

Amazing Amazon Web Services



SRIRAM

Dedication



Dedicated to Mr. Karthik
AWS Chief Cloud Architect

“Invention requires two things:

- ✓ The ability to try a lot of experiments
- ✓ Not having to live with the collateral damage of failed experiments”

— Andy Jassy CEO, Amazon Web

Services

Never be a prisoner of your past, Be a Cloud Architect of your future

If you are not loving Cloud you might be doing wrong in spending patterns

— Sriram, Chief Cloud Architect

Preface

I have been involved in IT Software development since 1997. I have a unique combination of process, technical and industrial skills. As an Enterprise Architect, I have expert level of knowledge in agile and technology practices such as AWS, Azure, DevOps, java, Hadoop, SharePoint & .Net with this combination I can help process and technology people, understand the world. Worked in India, USA, and UK which creates a global experience and awarded as a Best Enterprise Architect. Dedicated “[Amazing Amazon Web Services](#)” book to my family members, friends. This Guide made handy and recollect everything at one shot.

Organization of this Book

Amazing Amazon Web Services is designed to make you to success in the interview by providing valuable discussions on various topics along with the tips to achieve AWS Architect Certification. The progressive elaboration of AWS knowledge towards an AWS Architect is awesome. Enjoy Reading!

Table of Contents

ABOUT SRIRAM	5
CLOUD COMPUTING	9
AWS ESSENTIALS	23
COMPUTE	48
Amazon EC2	49
Elastic IP	88
Elastic File System	103
Elastic Block Store	114
Amazon Machine Image	147
Amazon EC2 Auto-scaling	154
AWS Elastic Bean Stack	177
AWS Lambda	191
STORAGE	195
Amazon S3	196
Glacier	236
Storage Gateway	244
Snowball	246
DATABASE	248
RDS	249
DynamoDB	275
Redshift	291
SECURITY, IDENTITY & COMPLIANCE	293
Identity Access & Management	294
NETWORKING & CONTENT DELIVERY	337
Amazon VPC	338
Amazon CloudFront	433
Amazon Route 53	453
AWS Load Balancing	472
MANAGEMENT TOOLS	482
AWS CloudWatch	483
Amazon CloudFormation	498

AWS OpsWorks	512
APPLICATION INTEGRATION & CUSTOMER ENGAGEMENT	515
AWS Simple Queue Service.....	516
AWS Simple Notification Service.....	530
AWS Simple Email Service	540
ANALYTICS	554
BUSINESS PRODUCTIVITY	561
DESKTOP & APP STREAMING.....	564
AWS ARCHITECTURE	568
AWS MIGRATION	590
AWS IOT	594



About Sriram



Working as a Technical Services Delivery Manager in ASAP Data Solutions – Malvern, Pennsylvania, United States of America. Having 18+ years of successful experience in architectural design, development, delivery and manage complex projects with high-performance in real time solutions. I have Worked in US, UK & India, expert in managing continuous business operations and support.

Four pillars of operations make success in the customer engagement such as business operations, team management, project management, customer management.

Business Operations

- Handled both technology & business teams get more focused on business objectives via agile methodologies with ROI over \$20 billion on each account
- Work closely with client partner to generate more business from the client
- Work with sales team to enhance the business and provide the incredible support to the customer
- 100% success in deals winning across multiple vendors
- Successful Demand creation and placed the resources on niche skills
- Generated more billability to team to enhance the business of an organization
- Build a customer rapport geographically to support their needs
- Generated a good profit margin for the organization

Team Management

- Delivery on time much less overhead
- Team collaboration & guidance to grow as per the client expectations
- Ensure adherence to defined development life cycle, good software design practices, and architecture strategy and intent.
- Prompt remediation of any issues & blockers for the team
- Being a certified professional & technical expertise by providing solutions to multiple technologies such as Java, Bigdata, Amazon Web Service, Microsoft Azure, SharePoint, .Net & DevOps
- Developed full stack developer to handle the project effectively and efficiently to handle the complex situations
- Being an enterprise coach, I have taken care of team, meeting monthly to shape them in agile career
- Produced zero defect bug free product

Project Management

- Responsible for the performance and success of azure projects to include planning and managing scope, schedule, cost, quality, risk, resources, procurement, communication and tracking of project results
- Work with project charter | plan, roles, tasks, milestones, budgets and measures of success.
- Ensure client requirements are captured accurately and completely. Create and maintain project documentation. Facilitate day-to-day coordination while adhering to standards and sponsor expectations in cloud.
- Monitor projects on an ongoing basis, evaluate progress, quality and manage issue resolution.
- Monitor financial delivery and issue management processes and escalate issues.
- Develop project risk management plans to ensure timely delivery, testing and commissioning of all projects with no impact to business continuity.
- Serve as primary contact to project team members, customers, senior department managers and key stakeholders for status updates and critical change initiatives.

Customer Management

- Business improvement across the geographical location & Cloud promotion for the existing and new projects
- Product Demonstration and customization delivery to the client
- Cost Reduction in maintaining the essential services and efficient storage
- Solution Architecture for cloud-based project
- Cloud Assessment & Migration
- Customer Satisfaction Index above 95% consistently

Achievements

- Generated revenue from 10 to 20 billion for the past 2 years for my company
- Cost reduction benefit to the client over 30% in effectively managing cloud platform
- Build the high-performance team in an agile fashion

AWS Certified Professional



Sriram Balasubramanian

has successfully completed the AWS Certification requirements and has achieved their:

AWS Certified Solutions Architect - Associate

Issue Date
November 30, 2017

Expiration Date
November 30, 2019

A handwritten signature in black ink, appearing to read "Maureen Loneragan".

Maureen Loneragan
Director, Training and Certification

Validation Number PRTJNWF2KF111P5W
Validate at: <http://aws.amazon.com/verification>

Awards & Honors

Best Cloud Solution Architect – “On The Spot Award”, Tata Consultancy Services, 2013





Best Architect Award

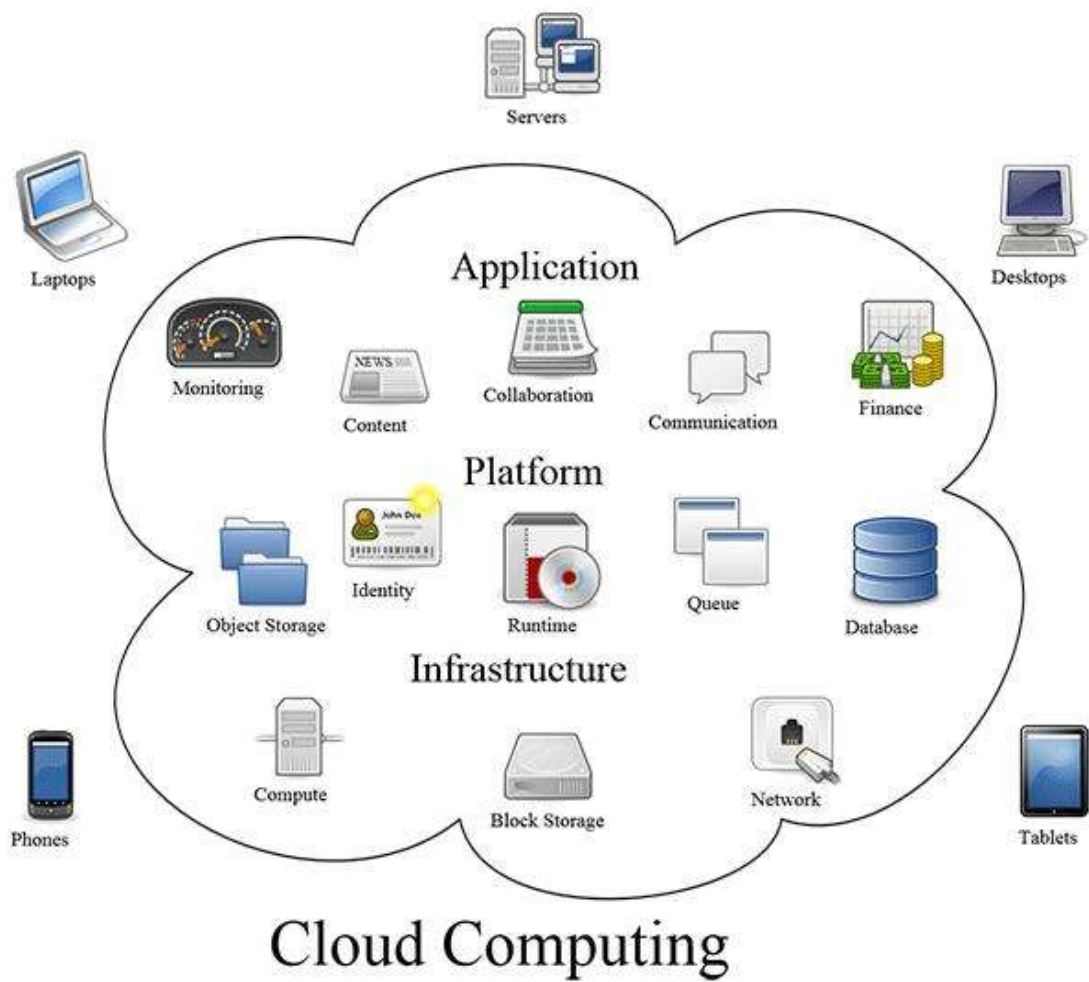
Dear Mr. Sriram Balasubramanian I would like to appreciate you on behalf of Banking & Financial Services for being a Chief Architect, in building and providing solutions in the area of Cloud Computing – Amazon Web Services (AWS), Microsoft Azure & DevOps project, which is more valuable towards client partner solutions with the alignment of our organization I look forward to your continued support.

2017

V. S. Raj
BU Head - BNFS

Cloud Computing

Cloud Computing Intro	3 Service Models	4 Deployment Models
5 Characteristics	Advantages of Cloud Computing	Cloud Service Providers



Cloud Computing Intro

What is Cloud Computing?

Practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or a personal computer is called Cloud Computing.

Companies offering these computing services are called “cloud providers” and typically charge for cloud computing services based on usage, similar to how you are billed for water or electricity at home. E.g.: AWS, AZURE, IBM Bluemix, GOOGLE CLOUD

This cloud model is composed of five essential characteristics, three service models, and four deployment models.

The primary reasons for moving to the cloud are: -

- You don't need to maintain or administer any infrastructure
- It will never run out of capacity, since it is a virtually infinite
- You can access your cloud-based applications from anywhere, you just need a device which can connect to the internet

What are the benefits of Cloud Computing?

- Totally free from Maintenance i.e., You do not have to maintain or administer any infrastructure for the same.
- Lower Computing Cost
- Improved Performance
- Reduced Software Cost
- Instant Software Updates
- Unlimited Storage Capacity i.e., It will never run out of capacity, since it is virtually infinite.
- Increased Data Reliability
- Device Independence and the “always on! anywhere and any place” i.e., You can access your cloud-based applications from anywhere, you just need a device which can connect to the internet.

Cloud Computing is the fastest growing part of network-based computing. It provides tremendous benefits to customers of all sizes: simple users, developers, enterprises and all types of organizations.

Why Cloud Computing?

- Lower TCO
- Reliability, Scalability & Sustainability
- Secure Store Management
- Low Capital Expenditure
- Frees from Internal Resources
- Utility Based
- Easy & Agile Deployment
- Device & Location Independent
- 24 * 7 Support
- Pay As You Use

What are the top 10 advantages of Cloud Computing?

- Pay as you Go Model
- Increased Mobility
- Less or No CAPEX
- High Availability
- Easy to Manage
- High Productivity
- Environment Friendly
- Less Deployment Time
- Dynamic Scaling
- Shared Resources

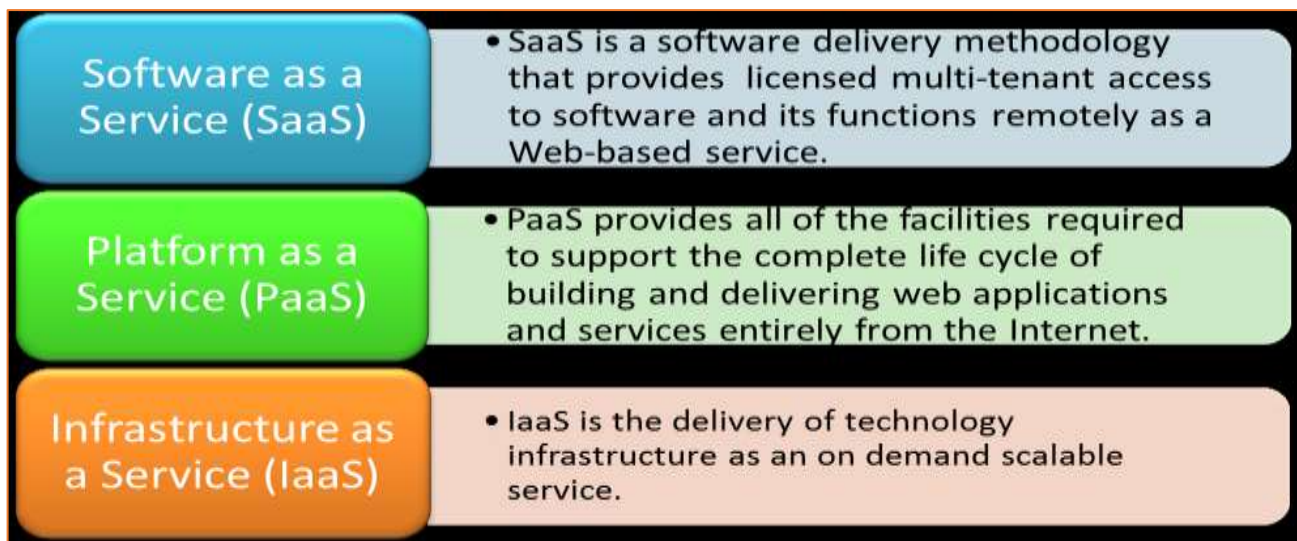
3 Service Models

What are the different layers (Service Models) of cloud computing?

Cloud computing consists of 3 layers in the hierarchy and these are as follows:

1. **Infrastructure as a Service (IaaS)** provides cloud infrastructure in terms of hardware like memory, processor speed etc.
2. **Platform as a Service (PaaS)** provides cloud application platform for the developers.
3. **Software as a Service (SaaS)** provides cloud applications which are used by the user directly without installing anything on the system. The application remains on the cloud and it can be saved and edited in there only.

What are the 3 service models of Cloud Computing?



IaaS

Infrastructure as a Service (IaaS) is the most basic category of cloud computing services. With IaaS, you rent IT infrastructure—servers and virtual machines (VMs), storage, networks, operating systems—from a cloud provider on a pay-as-you-go basis.

The Key features are: -

- Instead of purchasing hardware outright, users pay for IaaS on demand.
- Infrastructure is scalable depending on processing and storage needs.
- Saves enterprises the costs of buying and maintaining their own hardware.
- Because data is on the cloud, there is no single point of failure.
- Enables the virtualization of administrative tasks, freeing up time for other work.

PaaS

Platform-as-a-service (PaaS) refers to cloud computing services that supply an on-demand environment for developing, testing, delivering and managing software applications. PaaS is designed to make it easier for developers to quickly create web or mobile apps, without worrying about setting up or managing the underlying infrastructure of servers, storage, network and databases needed for development.

The Key features are: -

- PaaS provides a platform with tools to test, develop, and host applications in the same environment.
- Enables organizations to focus on development without having to worry about underlying infrastructure.
- Providers manage security, operating systems, server software, and backups.
- Facilitates collaborative work even if teams work remotely.

PaaS Examples: AWS Elastic Beanstalk, Microsoft Azure, Google Apps, Salesforce Force.com & IBM Bluemix

PaaS Billing: PaaS will typically be billed based on memory usage i.e., IBM Bluemix

SaaS

Software-as-a-service (SaaS) is a method for delivering software applications over the Internet, on demand and typically on a subscription basis. With SaaS, cloud providers host and manage the software application and underlying infrastructure and handle any maintenance, like software upgrades and security patching. Users connect to the application over the Internet, usually with a web browser on their phone, tablet or PC.

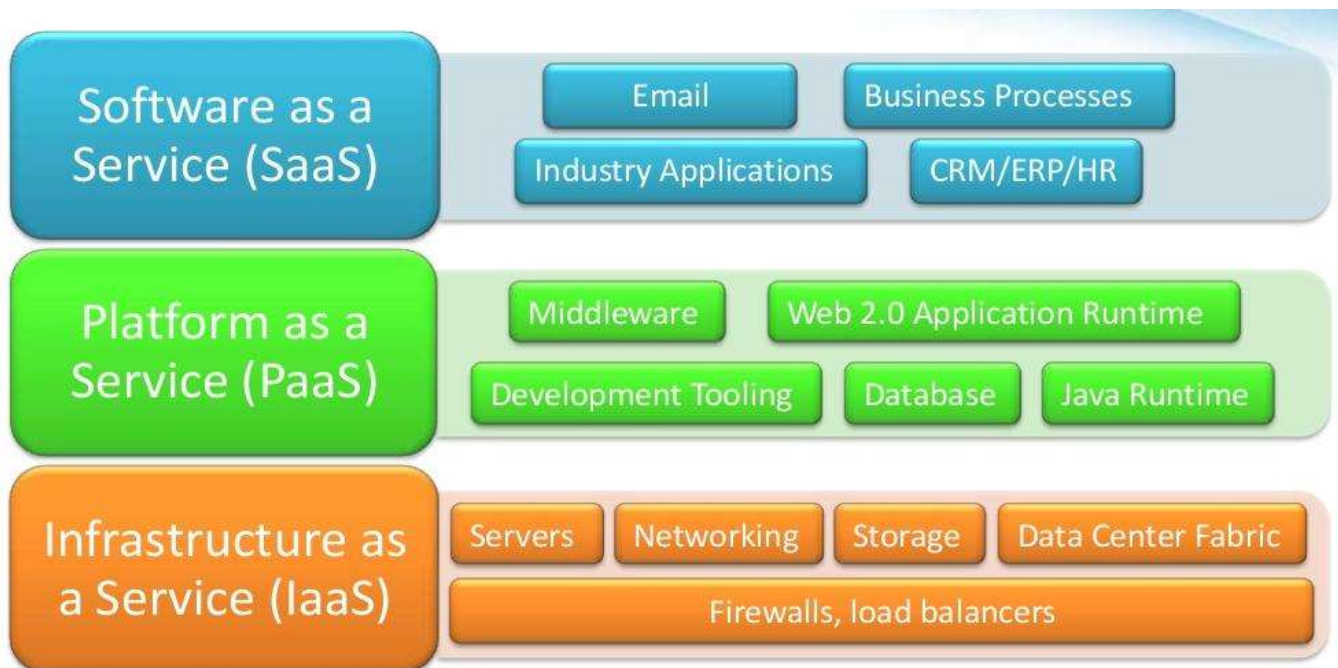
The Key features are: -

- SaaS vendors provide users with software and applications on a subscription model.
- Users do not have to manage, install, or upgrade software; SaaS providers manage this.
- Data is secure in the cloud; equipment failure does not result in loss of data.
- Use of resources can be scaled depending on service needs.
- Applications are accessible from almost any Internet-connected device, from virtually anywhere in the world.

