: Concepts

AWS CloudFormation introduces two concepts:

- **The template**, a JSON or YAML-format, text-based file that describes all the AWS resources you need to deploy to run your application
- **The stack**, the set of AWS resources that are created and managed as a single unit when AWS CloudFormation instantiates a template.

By default, the “automatic rollback on error” feature is enabled. So if any resource in the stack fails the whole stack is rolled back.

## Elastic Beanstalk: What is it?

AWS Elastic Beanstalk makes it even easier for developers to quickly deploy and manage applications in the AWS Cloud. Developers simply upload their application, and Elastic Beanstalk *automatically handles the deployment* details of capacity provisioning, load balancing, auto-scaling, and application health monitoring.

AWS Elastic Beanstalk supports Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker, and is *ideal for web applications*. However, due to Elastic Beanstalk's open architecture, non-web applications can also be deployed using Elastic Beanstalk.

There is no additional charge for Elastic Beanstalk. You pay only for the underlying AWS resources that your application consumes.

## Elastic Beanstalk: Concepts

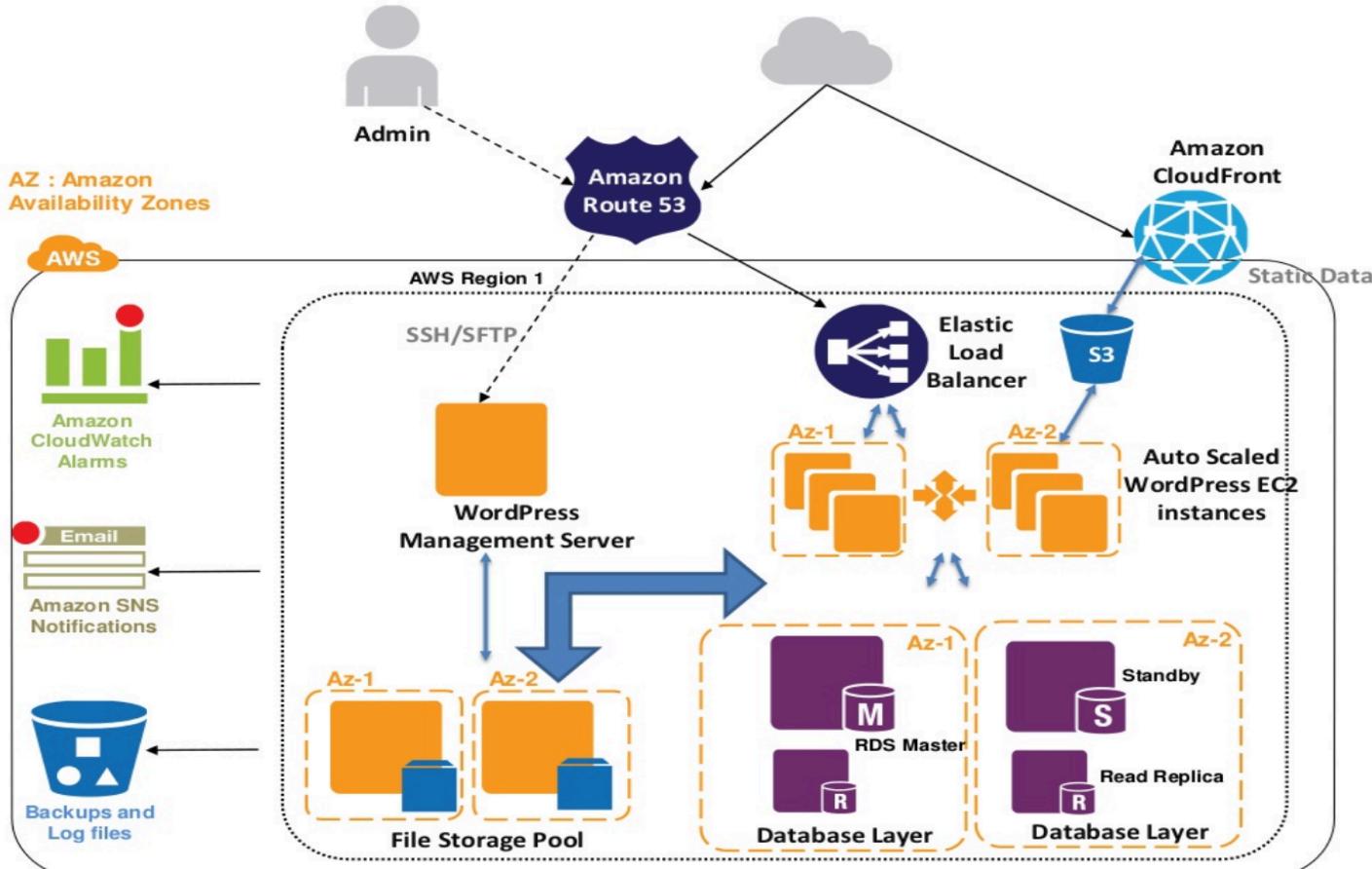
Most existing application containers or *platform-as-a-service* solutions, force developers to live with all the decisions predetermined by the vendor—with little to no opportunity to take back control. However, with AWS Elastic Beanstalk, developers can decide to manage some (or all) of the elements of their infrastructure, seamlessly by using Elastic Beanstalk's management capabilities.

AWS Elastic Beanstalk runs on the Amazon Linux AMI and the Windows Server 2012 R2 AMI.

Once your application is running, Elastic Beanstalk *automates management tasks*—such as monitoring, application version deployment, a basic health check—and facilitates log file access.

Information about the application—including metrics, events, and environment status—is available through the AWS Management Console, APIs, or Command Line Interfaces.

# Highly Available, Fault Tolerant and Scalable Application





Cognixia

**THANK YOU**