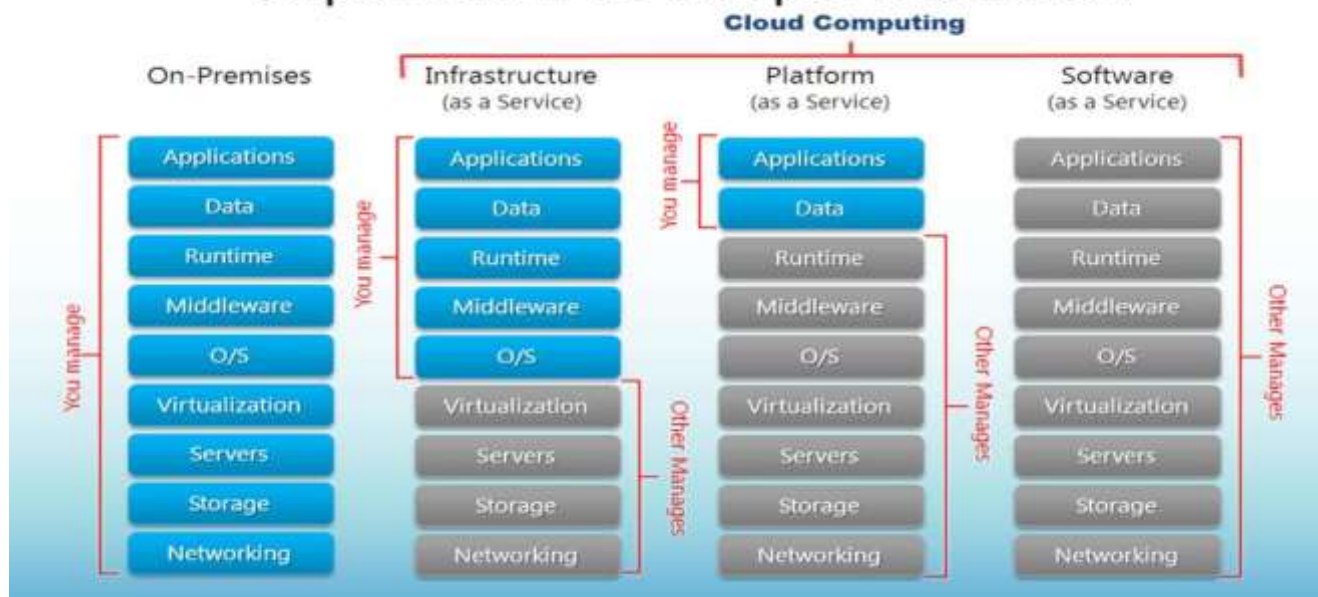
SaaS Examples: Microsoft Office 365, Salesforce.com, Intuit, Adobe Creative Cloud & Gmail

SaaS Billing:

- SaaS will typically have a monthly fee per user
- Multiple pricing tiers may be offered based on usage E.g. Microsoft Office 365



Separation of Responsibilities



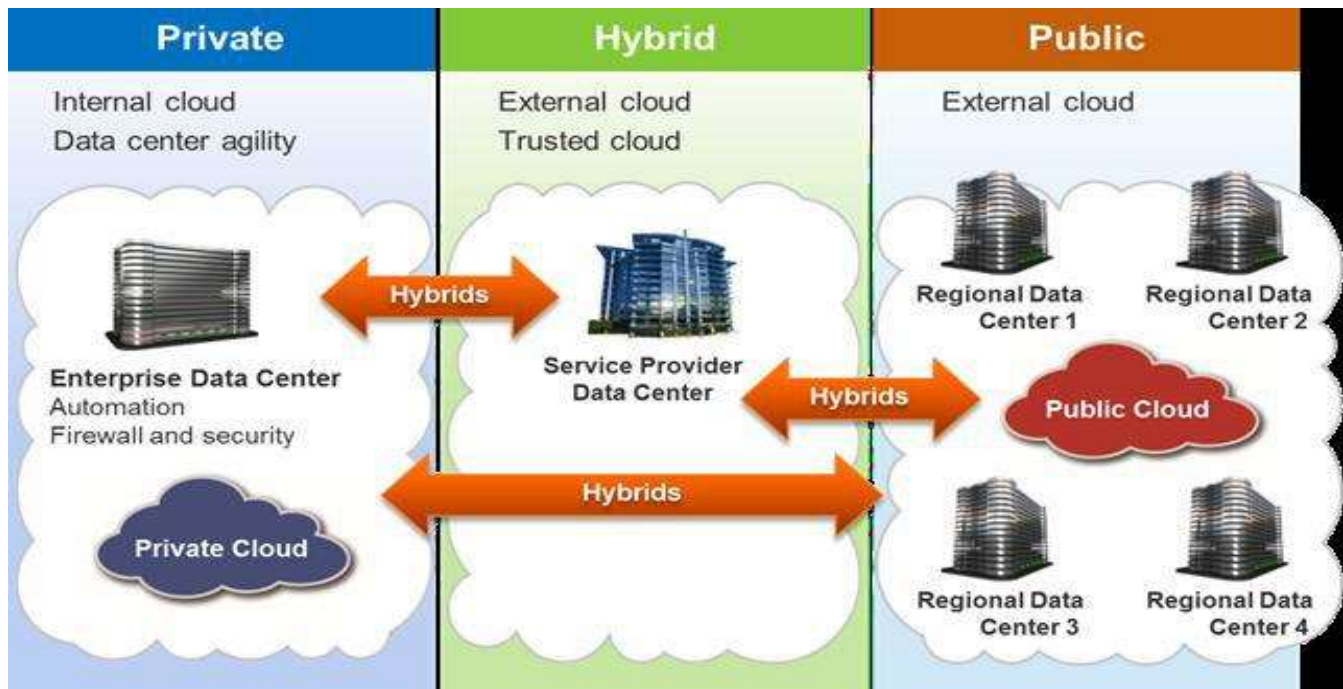
4 Deployment Models

What are the 4 deployment models of Cloud Computing?

The NIST define three Deployment Models of Cloud Computing:

- Public Cloud
- Private Cloud
- Hybrid Cloud

- Community Cloud



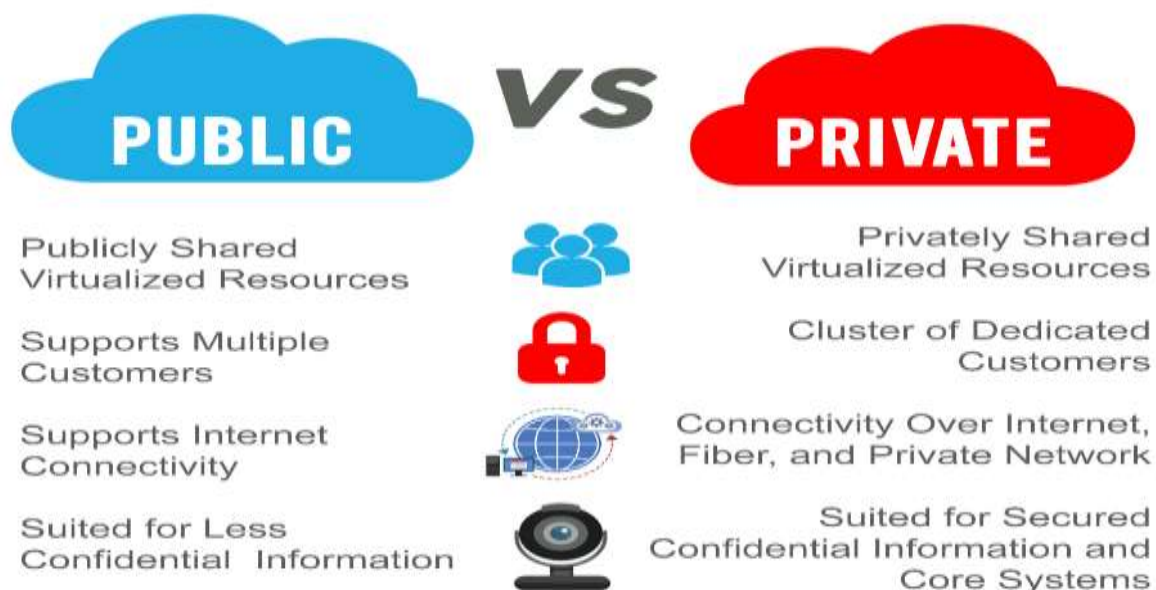
Public clouds are owned and operated by a third-party cloud service provider, which deliver their computing resources like servers and storage over the Internet. Made available to the general public or a large industry group.

Microsoft Azure is an example of a public cloud. With a public cloud, all hardware, software and other supporting infrastructure is owned and managed by the cloud provider. You access these services and manage your account using a web browser. The most common deployment model. **Examples:** All the cloud providers such as AWS, Microsoft Azure, IBM Bluemix, Salesforce, etc.,

Private Cloud works the same way as Public Cloud, but these services are provided to internal business units instead of to external public enterprises.

Operated solely for an organization, may be managed by the organization or a third party and Limits access to enterprise and partner network

A private cloud refers to cloud computing resources used exclusively by a single business or organization. A private cloud can be physically located on the company's on-site datacenter. Some companies also pay third-party service providers to host their private cloud. A private cloud is one in which the services and infrastructure are maintained on a private network. Examples: US DOD, Indian Military, Most of govt bodies and High revenue business.



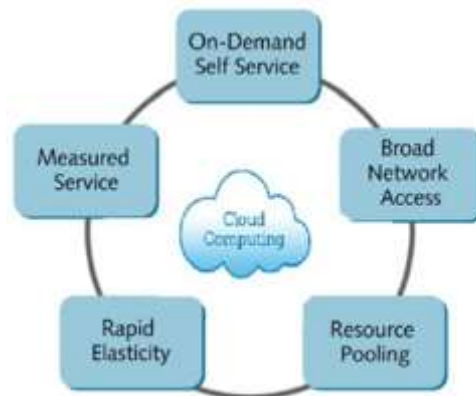
Hybrid clouds combine public and private clouds, bound together by technology that allows data and applications to be shared between them. By allowing data and applications to move between private and public clouds, hybrid cloud gives businesses greater flexibility and more deployment options. Companies with limited Private Cloud infrastructure may 'cloud burst' into Public Cloud for additional capacity when required. A company Cloud also have Private Cloud at their main site and use Public Cloud for their Disaster Recovery location

Composition of two or more clouds (private, community, or public) bound together by standardized or proprietary technology that enables data and application portability

Community Cloud is similar to a traditional extranet, but with full shared data center services instead of just network connectivity between On-Premise Offices.

5 Characteristics

What are the 5 characteristics of Cloud Computing? How Private Cloud differ than On-Premise?



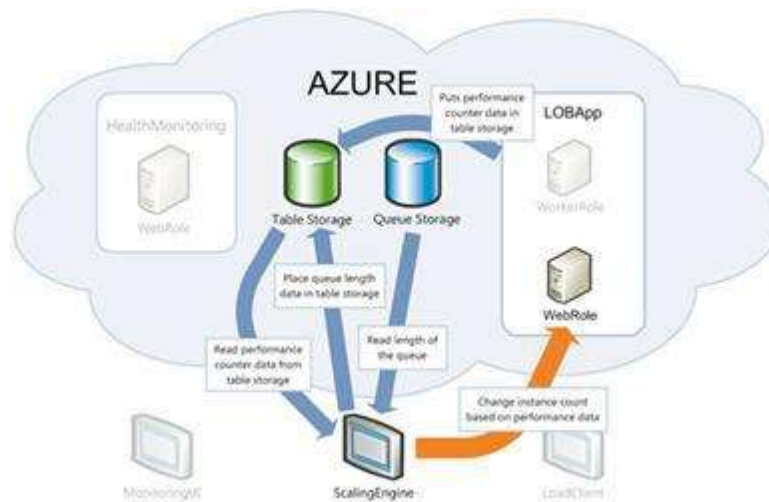
On-Demand Self Service: On-demand self-service: A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider. – NIST



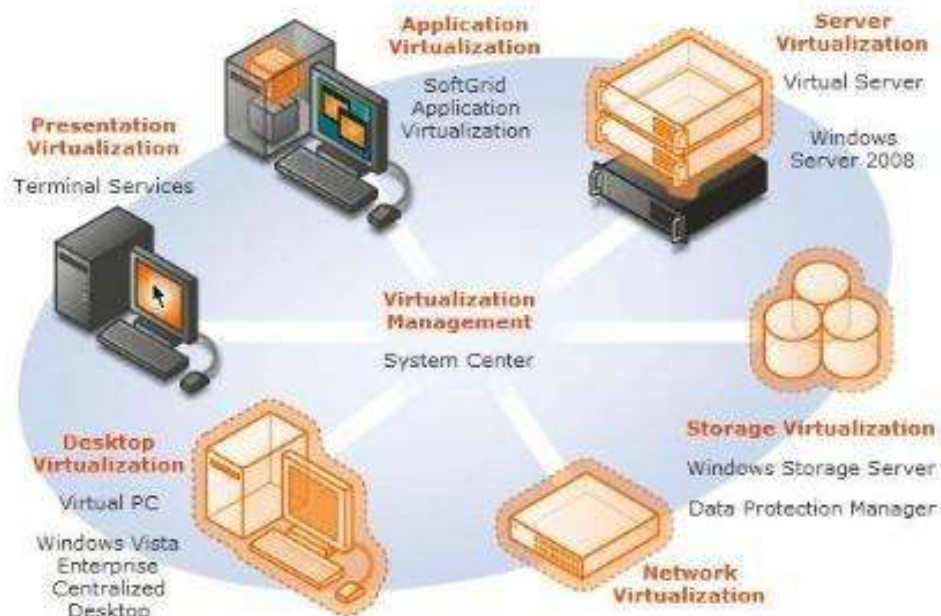
Broad (Ubiquitous) network access: Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations). – NIST



Rapid elasticity: Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time. – NIST



Resource pooling: The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources include storage, processing, memory, and network bandwidth. – NIST



Measured service: Cloud systems automatically control and optimize resource use by leveraging a metering capability¹ at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service. - NIST



Advantages of Cloud Computing

What are the advantages of Cloud Computing?

Scalability

- Cloud Computing provides business with the ability to regulate the service in accordance with their current requirements:
- Scale capacity up and down as needed
- Infinite computing capacity on demand
- Flexibility through cloud bursting

Business Agility

- Ability to handle expected or unexpected changes in load
- Reduced time to deploy an application into production

Cost Efficiency

- The customer pays just for what they need, resulting in directly proportional costs
- The customer avoids provisioning for the peak as a permanent fixture
- Move from a large upfront CapEx cost to a comparatively small monthly OpEx cost
- ICT costs are more transparent to the business
- The customer does not have depreciable hardware assets
- Technology refresh is the responsibility of the Cloud Provider
- The provider passes hardware maintenance costs onto the customer as part of the predictable monthly fee, there are no unexpected costs

Competitive Advantage

- Organizations can respond quickly to evolving market trends and focus on growing their core business
- Reducing capital spent on infrastructure releases funds to invest in innovation or other priority areas

Productivity

- IT staff can focus more on strategic decisions and developing and improving core applications rather than maintaining or troubleshooting in-house ICT

Availability & Reliability

- All major Cloud Providers facilities are located in hardened data centers with redundant power, no single point of failure and onsite security
 - The service will be certified to the relevant industry standards such as ISO 9001 (Quality) and 27001 (Security)

- The data center is built by facilities, server, networking and storage qualified specialists according to best practice
- Check the Service Level Agreement to see what is guaranteed and the compensation if the SLA is not met

Cost

- The advantages are all great to have, but a decision to deploy Cloud Computing usually comes down to the overall long-term cost
- The TCO of maintaining an On Premises solution should be compared to the TCO of maintaining a Cloud equivalent, and the advantages and disadvantages of each factored in when making the final decision
- It is not a wither or decision. The majority of companies who use Cloud services will have a mix of On Premise and Cloud solutions



Data Center Costs

- CapEx Cost: Hardware Procurement
- OpEx Cost: Rack space, Power and Cooling, On-going management

On Premises Solution				IaaS Cloud Solution			
		Cost of each server	\$6,000		Monthly	\$6,000	
		Server refresh cycle	5 Years		Yearly	\$72,000	
	Cost of running servers per year (power, cooling, rack space, maintenance)	\$3,000			Installation Fee	\$0	
		Number of servers	12				
	Cost of IT support per year for hardware and backups	\$50,000					
	Tape library and backup software (one off cost)	\$20,000					
	CapEx (No. servers x Cost per server + Tape library and backup software)	\$92,000					
	OpEx (No. of servers x Cost of running servers x 5 years + IT support x 5 years)	\$430,000					
		Total CapEx plus OpEx over 5 Years	\$522,000			\$360,000	
					Cost Saving	\$162,000	

Finally, we can conclude 30% of cost savings with zero down time and high performance. Organization can invest in Cloud to achieve the greater benefits.

Cloud Service Providers

Who are all the Cloud Service Providers?



General

Name the various layers of the cloud architecture?

There are 5 layers and are listed below

- CC- Cluster Controller
- SC- Storage Controller
- CLC- Cloud Controller
- Walrus
- NC- Node Controller

What is the way to secure data for carrying in the cloud?

One thing must be ensured that no one should seize the information in the cloud while data is moving from point one to another and also there should not be any leakage with the security key from several storerooms in the cloud. Segregation of information from additional companies' information and then encrypting it by means of approved methods is one of the options. Amazon Web Services offers you a secure way of carrying data in the cloud.

How to secure your data for transport in cloud?

Cloud computing provides very good and easy to use feature to an organization, but at the same time it brings lots of question that how secure is the data, which has to be transported from one place to another in cloud. So, to make sure it remains secure when it moves from point A to point B in cloud, check that there is no data leak with the encryption key implemented with the data you sending.

How does cloud computing provide on-demand functionality?

Cloud computing is a metaphor used for internet. It provides on-demand access to virtualized IT resources that can be shared by others or subscribed by you. It provides an easy way to provide configurable resources by taking it from a shared pool. The pool consists of networks, servers, storage, applications and services.



AWS Essentials

AWS Introduction	AWS Global Infrastructure	AWS Architecture
AWS Account Creation	AWS Products & Categories	AWS Essential - General

AWS Introduction

What is Amazon Web Services? What are its benefits?

- Amazon Web Services(AWS) are a collection of remote services (Also called as web service) offered by the amazon.com over the internet to build and run an application.
- 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 customers are currently leveraging AWS cloud products and solutions to build sophisticated applications with increased flexibility, scalability and reliability.
- Amazon Web Services (AWS) is robust, scalable and affordable infrastructure for cloud computing
- AWS is Elasticity: scale up or scale down as needed,
- We can get recourses instantly & AWS is fully on demand

The benefits are: -

Pay-per use model

You are only charged for disk space, CPU time and bandwidth that you use.

Instant scalability

Your Service automatically scales on AWS stack.

Reliable/Redundant

Data is redundant in the cloud. All services have built-in security

Security

AWS delivers a scalable cloud-computing platform that provides customers with end to-end security and end-to-end privacy.

Most services accessed via simple REST/SOAP API Libraries are available in all major languages.

Service Level Agreement

SLA between 99.99 and 100% availability

Amazon S3 maintains a durability of 99.99999%

Availability

Availability Zones exist on isolated fault lines, flood plains, and electrical grids to

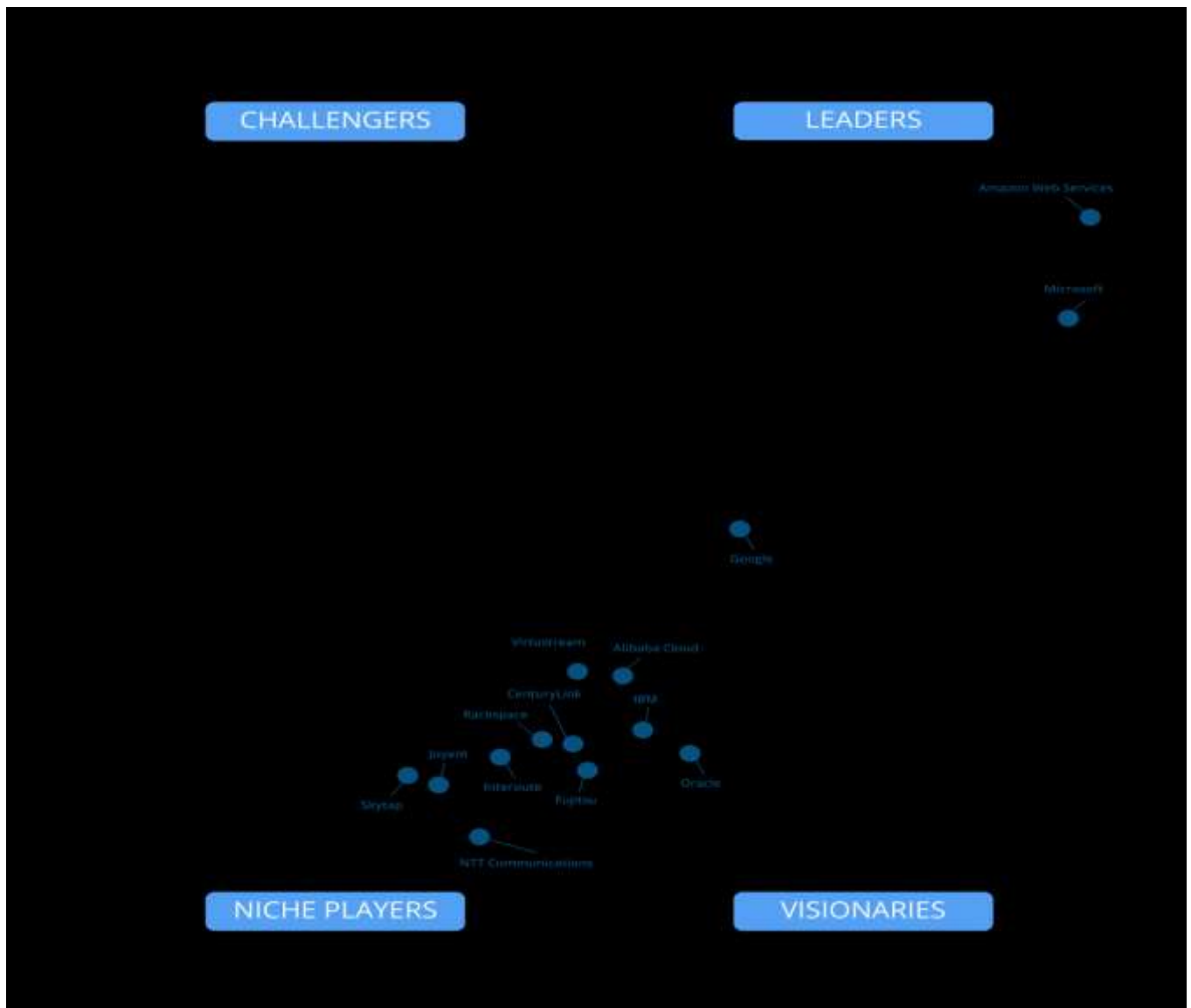
Substantially reduce the chance of simultaneous failure.

Support

AWS provides 24/7 support in the real-time operational status of all services around the globe

Why Amazon Web Services?

- Fastest growing cloud computing platform on the planet
- Largest public cloud computing platform on the planet
- More and More organizations are outsourcing their IT to AWS
- The AWS certifications are the most popular IT certifications right now
- Top Paid certification according to Forbes
- AWS was named as a leader in the “IaaS Magic Quadrant” for the 7th Consecutive Year – Gartner



AWS Global Infrastructure

What are the Amazon Web Services Global Infrastructure?

The AWS Cloud spans 54 Availability Zones within 18 geographic Regions around the world.

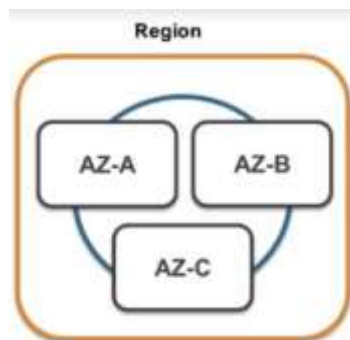
The following table list the AWS region name and code: -

Region Name	Region Code	Region Name	Region Code
US East (N. Virginia)	us-east-1	Asia Pacific (Mumbai)	ap-south-1
US East (Ohio)	us-east-2	Asia Pacific (Seoul)	ap-northeast-2
US West (N. California)	us-west-1	Asia Pacific (Singapore)	ap-southeast-1
US West (Oregon)	us-west-2	Asia Pacific (Sydney)	ap-southeast-2
Canada (Central)	ca-central-1	Asia Pacific (Tokyo)	ap-northeast-1
South America (São Paulo)	sa-east-1	EU (Frankfurt)	eu-central-1
China (Beijing)	cn-north-1	EU (Ireland)	eu-west-1
AWS GoVCloud	us-gov-west-1	EU (London)	eu-west-2

To know the latest information about AWS global infrastructure

Please Visit Website: <https://aws.amazon.com/about-aws/global-infrastructure/>

The AWS Cloud infrastructure is built around Regions and Availability Zones (“AZs”).



Regions

- A Region is a physical location in the world where we have multiple Availability Zones. A grouping of AWS data centers within a specific region. Designed to be independent of other regions.
- AWS Regions are completely isolated from each other and are in different parts of the world and AWS Regions is
 - A collection of data centers (Availability Zones or “AZ”)
 - Each region has a set number of AZs
 - All AZs in a region connected by high-bandwidth
 - Cost vary from Region to Region
 - Default Region in US East

- An AWS Region is a completely independent entity in a geographical area. There are two more Availability Zones in an AWS Region. Within a region, Availability Zones are connected through low-latency links. Since each AWS Region is isolated from another Region, it provides very high fault tolerance and stability. For launching an EC2 instance, we have to select an AMI within the same region.

Availability Zones

- An Availability Zone is
 - Subset of a Region
 - Physically isolated & independent infrastructure
 - High speed connectivity
 - Low latency
 - Every Region has a minimum of 2 AZs
- An Availability Zone consist of one or more discrete data centers, each with redundant power, networking and connectivity, housed in separate facilities. These Availability Zones offer you the ability to operate production applications and databases which are more highly available, fault tolerant and scalable than would be possible from a single data center
- Minimum two availability zones in a single region for high availability of AWS
- Individual datacenter within an AWS region. A region is made up of multiple datacenters and the fundamental property of AWS is building across the different availability zones and regions.
- Hosting across the regions based on the business promotion

Edge Locations

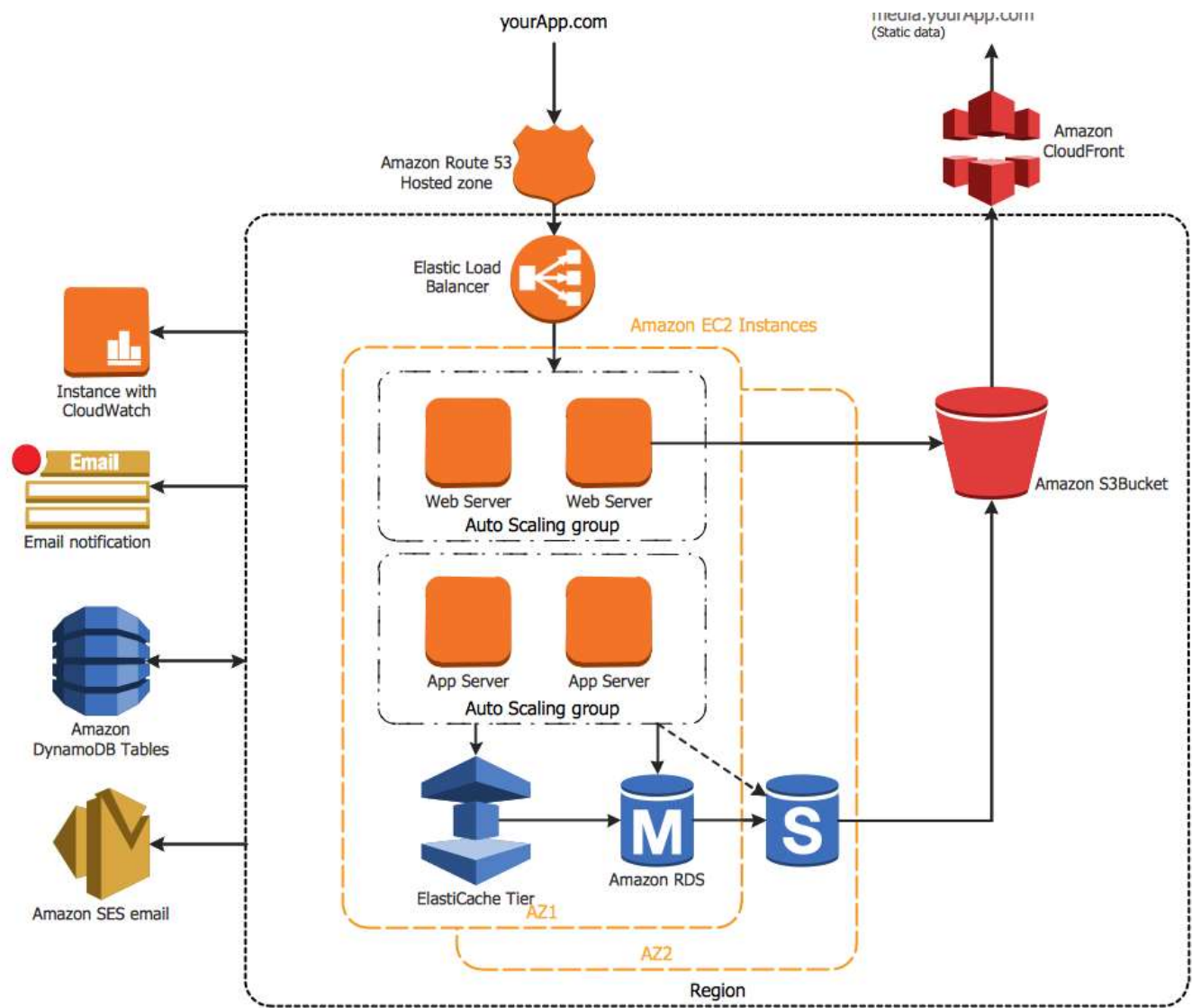
- Edge Locations are endpoints for AWS which are used for caching content. Typically, this consists of CloudFront, Content Delivery Network (CDN). There are many more edge locations than regions. Currently there are over 100 edge locations. Based on the nearest edge location, customers can communicate and get data
- Locations built to deliver cached data across the world. CloudFront CDN utilizes this service for faster delivery to countries without AWS regions.

Edge Cache

- Edge Cache is used to store my frequently accessed data in the server

AWS Architecture

Explain Amazon Web Services Architecture Diagram?



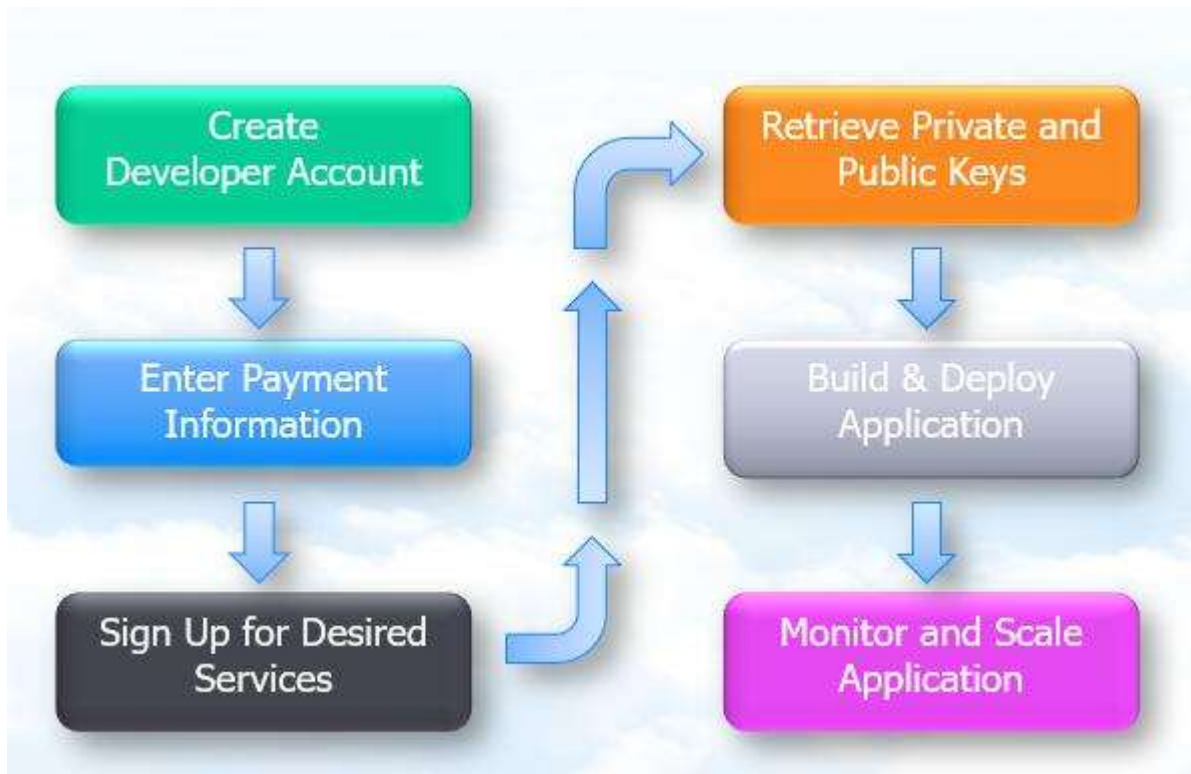
AWS Account Creation

Create an Amazon Web Services account?

Amazon Web Services (AWS) is a cloud computing platform with a bunch of services. The most popular of these services are Amazon EC2 (provides resizable compute capacity in the cloud), Amazon EBS (provides block storage volumes for use with Amazon EC2 instances) and Amazon S3 (provides service for storing and retrieving data, at any time, from anywhere on the web).

The really great thing that new AWS customers are able to try all these services for free - [AWS introducing a free usage tier](#). The AWS sign-up process is pretty straightforward and fully automated. The process may take from several minutes to half an hour

Account Creation Steps



The following steps illustrate how you can easily setup AWS account:

1. Open Amazon Web Services site and select "Create an AWS Account" at the top right corner of the page.
2. Fill in the form with your email address and select "I am a new user" option.

My e-mail address is:

☒ **I am a new user.**

☐ **I am a returning user
and my password is:**

3. In the login credentials form type your name, email address and password once more time and click Continue button.

My name is:

My e-mail address is:

Type it again:

note: this is the e-mail address that we
will use to contact you about your
account

Enter a new password:

Type it again:

4. Fill in the contact information form and click Create Account and Continue

Contact Information

* required fields

Full Name*:

Company Name:

Country*:

Address Line 1*:
Street address, P.O. box, company name, c/o

Address Line 2:
Apartment, suite, unit, building, floor, etc.


City*:

State, Province or Region*:

ZIP or Postal Code*:

Phone number*:

Security Check

Image: 

[Try a different image](#) [Why do we ask you to type these characters?](#)

Type the characters in the above image*:

[Having Trouble? Contact us.](#)

AWS Customer Agreement



Check here to indicate that you have read and agree to the terms of the [Amazon Web Services Customer Agreement](#).

Create Account and Continue

5. Fill in the payment form with information from your credit card.
6. Fill in the identity verification form. You will receive automation message on your phone and you have to type in a PIN displayed in the web browser.
7. Additionally Amazon support may call you to verify your credit card number, billing address and personal information.
8. Confirmation message sent on your mail box will be final step of a registration process.

Amazon Web Services no-reply-aws@amazon.com



Greetings from Amazon Web Services,

Thank you for signing up. You can now begin using Amazon Web Services. You will not be charged until you begin using the services--and you will only pay for what you use. [View detailed service pricing.](#)

Now you can navigate to the manage account section and verify status of services.

Manage Your Account Welcome | [Sign Out](#)
Account Number

Services You're Signed Up For

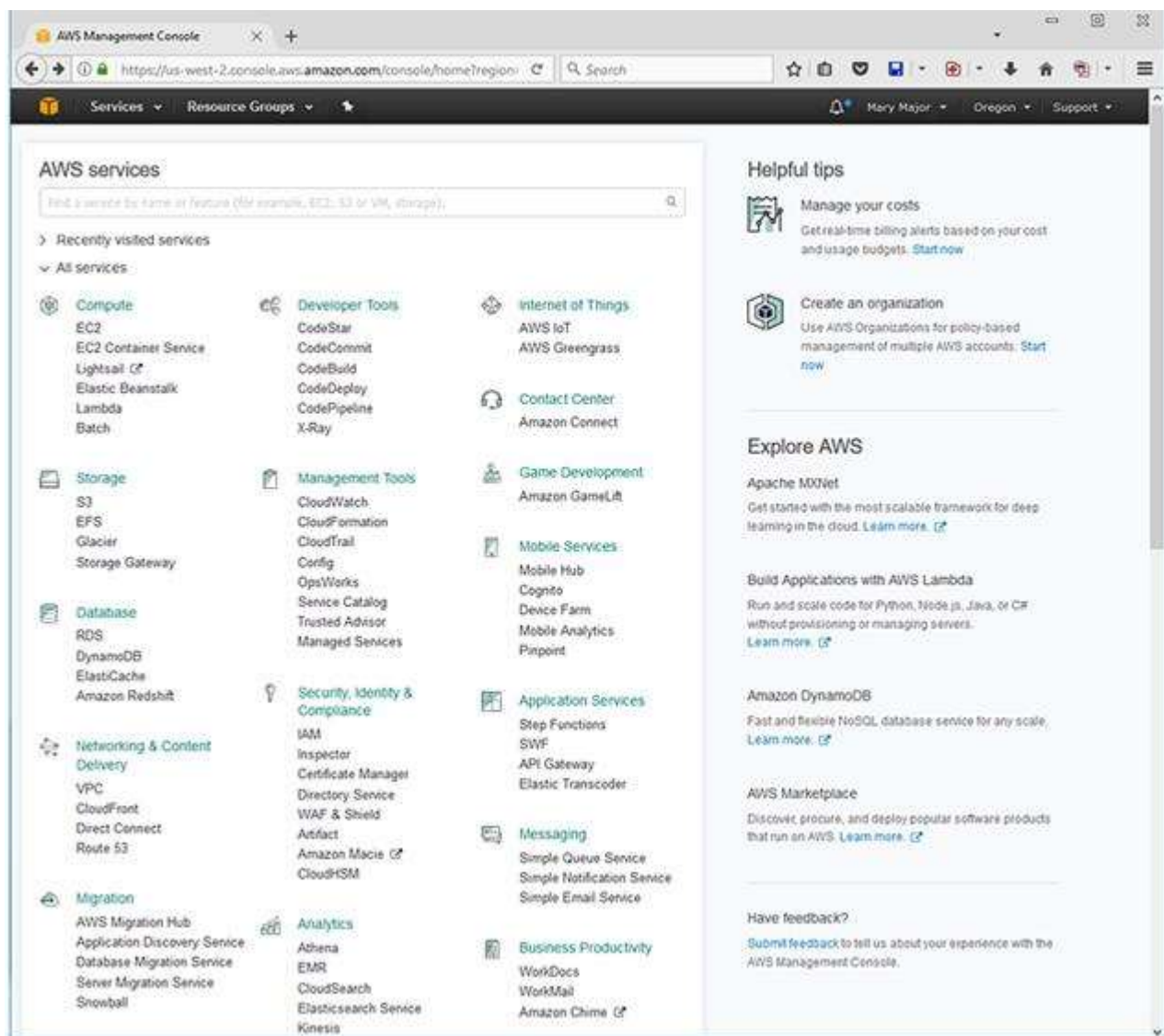
Amazon CloudFormation	Amazon Simple Queue Service (SQS)
Amazon CloudFront	Amazon Simple Storage Service (S3)
Amazon CloudSearch	Amazon Simple Workflow Service (SWF)
Amazon CloudWatch	Amazon SimpleDB
Amazon DynamoDB	Amazon Virtual Private Cloud (VPC)
Amazon Elastic Compute Cloud (EC2)	Auto Scaling
Amazon Elastic MapReduce	AWS Elastic Beanstalk
Amazon ElastiCache	AWS Import/Export
Amazon Mechanical Turk	AWS Storage Gateway
Amazon Relational Database Service (RDS)	Elastic Block Store (EBS)
Amazon Route 53	Elastic Load Balancing
Amazon Simple Email Service (SES)	Product Advertising API
Amazon Simple Notification Service (SNS)	

Using menu from the top right corner you can navigate to the [AWS Management Console](#) or check account status.

What Is the AWS Management Console?

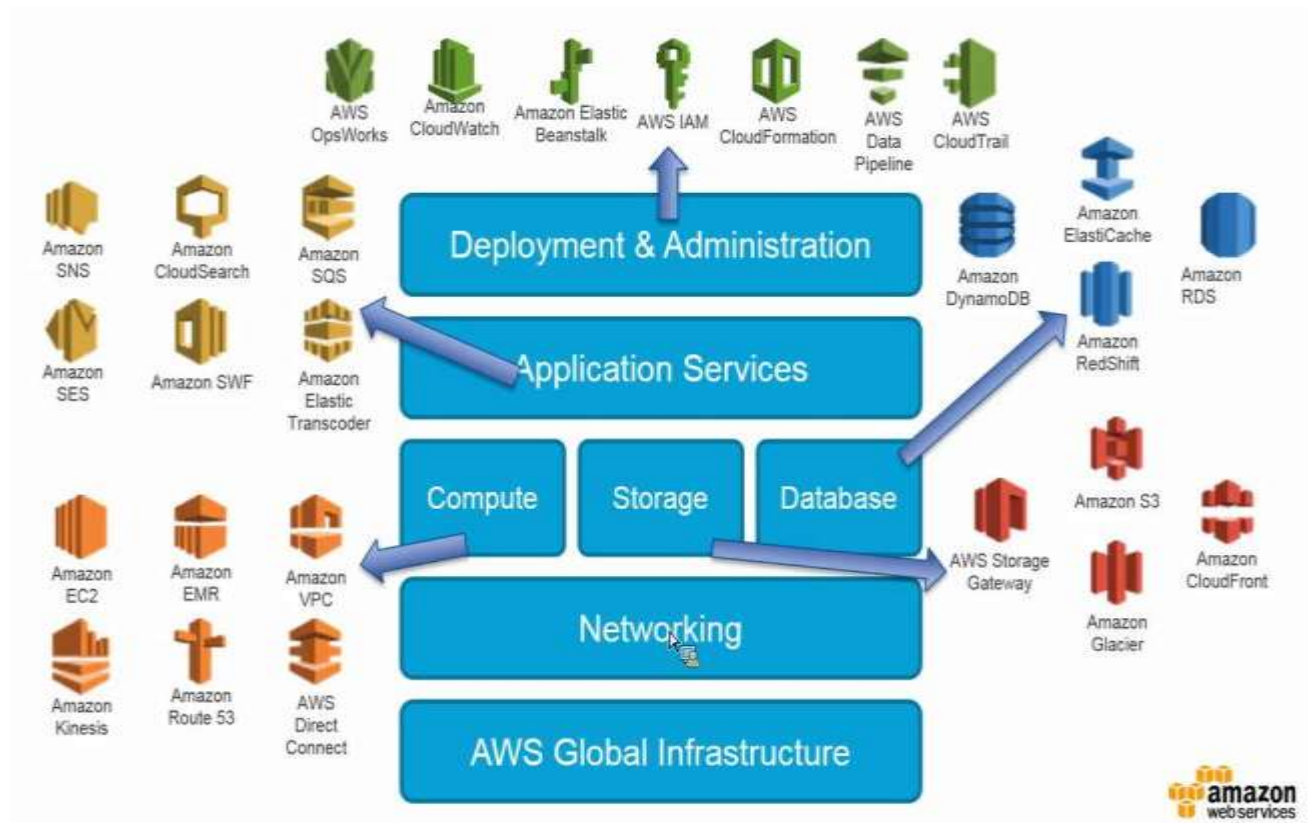
The **AWS Management Console** is a web application that comprises and refers to a broad collection of service consoles for managing Amazon Web Services. When you first sign in, you see the console home page.

The home page provides access to each service console as well as an intuitive user interface for exploring AWS and getting helpful tips. Among other things, the individual service consoles offer tools for working with **Amazon S3** buckets, launching and connecting to **Amazon EC2** instances, setting **Amazon CloudWatch** alarms, and getting information about your account and about **billing**.

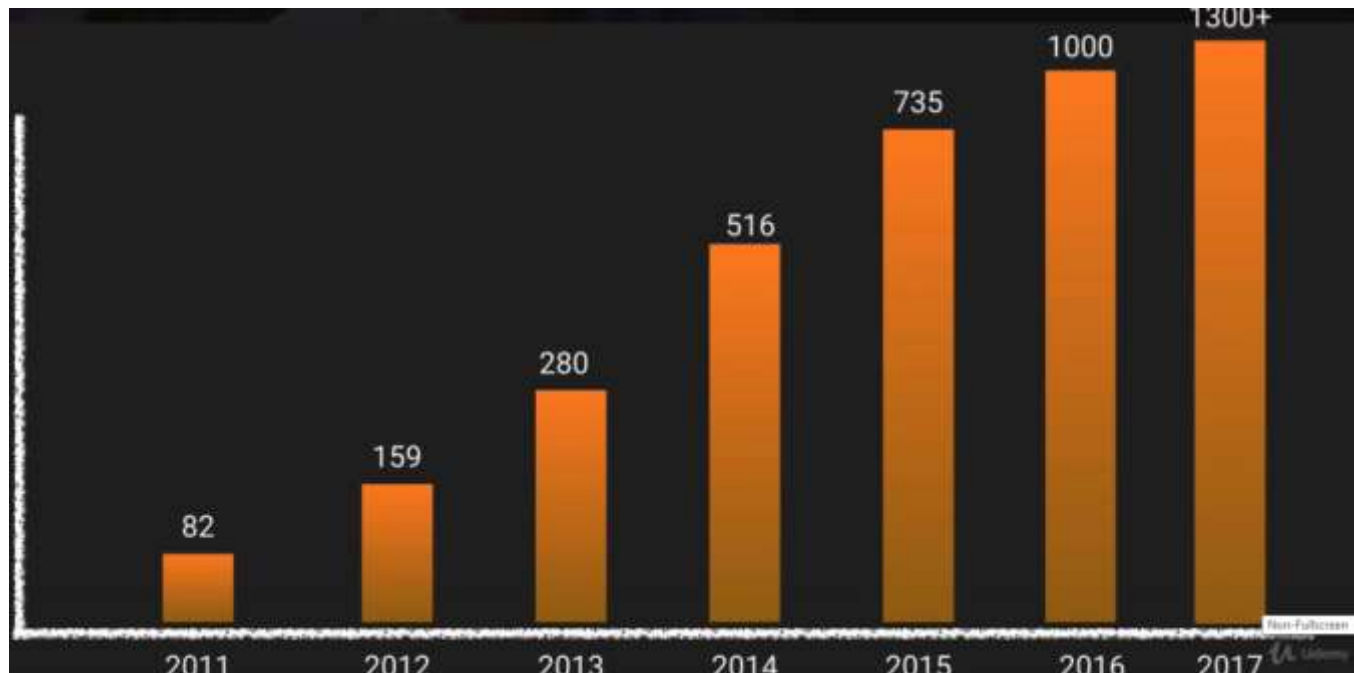


AWS Products & Services

Share the high-level view of AWS Products & Services?

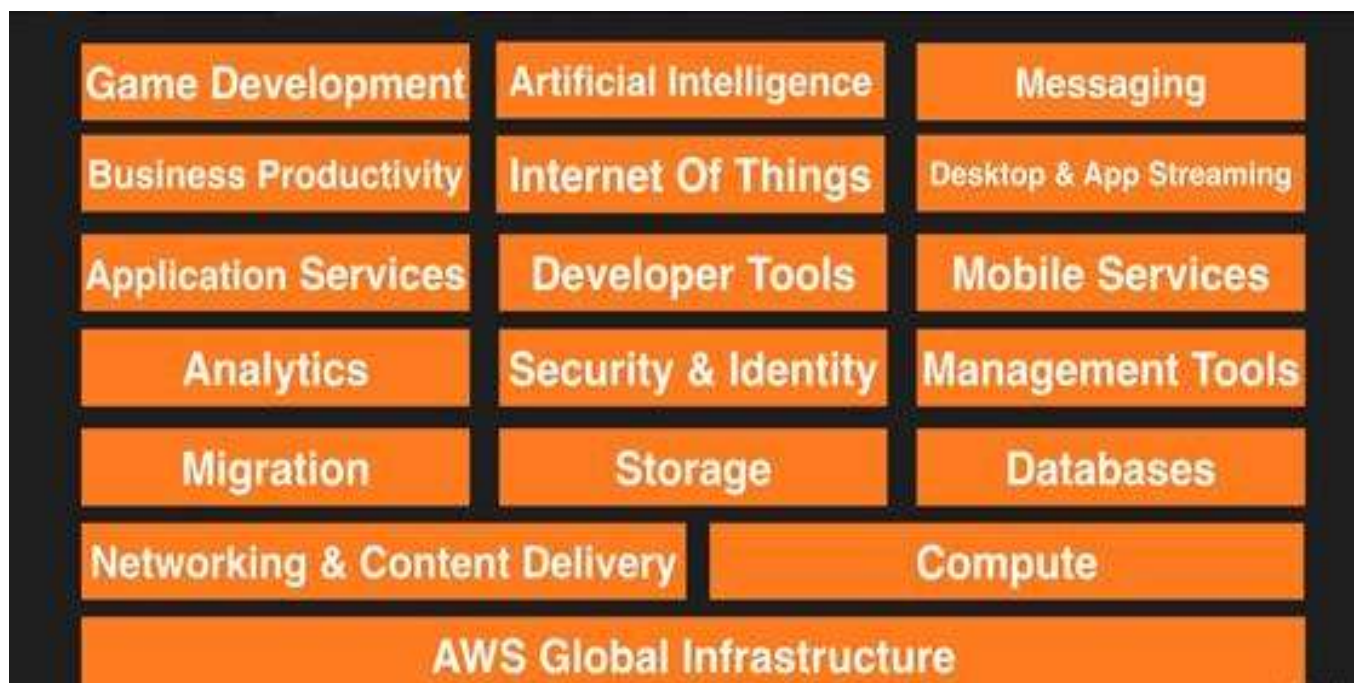


Share the Service Announcements in Amazon Web Services?



Explain Amazon Web Services Product Categories?

Amazon Web Services offers broad set of product categories in AWS platform are –



Compute

It is used to process data on the cloud by making use of powerful processors which serve multiple instances at a time.

Storage

The storage as the name suggests, is used to store data in the cloud, this data can be stored anywhere but content delivery on the other hand is used to cache data nearer to the user so as to provide low latency.

Database

The database domain is used to provide reliable relational and non-relational database instances managed by AWS.

Networking and Content Delivery

It includes services which provide a variety of networking features such as security, faster access etc.

Management Tools

It includes services which can be used to manage and monitor your AWS instances.

Security and Identity

It includes services for user authentication or limiting access to a certain set of audience on your AWS resources.

Application Services

It includes simple services like notifications, emailing and queuing.

Compute

EC2

Amazon Elastic Compute Cloud (EC2) provides resizable compute capacity in the cloud. Provides the virtual server in the AWS Cloud.

EC2 Container Service

Amazon ECS allows you to easily run and manage Docker containers across a cluster of Amazon EC2 instances.

Elastic Beanstalk

AWS Elastic Beanstalk (EBS) is an application container for deploying and managing applications.

Lambda

AWS Lambda is a compute service that runs your code in response to events and automatically manages the compute resources for you.

Elastic Load Balancing

Distributes network traffic across your set of Virtual Servers.

LightSail

Amazon LightSail is the easiest way to get started with AWS for developers who just need virtual private servers. LightSail includes everything you need to launch your project quickly – a virtual machine, SSD-based storage, data transfer, DNS management, and a static IP – for a low, predictable price.

Storage

S3

Amazon Simple Storage Service (S3) can be used to store and retrieve any amount of data.

Glacier

Amazon Glacier is a low-cost storage service that provides secure and durable storage for data archiving and backup.

EFS

Amazon Elastic File System (Amazon EFS) is a file storage service for Amazon Elastic Compute Cloud (Amazon EC2) instances.

Storage Gateway

AWS Storage Gateway securely integrates on-premises IT environments with cloud storage for backup and disaster recovery.

Databases

RDS

Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale familiar relational databases in the cloud.

Redshift

Amazon Redshift is a fast, fully managed, peta byte- scale data warehouse that makes it cost-effective to analyze all your data using your existing business intelligence tools.

DynamoDB

Amazon DynamoDB is a scalable NoSQL data store that manages distributed replicas of your data for high availability.

Elasticache

Amazon ElastiCache improves application performance by allowing you to retrieve information from an in-memory caching system.

Networking & Content Delivery

VPC

Amazon Virtual Private Cloud (VPC) lets you launch AWS resources in a private, isolated cloud.

Route 53

Amazon Route 53 is a scalable and highly available Domain Name System (DNS) and Domain Name Registration service.

Direct Connect

AWS Direct Connect lets you establish a dedicated network connection from your network to AWS.

CloudFront

Amazon CloudFront provides a way to distribute content to end users with low latency and high data transfer speeds.

Migration

Snowball

Snowball is a petabyte-scale data transport solution that uses secure appliances to transfer large amounts of data into and out of the AWS cloud.

Database Migration Service

AWS Database Migration Service helps you migrate databases to AWS quickly and securely. The source database remains fully operational during the migration, minimizing downtime to applications that rely on the database.

Server Migration Service (SMS)

AWS Server Migration Service (SMS) is an agentless service which makes it easier and faster for you to migrate thousands of on-premises workloads to AWS. AWS SMS allows you to automate, schedule, and track incremental replications of live server volumes, making it easier for you to coordinate large-scale server migrations.

Developer Tools

CodeCommit

AWS CodeCommit is a highly scalable, managed source control service that hosts private Git repositories.

CodeDeploy

AWS CodeDeploy lets you fully automate code deployments.

CodePipeline

AWS CodePipeline is a continuous delivery service that enables you to model, visualize, and automate the steps required to release your software.

Code Build

AWS CodeBuild tool is a fully managed build service that compiles source code, runs tests, and produces software packages that are ready to deploy. With CodeBuild, you don't need to provision, manage, and scale your own build servers.

Management Tools

CloudWatch

Amazon CloudWatch provides monitoring for resources and applications.

CloudFormation

AWS CloudFormation lets you create and update a collection of related AWS resources in a predictable fashion.

CloudTrail

AWS CloudTrail provides increased visibility into user activity by recording API calls made on your account.

OpsWorks

AWS OpsWorks is a DevOps platform for managing applications of any scale or complexity on the AWS cloud.

Service Catalog

AWS Service Catalog allows organizations to manage approved catalogs of IT resources and make them available to employees via a personalized portal.

Config

AWS Config gives you inventory of your AWS resources, lets you audit resource configuration history, and notifies you when resource configurations change.

Trusted Advisor

AWS Trusted Advisor inspects your AWS environment and finds opportunities to save money, improve system performance and reliability, or help close security gaps.

Security & Identity

IAM

AWS Identity and Access Management (IAM) lets you securely control access to AWS services and resources.

Inspector

Amazon Inspector enables you to analyze the behavior of the applications you run in AWS and helps you to identify potential security issues.

Certificate Manager

AWS Certificate Manager lets you easily provision, manage, and deploy Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates for use with AWS services.

Directory Service

AWS Directory Service provides managed directories in the cloud.

WAF

AWS WAF (Web Application Firewall) protects web applications from attack by providing web traffic filtering against common web exploits like SQL injection.

Artifacts

AWS Artifact provides on-demand access to AWS' security and compliance reports and select online agreements.

Analytics

Athena

Amazon Athena is an ETL-like service launched in November 2016. It allows server-less querying of S3 content using standard SQL.

Kinesis

Amazon Kinesis is a cloud-based service for real-time data processing over large, distributed data streams. It streams data in real time with the ability to process thousands of data streams on a per-second basis. The service, designed for real-time apps, allows developers to pull any amount of data, from any number of sources, scaling up or down as needed. It has some similarities in functionality to Apache Kafka.

EMR

Amazon Elastic MapReduce (EMR) Provides a PaaS service delivering Hadoop for running MapReduce queries framework running on the web-scale infrastructure of EC2 and Amazon S3.

Cloud Search

Amazon CloudSearch is a managed service in the AWS Cloud that makes it simple and cost-effective to set up, manage, and scale a search solution for your website or application.

Data Pipeline

AWS Data Pipeline provides reliable service for data transfer between different AWS compute and storage services (e.g., Amazon S3, Amazon RDS, Amazon DynamoDB, Amazon EMR). In other words, this service is simply a data-driven workload management system, which provides a management API for managing and monitoring of data-driven workloads in cloud applications

Elastic Search

Amazon Elasticsearch Service provides fully managed Elasticsearch and Kibana services.

Quick Sight

Amazon QuickSight is a business intelligence, analytics, and visualization tool launched in November 2016. It provides ad-hoc services by connecting to AWS or non-AWS data sources.

Artificial Intelligence

Machine Learning

Amazon Machine Learning is a service that enables you to easily build smart applications.

Lex

Amazon Lex is an AWS service for building conversational interfaces for any applications using voice and text. With Amazon Lex, the same conversational engine that powers Amazon Alexa is now

available to any developer, enabling you to build sophisticated, natural language chatbots into your new and existing applications.

Polly

Amazon Polly is a service that turns text into lifelike speech. Amazon Polly enables existing applications to speak as a first-class feature and creates the opportunity for entirely new categories of speech-enabled products, from mobile apps and cars, to devices and appliances.

Rekognition

Amazon Rekognition is a service that makes it easy to add image analysis to your applications. With Rekognition, you can detect objects, scenes, faces; recognize celebrities; and identify inappropriate content in images. You can also search and compare faces. Rekognition's API enables you to quickly add sophisticated deep learning-based visual search and image classification to your applications.

Internet of Things

IoT

AWS IoT is a managed cloud service that lets connected devices easily and securely interact with cloud applications and other devices.

Mobile Services

Mobile Hub

AWS Mobile Hub lets you quickly build, test, and monitor usage of your mobile apps.

Cognito

Amazon Cognito is a simple user identity and data synchronization service that helps you securely manage and synchronize app data for your users across their mobile devices.

Device Farm

AWS Device Farm helps you improve the quality of your Android, Fire OS, and iOS apps by testing them against real phones and tablets in the AWS Cloud.

Mobile Analytics

Amazon Mobile Analytics is a service that lets you easily collect, visualize, and understand app usage data at scale

Pinpoint

Amazon Pinpoint makes it easy to engage your customers by tracking the ways in which they interact with your applications. You can then use this information to create segments based on customer attributes and behaviors, and to communicate with those customers using the channels they prefer, including email, SMS and mobile push.

Application Services

AppStream

Amazon AppStream lets you stream resource intensive applications and games from the cloud to multiple end-user devices

SWF

Amazon Simple Workflow (SWF) coordinates all of the processing steps within an application.

API Gateway

Amazon API Gateway makes it easy to create, maintain, monitor, and secure APIs at any scale.

Elastic Transcoder

Amazon Elastic Transcoder lets you convert your media files in the cloud easily, at low cost, and at scale.

Step Functions

AWS Step Functions makes it easy to coordinate the components of distributed applications and microservices using visual workflows.

Messaging

SNS

Amazon Simple Notification Service (**SNS**) lets you publish messages to subscribers or other applications.

SQS

Amazon Simple Queue Service (**SQS**) offers a reliable, highly scalable, hosted queue for storing messages.

Customer Engagement

SES

Amazon Simple Email Service (**SES**) enables you to send and receive email.

Business Productivity

WorkDocs

Amazon WorkDocs is a fully managed, secure enterprise storage and sharing service with strong administrative controls and feedback capabilities that improve user productivity.

WorkMail

Amazon WorkMail is a managed email and calendaring service that offers strong security controls and support for existing desktop and mobile clients.

Desktop & App Streaming

WorkSpaces

Amazon WorkSpaces is a fully managed desktop computing service in the cloud.

AppStream

Amazon AppStream lets you stream resource intensive applications and games from the cloud to multiple end-user devices.

What are the key components of AWS (Amazon Web Service)?

The key components of AWS are: -

Route 53: A DNS web service

Simple E-mail Service: It allows sending e-mail using RESTFUL API call or via regular SMTP

Identity and Access Management: It provides enhanced security and identity management for your AWS account

Simple Storage Device or (S3): It is a storage device and the most widely used AWS service

Elastic Compute Cloud (EC2): It provides on-demand computing resources for hosting applications. It is very useful in case of unpredictable workloads

Elastic Block Store (EBS): It provides persistent storage volumes that attach to EC2 to allow you to persist data past the lifespan of a single EC2

CloudWatch: To monitor AWS resources, It allows administrators to view and collect key metrics. Also, one can set a notification alarm in case of trouble.

AWS Essentials – General Questions

How Security is implemented in Amazon Web Services?

AWS provides a secure global infrastructure, plus a range of features that you can use to secure your data in the cloud. The following are highlights:

- Physical access to AWS data centers is strictly controlled, monitored, and audited.
- Access to the AWS network is strictly controlled, monitored, and audited.
- You can manage the security credentials that enable users to access your AWS account using AWS Identity and Access Management (IAM).
- You can create fine-grained permissions to AWS resources and apply them to users or groups of users. You can apply ACL-type permissions on your data and can also use encryption of data at rest.
- You can set up a virtual private cloud (VPC), which is a virtual network that is logically isolated from other virtual networks in the AWS cloud.
- You can control whether the network is directly routable to the Internet. You control and configure the operating system on your virtual server.
- You can set up a security group, which acts as a virtual firewall to control the inbound and outbound traffic for your virtual servers.
- You can specify a key pair when you launch your virtual server, which is used to encrypt your login information. When you log in to your virtual server, you must present the private key of the key pair to decrypt the login information.

What are the top 10 reasons to go with Amazon Web Services?

This might easily be the most popular of the top 10 reasons to go with AWS.

1. Pricing
2. Flexibility & Scalability
3. Global Architecture
4. PaaS Offerings
5. Consistency & Reliability
6. Scheduling
7. Customization
8. Recovery
9. Security
10. API

Compare AWS and OpenStack?

Criteria	AWS	OpenStack
License	Amazon proprietary	Open Source
Operating System	Whatever cloud administrator provides	Whatever AMIs provided by AWS

Performing
repeatable operations

Through Templates

Through text files

What is the difference between Region, Availability Zone and Endpoint in AWS?

In AWS, every region is an independent environment. Within a Region there can be multiple Availability Zones. Every Availability Zone is an isolated area. But there are low-latency links that connect one Availability Zone to another within a region.

An endpoint is just an entry point for a web service. It is written in a URL form. E.g. `https://dynamodb.us-east-2.amazonaws.com` is an endpoint for Amazon DynamoDB service. Most of the AWS services offer an option to select a regional endpoint for incoming requests. But many services in AWS do not support regions. E.g. IAM. So, their endpoints do not have a region.



Compute

Amazon EC2 Virtual Servers in the Cloud	Amazon EC2 Auto Scaling Scale Compute Capacity to Meet Demand	AWS Elastic Container Service Run and Manage Docker Containers
Amazon Elastic Container Service for Kubernetes Run Managed Kubernetes on AWS	Amazon Elastic Container Registry Store and Retrieve Docker Images	Amazon LightSail Launch and Manage Virtual Private Servers
AWS Batch Run Batch Jobs at Any Scale	AWS Beanstalk Run and Manage Web Apps	AWS Fargate Run Containers without managing servers on clusters
AWS Lambda Run your code in Response to Events	AWS Serverless Application Repository Discover, Deploy, and Publish Serverless Applications Auto Scaling	VMware Cloud on AWS Build a Hybrid Cloud without Custom Hardware



Compute

Amazon EC2

EC2 Highlights

Amazon Elastic Compute Cloud (EC2) is a web service that provides resizable compute capacity in the cloud. It 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.

The EC2 options are: -

OnDemand – Allows you to pay a fixed rate by hour (or by the second) with no commitment

Reserved – Provide you with a capacity reservation and offer a significant discount on the hourly charge for an instance 1 Year to 3 Year terms.

Spot – Enable you to bid whatever price you want for instance capacity, providing for even greater savings if your applications have flexible start and end times

Dedicated Hosts – Physical EC2 server is dedicated for your use. Dedicated Hosts can help you reduce costs by allowing you to use your existing server-bound software licenses.

The EC2 Instances types are: -

Family	Speciality	Use case
D2	Dense Storage	Fileservers/Data Warehousing/Hadoop
R4	Memory Optimized	Memory Intensive Apps/DBs
M4	General Purpose	Application Servers
C4	Compute Optimized	CPU Intensive Apps/DBs
G2	Graphics Intensive	Video Encoding/ 3D Application Streaming
I2	High Speed Storage	NoSQL DBs, Data Warehousing etc
F1	Field Programmable Gate Array	Hardware acceleration for your code.
T2	Lowest Cost, General Purpose	Web Servers/Small DBs
P2	Graphics/General Purpose GPU	Machine Learning, Bit Coin Mining etc
X1	Memory Optimized	SAP HANA/Apache Spark etc

For easy remember, **DR MC GIFT PX**

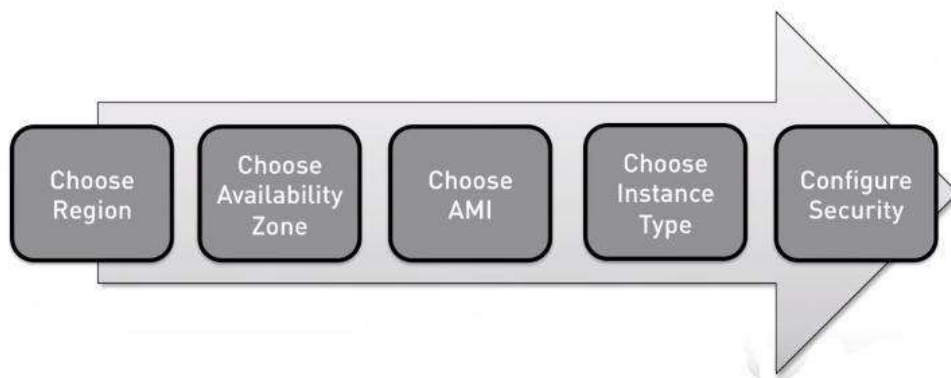
D- Density **R**-Ram

M- Main choice for general purpose apps **C**-Compute

G-Graphics **I**-IOPS **F**- FPGA **T**-Cheap general purpose (Think T2 Micro)

P-Graphics (think Pics) **X**-Extreme Memory

Share the Elastic Compute Cloud Configuration Step by Step?



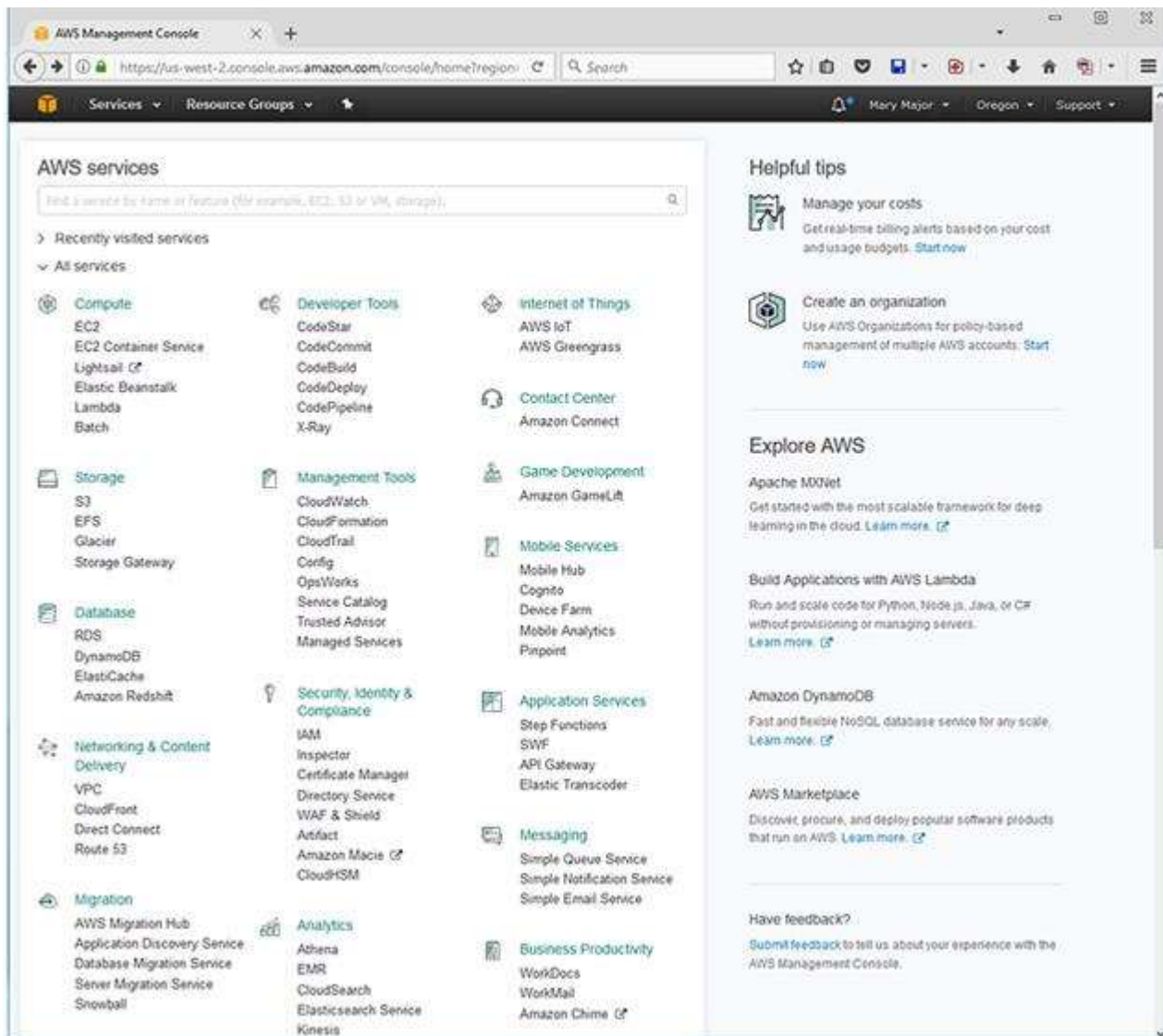
Sign up for AWS at <http://aws.amazon.com>

- Apply the service credit you received by email
- Create and download a Key-Pair, save it in your home directory
- Create a VM via the AWS Console
- Connect to your newly-created VM like this:

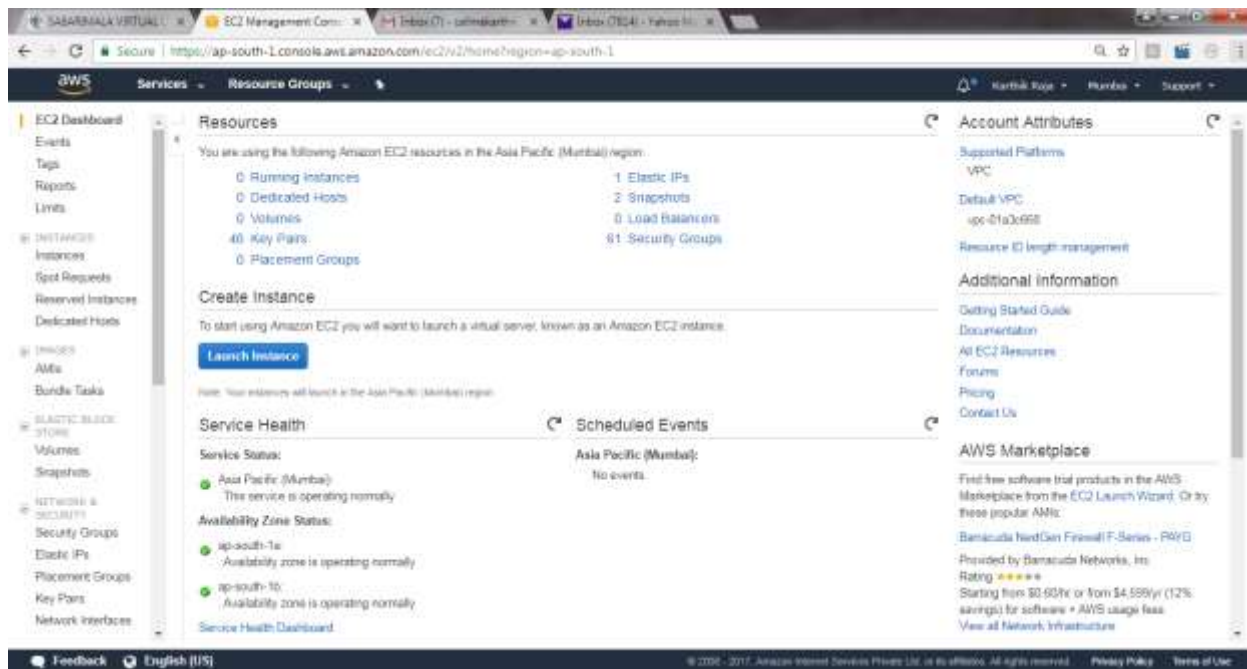
```
ssh -i my-aws-keypair.pem ec2-user@ipaddress-of-vm
```

Login to the Console

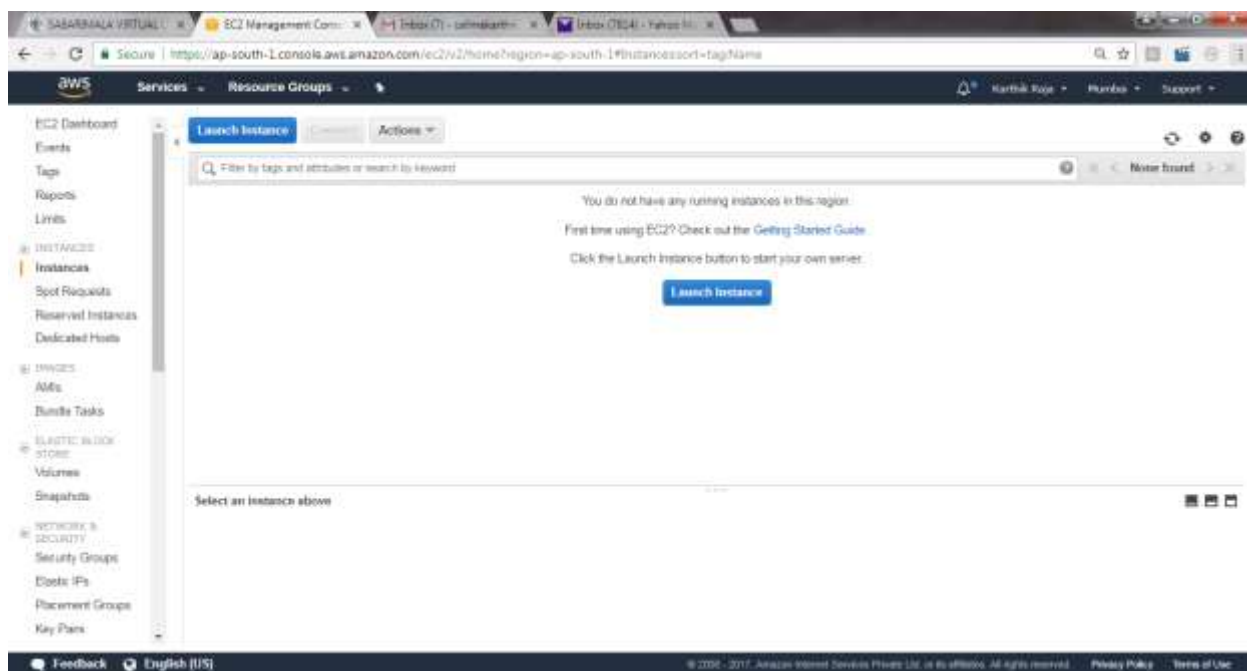
Login into the console and choose EC2 from the services under the compute list,



Identify the EC2 Dashboard



By clicking the “Launch Instance” button to launch the EC2 Instance



Choose an Amazon Machine Image (AMI)

The screenshot shows the AWS Management Console interface for the 'Launch Instance Wizard'. The browser address bar indicates the URL: <https://ap-south-1.console.aws.amazon.com/ec2/v2/home?region=ap-south-1#LaunchInstanceWizard>. The navigation bar at the top shows the user 'Karthik Raja' and the region 'Pune'. The wizard progress bar at the top indicates the current step is '1. Choose AMI'.

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**

Image ID	Image Name	Image Description	Root Device Type	Virtualization Type	Architecture
ami-4f56420	Amazon Linux AMI 2017.09.0 (HVM), SSD Volume Type	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.	Root device type: ebs	Virtualization type: hvm	64-bit
ami-e41b615b	Red Hat Enterprise Linux 7.4 (HVM), SSD Volume Type	Red Hat Enterprise Linux version 7.4 (HVM), EBS General Purpose (SSD) Volume Type.	Root device type: ebs	Virtualization type: hvm	64-bit
ami-e310579c	SUSE Linux Enterprise Server 12 SP3 (HVM), SSD Volume Type	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.	Root device type: ebs	Virtualization type: hvm	64-bit
ami-099fa766	Ubuntu Server 16.04 LTS (HVM), SSD Volume Type	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	64-bit

Feedback | English [US] | © 2009 - 2017 Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

This is a duplicate of the screenshot above, showing the same AWS Management Console interface for the 'Step 1: Choose an Amazon Machine Image (AMI)' wizard. It displays the same list of AMIs and the navigation elements of the AWS console.

Choose Instance Type

The screenshot shows the 'Choose Instance Type' step of the AWS Launch Instance Wizard. The breadcrumb trail at the top indicates the current step is '2. Choose Instance Type'. Below the breadcrumb, a progress bar shows the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, 7. Review. The main heading is 'Step 2: Choose an Instance Type'. Below this, a paragraph explains that Amazon EC2 provides a wide selection of instance types optimized for different use cases. A 'Filter by:' section shows 'All instance types' selected. A 'Currently selected:' summary shows 't2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GB memory, EBS only)'. A table lists various instance types with columns for Family, Type, vCPUs, Memory (GiB), Instance Storage (GiB), EBS-Optimized Available, Network Performance, and IPv6 Support. The 't2.micro' instance type is highlighted. At the bottom, there are buttons for 'Cancel', 'Previous', 'Review and Launch', and 'Next: Configure Instance Details'.

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

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more about instance types and how they can meet your computing needs.](#)

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 GB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GiB)	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	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"/>	Platform optimized	m5.xlarge	8	16	FBS only	Yes	High bandwidth	Yes

Cancel Previous **Review and Launch** Next: Configure Instance Details

Feedback English (US) © 2009 - 2017 Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Configure Instance

The screenshot shows the 'Configure Instance Details' step of the AWS Launch Instance Wizard. The breadcrumb trail at the top indicates the current step is '3. Configure Instance'. Below the breadcrumb, a progress bar shows the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, 7. Review. The main heading is 'Step 3: Configure Instance Details'. Below this, a paragraph explains that you can configure the instance to suit your requirements. A 'Number of instances' field is set to '1'. A 'Purchasing option' section has 'Request Spot instances' selected. A 'Network' section has 'vpc-01a3c668 (default)' selected. A 'Subnet' section has 'No preference (default subnet in any Availability Zone)' selected. An 'Auto-assign Public IP' section has 'Use subnet setting (Enable)' selected. An 'IAM role' section has 'None' selected. A 'Shutdown behavior' section has 'Stop' selected. An 'Enable termination protection' section has 'Protect against accidental termination' selected. A 'Monitoring' section has 'Enable CloudWatch detailed monitoring' selected. A 'Tenancy' section has 'Shared' selected. At the bottom, there are buttons for 'Cancel', 'Previous', 'Review and Launch', and 'Next: Add Storage'.

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-01a3c668 (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

Feedback English (US) © 2009 - 2017 Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Add Storage

The screenshot shows the 'Add Storage' step of the AWS EC2 Launch Wizard. The breadcrumb trail at the top indicates the sequence: 1. Choose VPC, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage (current step), 5. Add Tags, 6. Configure Security Group, 7. Review. The main heading is 'Step 4: Add Storage'. Below it, a paragraph explains that the instance will be launched with the following storage device settings and that additional EBS volumes can be attached later. A table lists the storage configuration for the 'Root' volume: Device is '/dev/sda1', Snapshot is 'snap-009c54c0000c30d4', Size is '8 GB', Volume Type is 'General Purpose SSD (GP2)', IOPS is '100 / 3000', Throughput (MB/s) is 'NA', Delete on Termination is checked, and Encrypted is 'Not Encrypted'. An 'Add New Volume' button is present. A blue box contains a note about a free tier for eligible customers. At the bottom, there are 'Cancel', 'Previous', 'Review and Launch', and 'Next: Add Tags' buttons.

1. Choose VPC 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 (GB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-009c54c0000c30d4	8	General Purpose SSD (GP2)	100 / 3000	NA	<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](#)

[Feedback](#) [English \[US\]](#) © 2016 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

Add tags: Name the EC2 to be created.

The screenshot shows the 'Add Tags' step of the AWS EC2 Launch Wizard. The breadcrumb trail at the top indicates the sequence: 1. Choose VPC, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags (current step), 6. Configure Security Group, 7. Review. The main heading is 'Step 5: Add Tags'. Below it, a paragraph explains that a tag consists of a case-sensitive key-value pair and that tags will be applied to all instances and volumes. A table for adding tags has two columns: 'Key' and 'Value'. A tag is added with Key 'Name' and Value 'EC2 Linux Ubuntu Amazon'. There are checkboxes for 'Instances' and 'Volumes', both of which are checked. An 'Add another tag' button is at the bottom left. At the bottom, there are 'Cancel', 'Previous', 'Review and Launch', and 'Next: Configure Security Group' buttons.

1. Choose VPC 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	Value	Instances	Volumes
Name	EC2 Linux Ubuntu Amazon	<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](#)

[Feedback](#) [English \[US\]](#) © 2016 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

Configure Security Group: Keep as default, just add HTTP/HTTPS for application webservices.

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:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin-Desktop
HTTP	TCP	80	Custom 0.0.0.0/0	e.g. SSH for Admin-Desktop
HTTPS	TCP	443	Custom 0.0.0.0/0	e.g. SSH for Admin-Desktop

[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](#)

Review it and launch.

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.

Improve your instances' security. Your security group, launch-wizard-53, 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](#)

Amazon Linux AMI 2017.06.0 (HVM), SSD Volume Type - ami-4fc58426
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 ecosystem includes C++, PHP, MySQL, PostgreSQL, and other packages.
Root Device Type: x86 Virtualization type: hvm

Instance Type [Edit instance type](#)

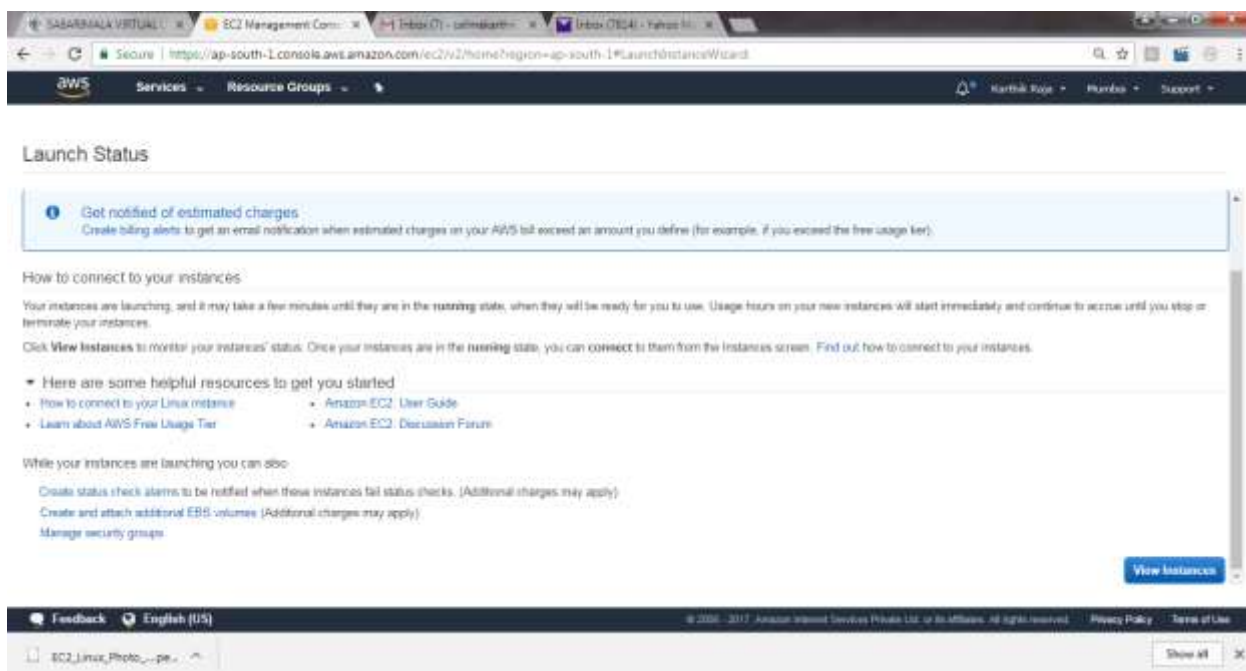
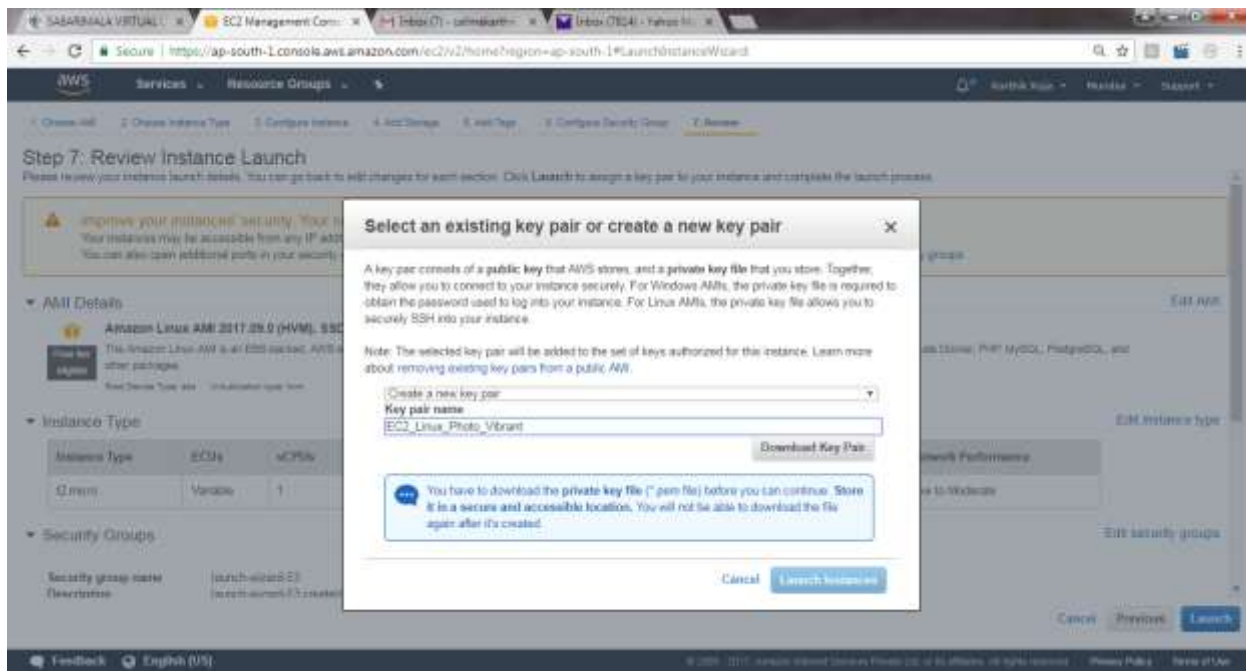
Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GiB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups [Edit security groups](#)

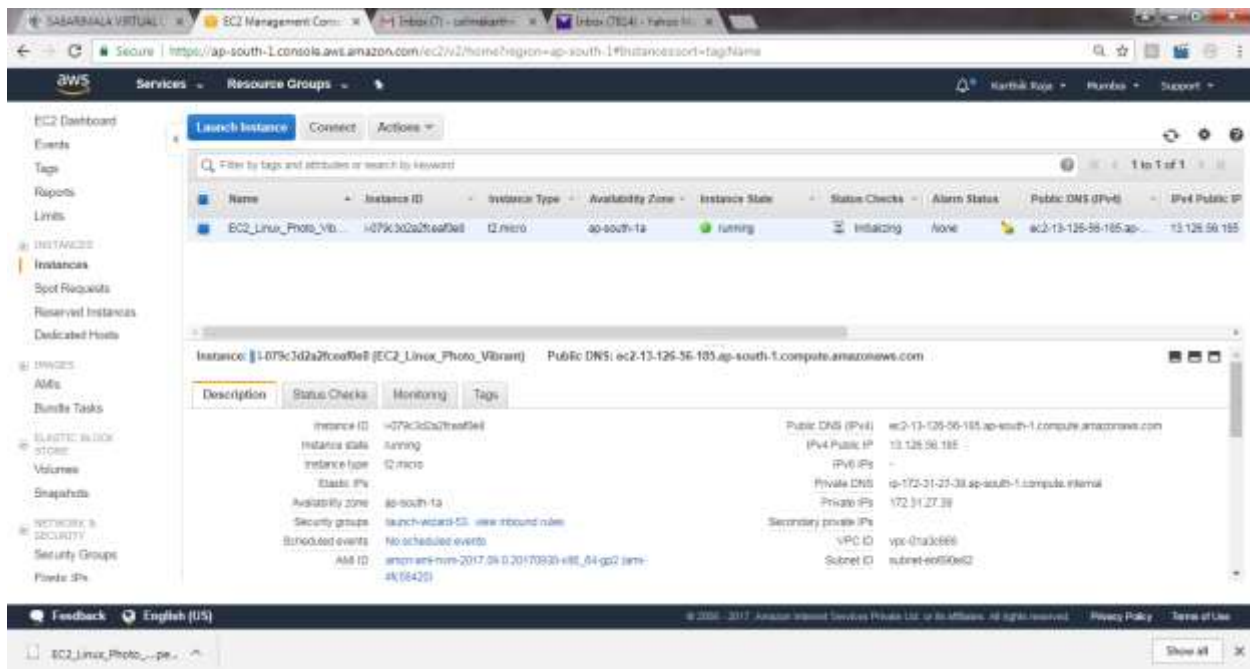
Security group name:
Description:

[Cancel](#) [Previous](#) [Launch](#)

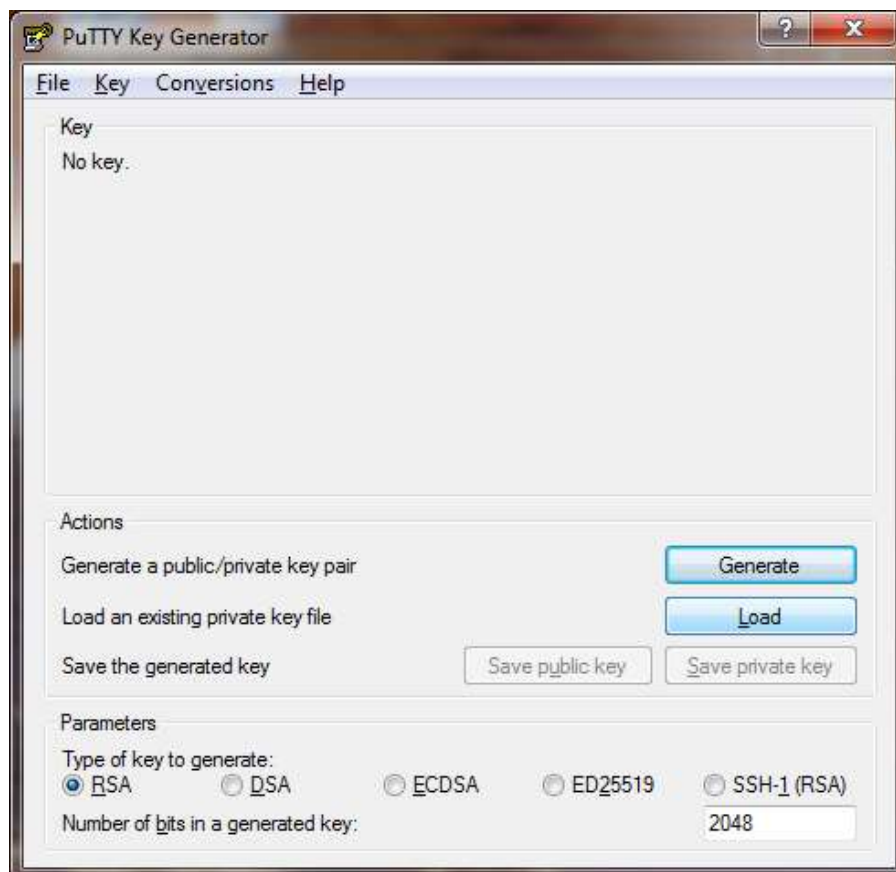
Download the key pair by keeping the same name for the keys as the EC2 name.

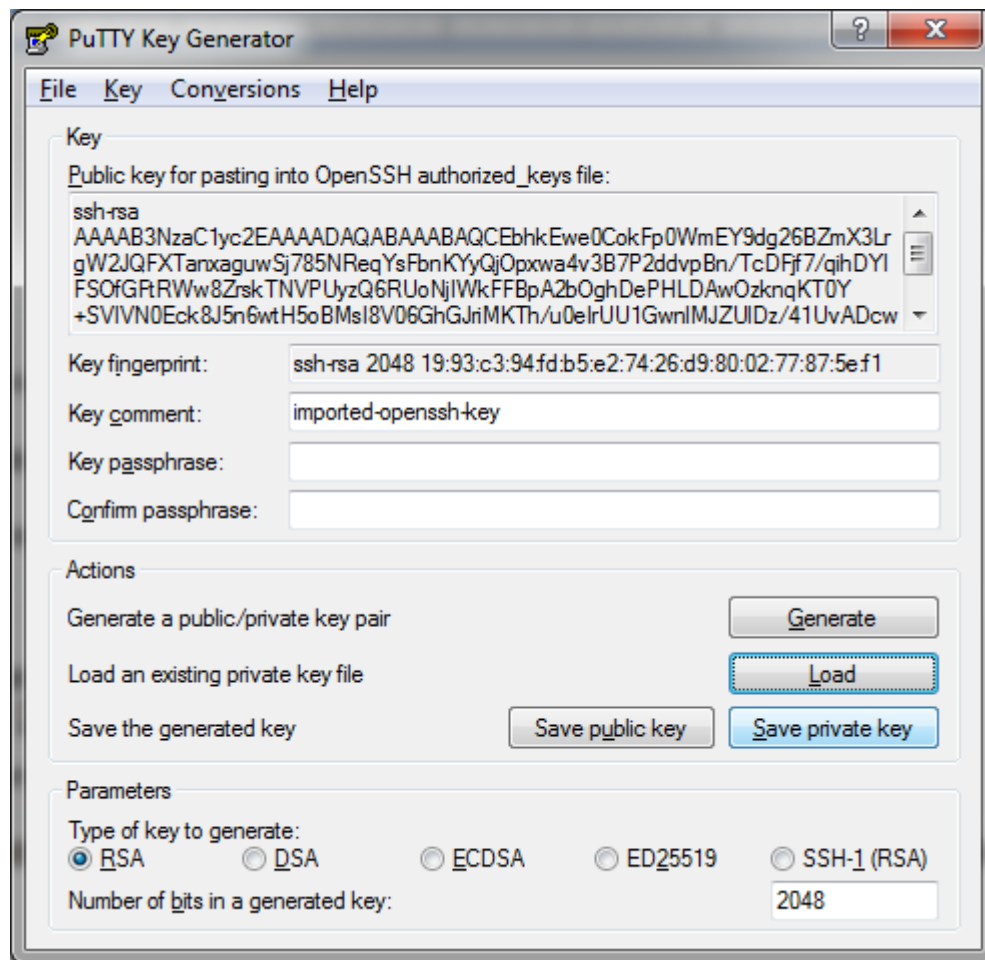


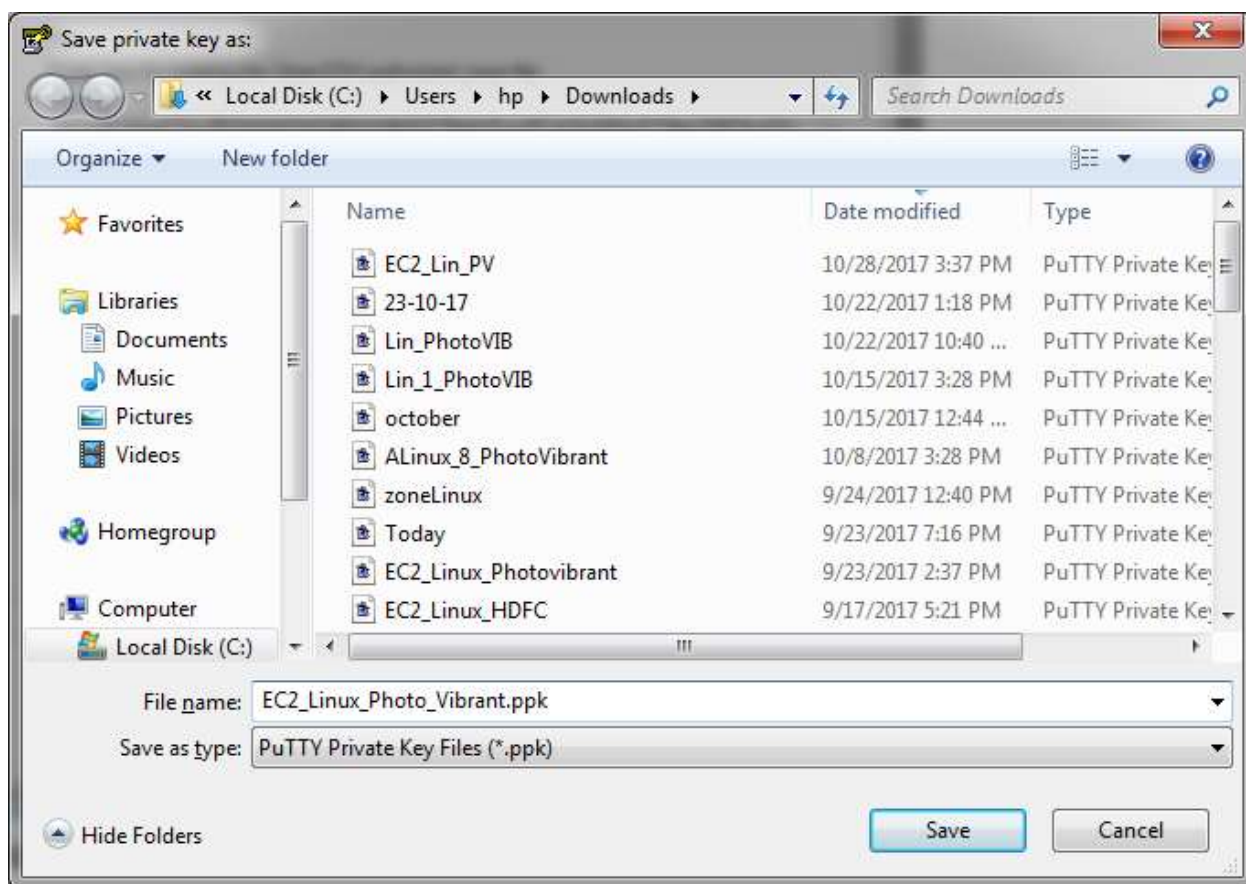
Allow sometime for the instances to launch completely, then pick the IP address, User [ec-user].



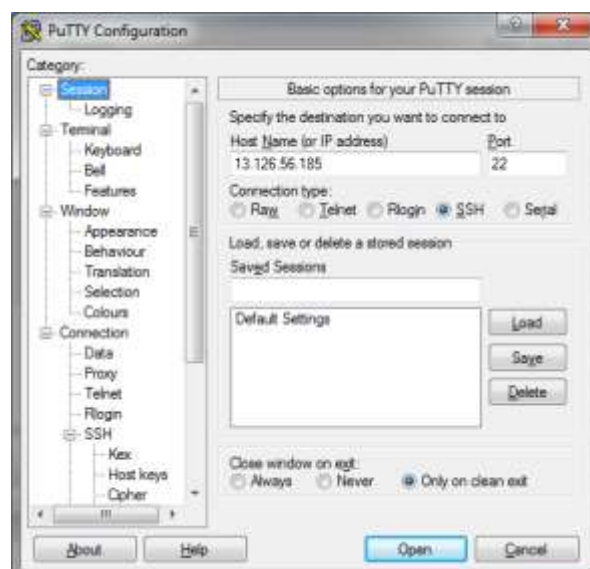
Use the tool – Puttygen to convert the pem file -> .ppk file. Load it and download it as private key.

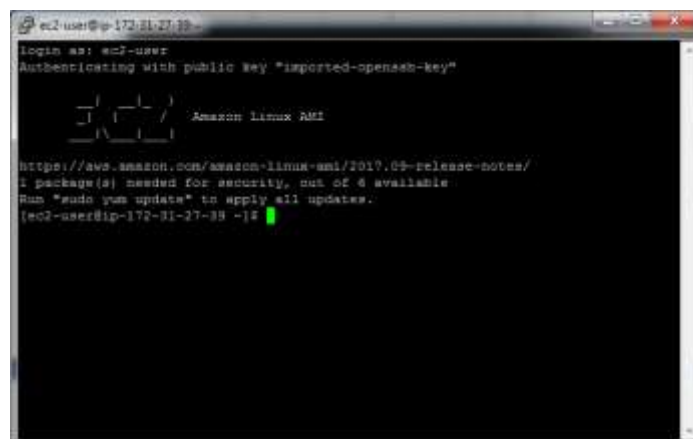
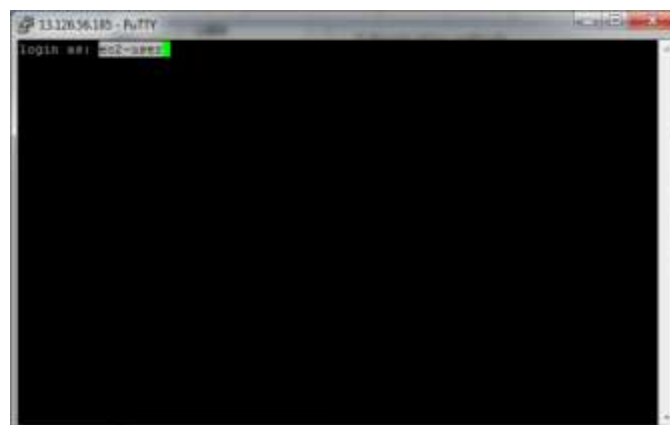
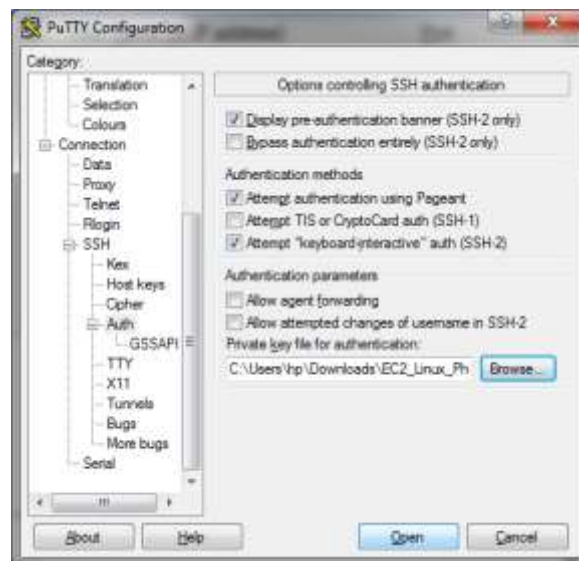




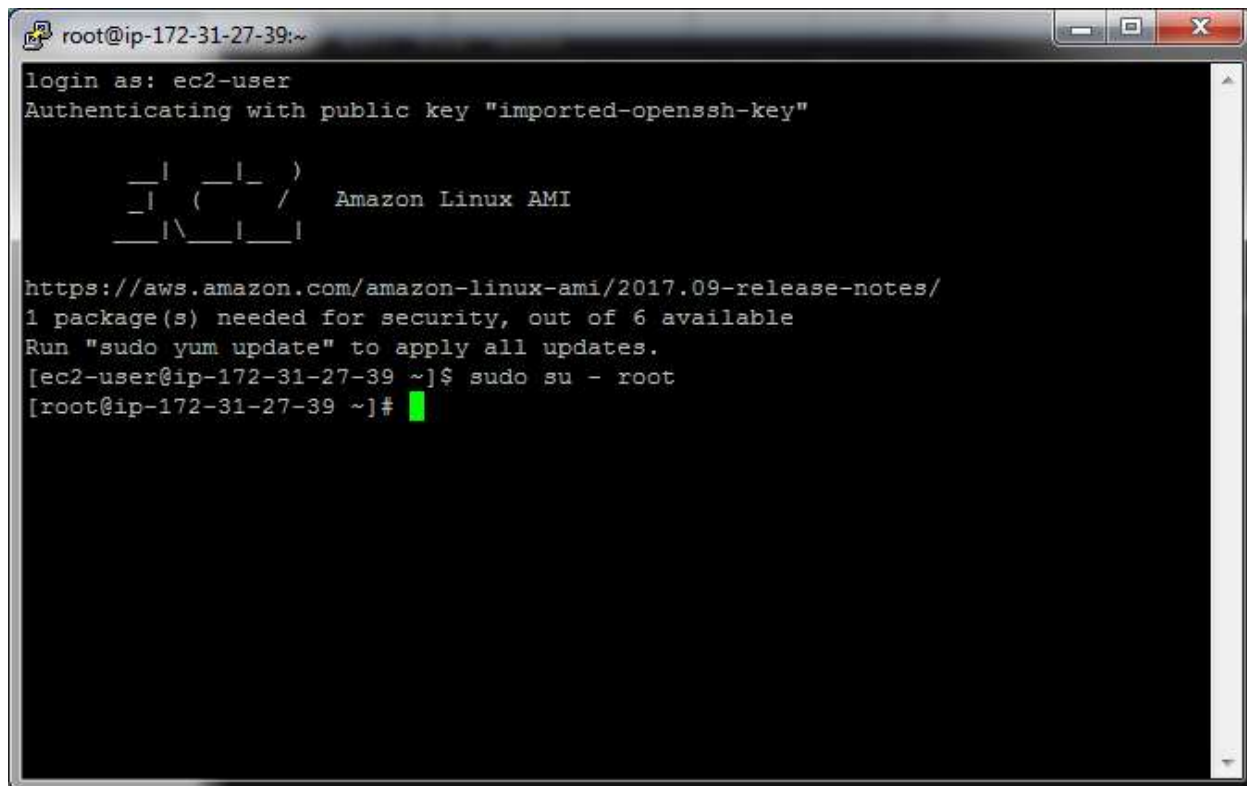


Login into the putty using the ip address, user and the ppk file. Load the .ppk file as below screen shot under ssl -> auth.



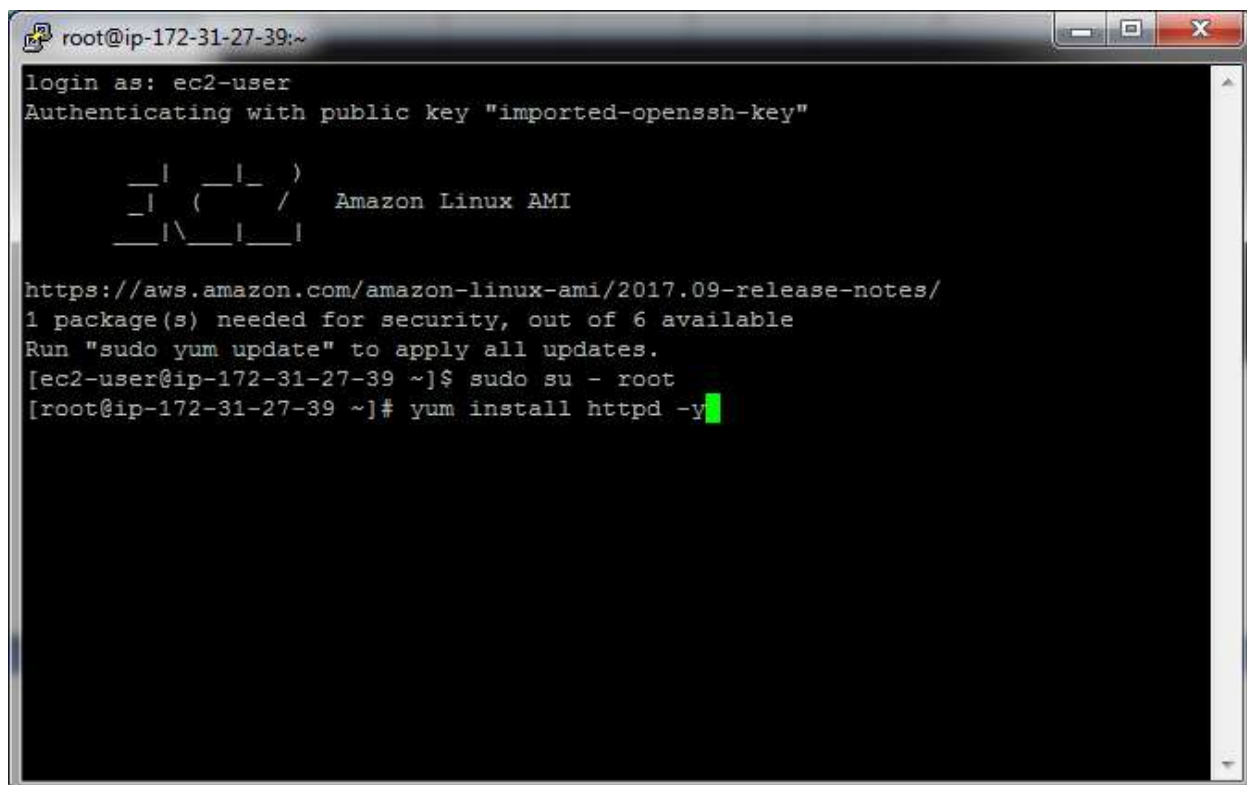


Switch user to **root**, by using below command.



```
root@ip-172-31-27-39:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
  
  _| _|_ )  
 _| ( _| /  Amazon Linux AMI  
__| \__|__|  
  
https://aws.amazon.com/amazon-linux-ami/2017.09-release-notes/  
1 package(s) needed for security, out of 6 available  
Run "sudo yum update" to apply all updates.  
[ec2-user@ip-172-31-27-39 ~]$ sudo su - root  
[root@ip-172-31-27-39 ~]#
```

Install **apache** using the below command.



```
root@ip-172-31-27-39:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
  
  _| _|_ )  
 _| ( _| /  Amazon Linux AMI  
__| \__|__|  
  
https://aws.amazon.com/amazon-linux-ami/2017.09-release-notes/  
1 package(s) needed for security, out of 6 available  
Run "sudo yum update" to apply all updates.  
[ec2-user@ip-172-31-27-39 ~]$ sudo su - root  
[root@ip-172-31-27-39 ~]# yum install httpd -y
```

```
root@ip-172-31-27-39:~  
Transaction test succeeded  
Running transaction  
  Installing : apr-1.5.1-1.12.amzn1.x86_64                1/5  
  Installing : apr-util-1.4.1-4.17.amzn1.x86_64           2/5  
  Installing : httpd-tools-2.2.34-1.15.amzn1.x86_64       3/5  
  Installing : apr-util-ldap-1.4.1-4.17.amzn1.x86_64      4/5  
  Installing : httpd-2.2.34-1.15.amzn1.x86_64             5/5  
  Verifying   : apr-1.5.1-1.12.amzn1.x86_64                1/5  
  Verifying   : httpd-tools-2.2.34-1.15.amzn1.x86_64       2/5  
  Verifying   : apr-util-1.4.1-4.17.amzn1.x86_64           3/5  
  Verifying   : apr-util-ldap-1.4.1-4.17.amzn1.x86_64      4/5  
  Verifying   : httpd-2.2.34-1.15.amzn1.x86_64             5/5  
  
Installed:  
  httpd.x86_64 0:2.2.34-1.15.amzn1  
  
Dependency Installed:  
  apr.x86_64 0:1.5.1-1.12.amzn1  
  apr-util.x86_64 0:1.4.1-4.17.amzn1  
  apr-util-ldap.x86_64 0:1.4.1-4.17.amzn1  
  httpd-tools.x86_64 0:2.2.34-1.15.amzn1  
  
Complete!  
[root@ip-172-31-27-39 ~]#
```

Start the **apache** using the below command and test the same using the DNS url of the EC2 in the browser.

```
root@ip-172-31-27-39:~  
  Installing : apr-1.5.1-1.12.amzn1.x86_64                1/5  
  Installing : apr-util-1.4.1-4.17.amzn1.x86_64           2/5  
  Installing : httpd-tools-2.2.34-1.15.amzn1.x86_64       3/5  
  Installing : apr-util-ldap-1.4.1-4.17.amzn1.x86_64      4/5  
  Installing : httpd-2.2.34-1.15.amzn1.x86_64             5/5  
  Verifying   : apr-1.5.1-1.12.amzn1.x86_64                1/5  
  Verifying   : httpd-tools-2.2.34-1.15.amzn1.x86_64       2/5  
  Verifying   : apr-util-1.4.1-4.17.amzn1.x86_64           3/5  
  Verifying   : apr-util-ldap-1.4.1-4.17.amzn1.x86_64      4/5  
  Verifying   : httpd-2.2.34-1.15.amzn1.x86_64             5/5  
  
Installed:  
  httpd.x86_64 0:2.2.34-1.15.amzn1  
  
Dependency Installed:  
  apr.x86_64 0:1.5.1-1.12.amzn1  
  apr-util.x86_64 0:1.4.1-4.17.amzn1  
  apr-util-ldap.x86_64 0:1.4.1-4.17.amzn1  
  httpd-tools.x86_64 0:2.2.34-1.15.amzn1  
  
Complete!  
[root@ip-172-31-27-39 ~]# service httpd start  
Starting httpd: [ OK ]  
[root@ip-172-31-27-39 ~]#
```

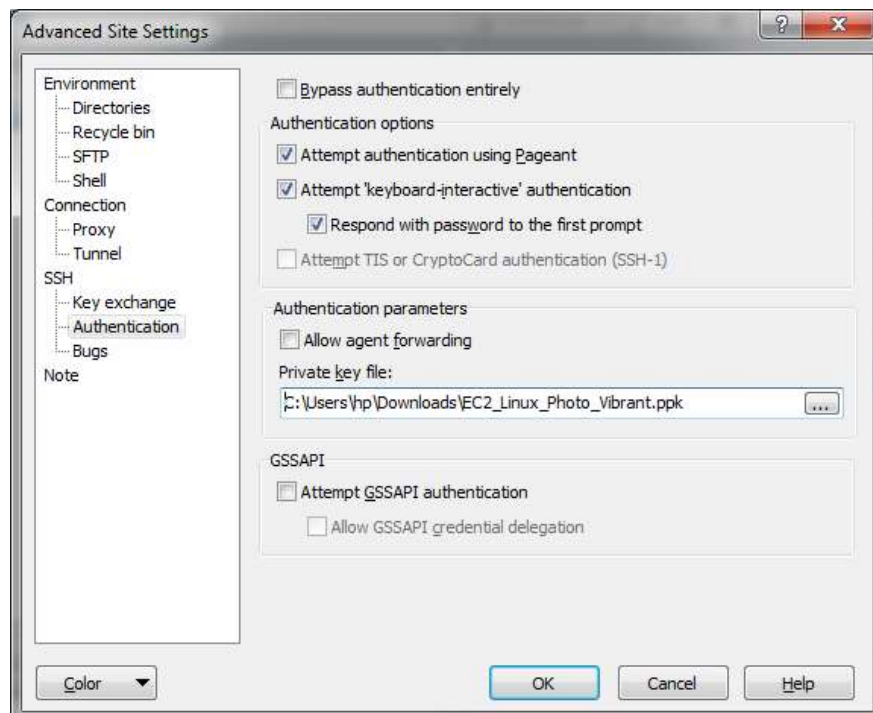
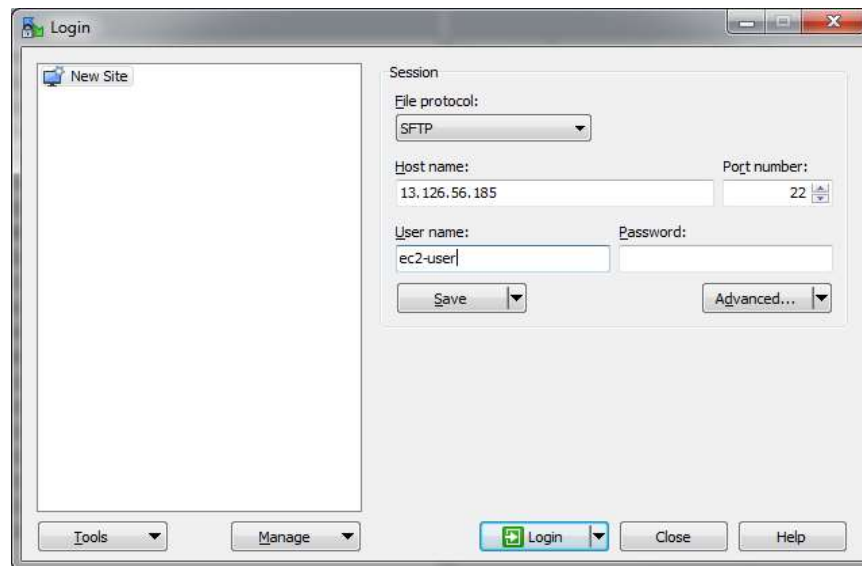

The screenshot shows the AWS Management Console interface. On the left is a navigation menu with categories like EC2 Dashboard, INSTANCES, IMAGES, ELASTIC BLOCK STORE, NETWORK & SECURITY, and FLOPPY DISK. The main content area displays the 'Launch Instance' page for an EC2 instance named 'EC2_Linux_Photo_Vibrant'. The instance is in the 'running' state, located in the 'ap-south-1a' availability zone. Below the instance summary, there are tabs for 'Description', 'Status Checks', 'Monitoring', and 'Tags'. The 'Description' tab is active, showing details such as Instance ID (i-079c3d2a2fced9d61), Instance Type (t2.micro), Elastic IP, Availability Zone (ap-south-1a), Security Groups (launch-wizard-SG), and Subnet (subnet-e0f50d62). The Public DNS (IPv4) is listed as ec2-13-126-56-185.ap-south-1.compute.amazonaws.com.

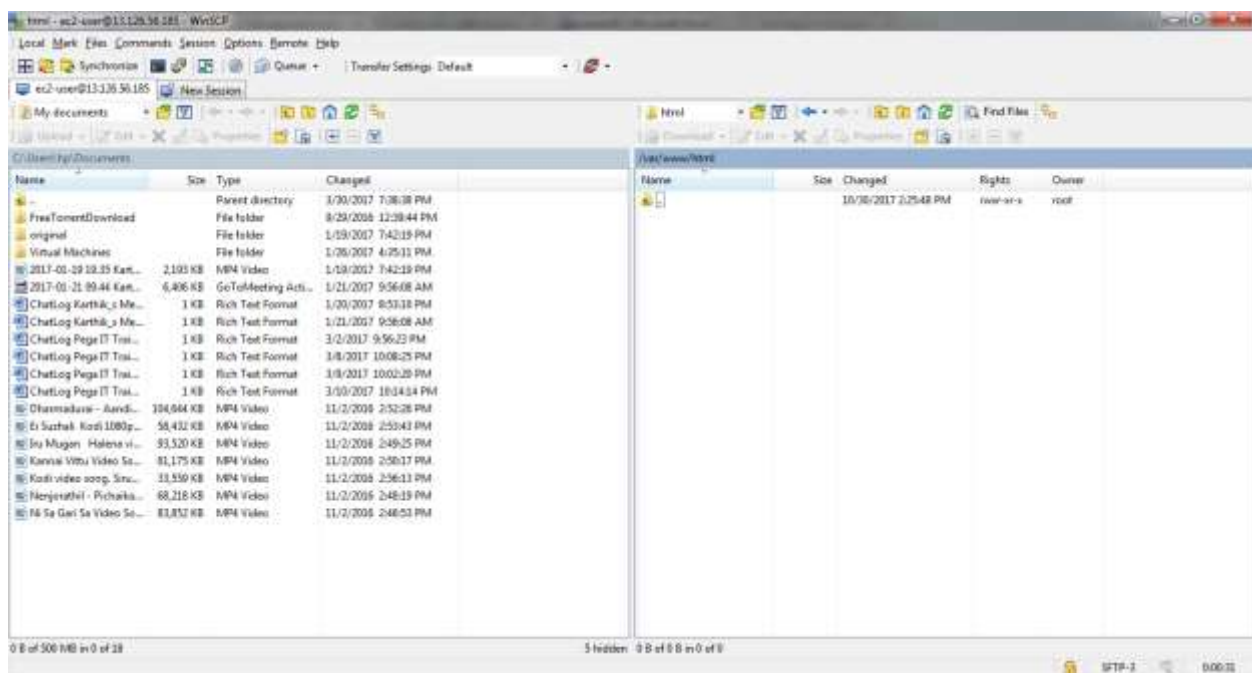
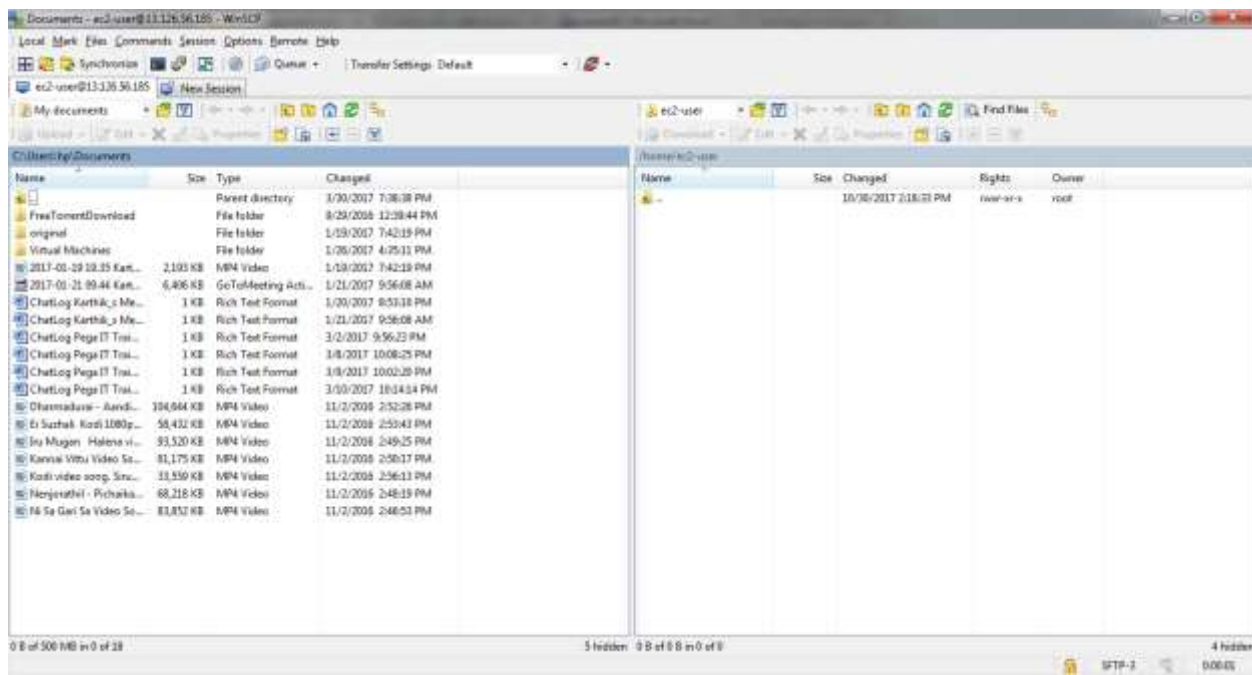
The screenshot shows a web browser displaying the 'Amazon Linux AMI Test Page'. The page title is 'Amazon Linux AMI Test Page'. The content includes a message stating: 'This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.' Below this, there are two sections: 'If you are a member of the general public:' and 'If you are the website administrator:'. The 'If you are the website administrator:' section includes instructions on how to add content to the directory and mentions the 'httpd.conf' file. At the bottom of the page, there is a 'Powered by' logo for Apache 2.2.

EC2 Instance launched Successfully.

Wow!!!!!!!!!!!!!! Great !!!!!!!!!!!!!

Use **WINSCP** tool to transfer the application files to the server before that download the file as per the screen shot.

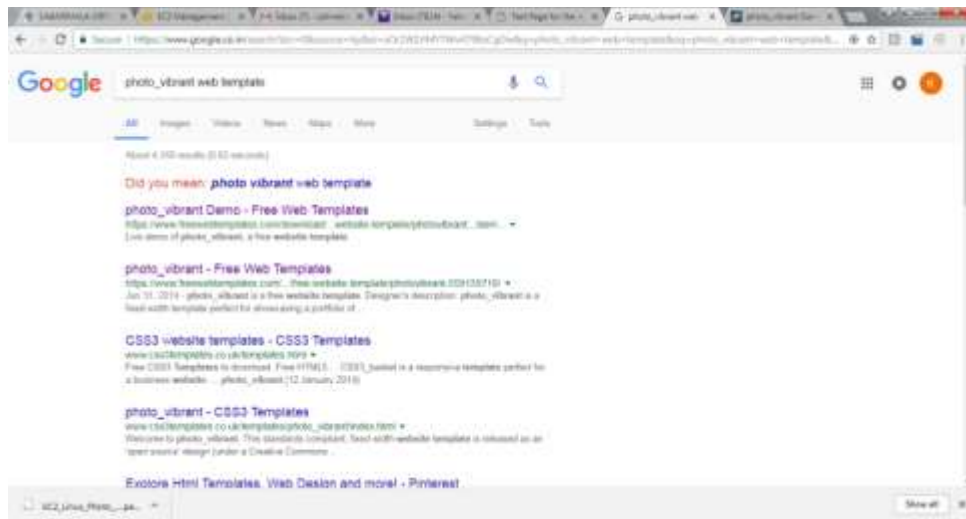
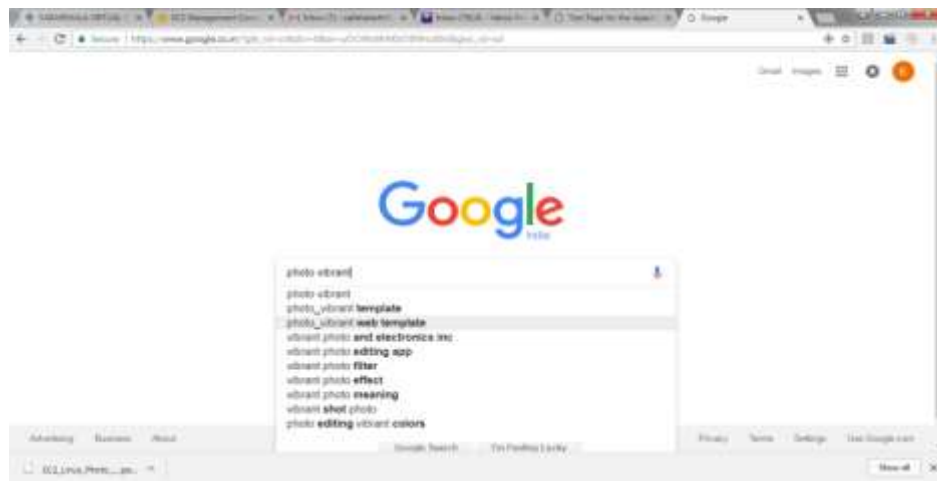




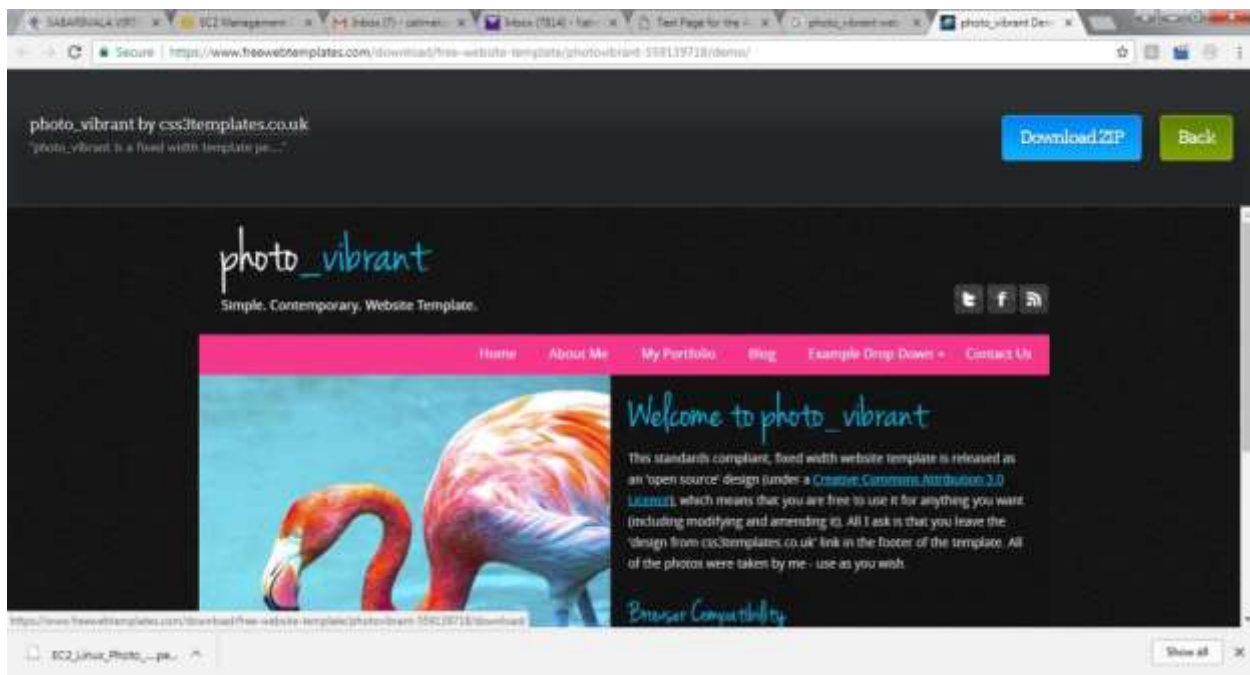
Give full permission to the folder before placing the application files.

```
[root@ip-172-31-27-39 ~]#
[root@ip-172-31-27-39 ~]#
[root@ip-172-31-27-39 ~]# chmod -Rf 777 /var/www/html
[root@ip-172-31-27-39 ~]#
[root@ip-172-31-27-39 ~]#
```

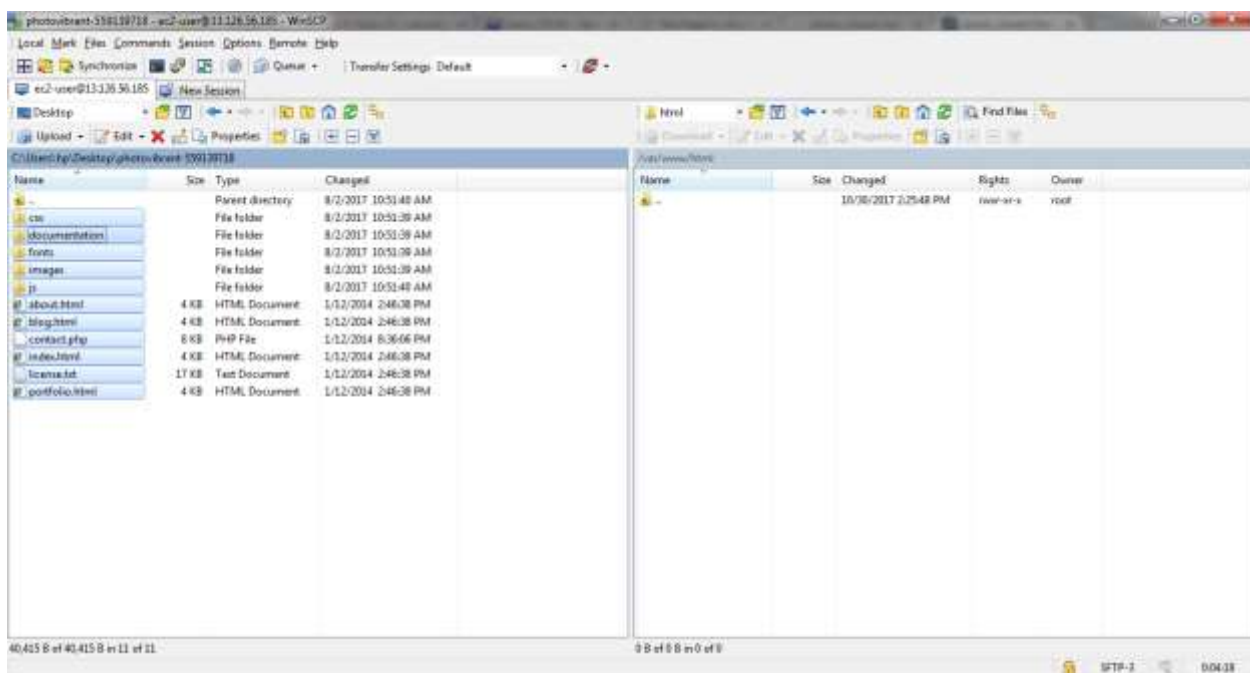
Suppose if we don't have developer to provide the application file, download it from google under free web template downloads,

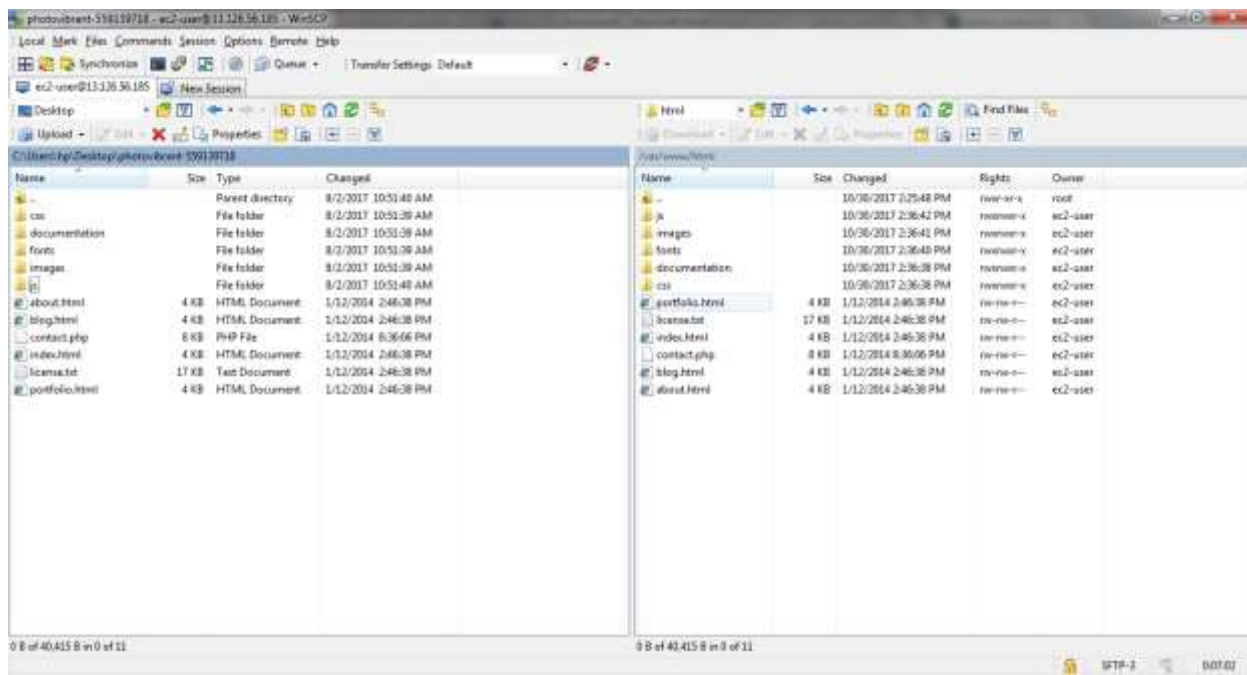


Download the Zip file from **TOP CORNER** of the Photo vibrant page

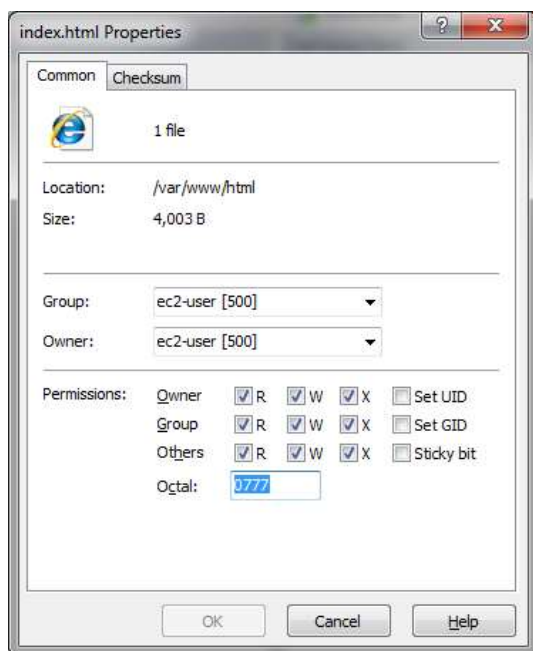


Extract the downloaded zip file & convert it into folder and keep it in the desktop.

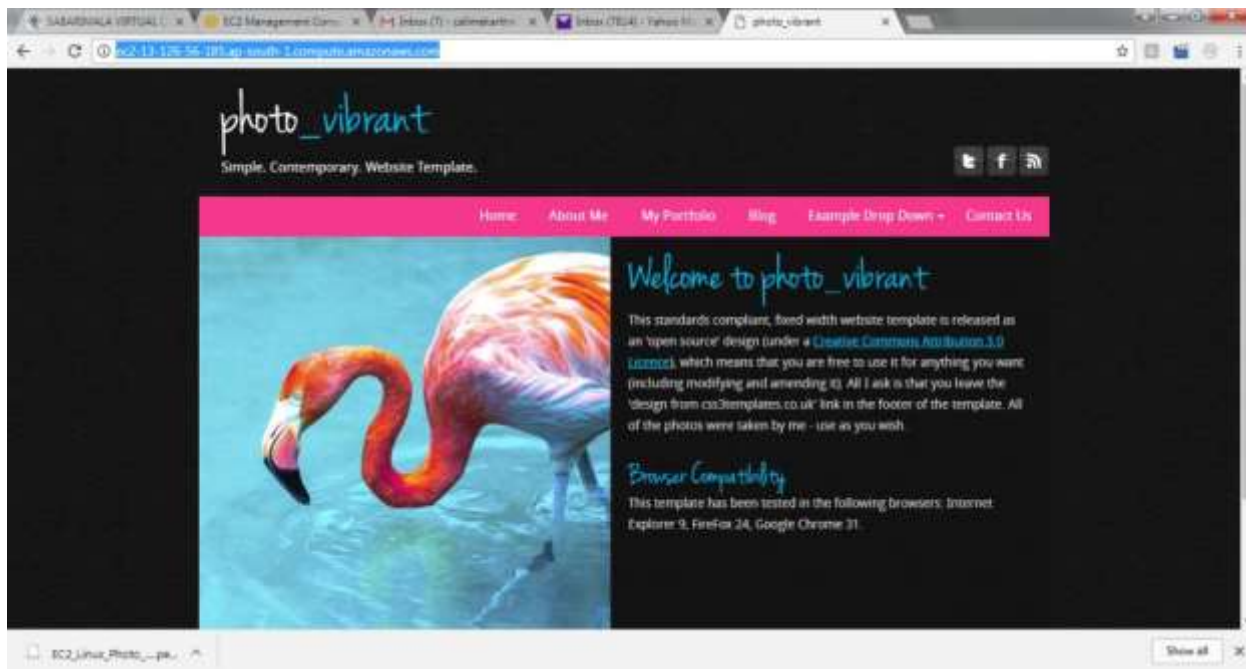




Then give full permissions for the files in the folder `/var/www/html` as below.



Then hit the same URL of the EC2 instance, from the browser, the page will be loaded and viewed.



Now, you have successfully launched your website. Wow Good Job!!!!!!!!!!!!!!

What is EC2?

Amazon Elastic Compute Cloud (EC2) is a web service that provides resizable compute capacity in the cloud. It 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.

The Amazon EC2 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 (called Amazon EC2 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. It provides developers the tools to build failure resilient applications and isolate themselves from common failure scenarios.

What are the benefits of EC2?

Easier and Faster – 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.

Elastic and Scalable – Quickly add and subtract resources to applications to meet customer demand and manage costs. Avoid provisioning resources upfront for projects with variable consumption rates or short lifetimes.

High Availability – Amazon EC2 provides developers the tools to build failure resilient applications and isolate themselves from common failure scenarios.

Cost-Effective – Consume only the amount of compute, storage and other IT resources needed. No long-term commitment, minimum spend or up-front investment is required

What are the main features of Amazon Elastic Compute Cloud (EC2)?

There are following main features of Amazon EC2: -

Instance: EC2 has instances in place of real hardware. An instance is a virtual computing environment with memory and compute capability.

Amazon Machine Images (AMI): In EC2 there are preconfigured templates for our instances. These are known as Amazon Machine Images (AMIs). We can create an AMI with the software that we need to start and run the server. It contains Operating System as well as other software.

Configuration: Amazon EC2 supports multiple configurations of CPU, memory, networking and storage capacity for instances.

Security: Amazon EC2 provides security in AWS by using public private key value pairs. We store public key in AWS and keep the private key in a secure place.

Elastic Block Store (EBS): With EC2 we can use EBS to persist the large amount of data.

Availability Zones: We can deploy our applications in multiple geographic locations called Availability Zones in AWS EC2.

Elastic IP Address: We can use static IP addresses for dynamic cloud computing in EC2. These IP addresses help in scaling and maintain high availability of the application in AWS.

Firewall: AWS EC2 also supports the firewall that can be used to specify the protocol, port and source IP range allowed to access instances in EC2.

What are the EC2 options?

The EC2 options are: -

OnDemand – Allows you to pay a fixed rate by hour (or by the second) with no commitment

Reserved – Provide you with a capacity reservation and offer a significant discount on the hourly charge for an instance 1 Year to 3 Year terms.

Spot – Enable you to bid whatever price you want for instance capacity, providing for even greater savings if your applications have flexible start and end times

Dedicated Hosts – Physical EC2 server is dedicated for your use. Dedicated Hosts can help you reduce costs by allowing you to use your existing server-bound software licenses.

[In detail](#)

OnDemand

- Users that want the low cost and flexibility of Amazon EC2 without any up-front payment or long-term commitment
- Applications with Short-term, spiky or unpredictable workloads that cannot be interrupted
- Applications being developed or tested on Amazon EC2 for the first time

Reserved

- Applications with steady state or predictable usage
- Applications that require reserved capacity
- Users able to make upfront payments to reduce their total computing costs even further: -
 - Standard RI's (up to 75% off on demand)
 - Convertible RI's (up to 54 % off on demand) capability to change the attributes of the RI as long as the exchange results in the creation of Reserved Instances of equal or greater value
 - Scheduled RI's available to launch within the time windows you reserve. This option allows you to match your capacity reservation to a predictable recurring schedule that only fraction of a day, a week, or a month

Spot

- Applications that have flexible start and end times
- Applications that are only feasible at very low compute prices
- Users with urgent computing needs for large amounts of additional capacity
- Remember with spot instances of cost involved: -
 - If you terminate the instance, you pay for the hour
 - If AWS terminates the spot instance, you get the hour it was terminated in for free

Dedicated Hosts

- Useful for regulatory requirements that may not support multi-tenant virtualization
- Great for licensing which does not support multi-tenancy or cloud deployments
- Can be purchased On-Demand (hourly)
- Can be purchased as a Reservation for up to 70 % off the On-Demand price.

What are the EC2 Instance Types? How do I remember?

Family	Speciality	Use case
D2	Dense Storage	Fileservers/Data Warehousing/Hadoop
R4	Memory Optimized	Memory Intensive Apps/DBs
M4	General Purpose	Application Servers
C4	Compute Optimized	CPU Intensive Apps/DBs
G2	Graphics Intensive	Video Encoding/ 3D Application Streaming
I2	High Speed Storage	NoSQL DBs, Data Warehousing etc
F1	Field Programmable Gate Array	Hardware acceleration for your code.
T2	Lowest Cost, General Purpose	Web Servers/Small DBs
P2	Graphics/General Purpose GPU	Machine Learning, Bit Coin Mining etc
X1	Memory Optimized	SAP HANA/Apache Spark etc

For easy remember, **DR MC GIFT PX**

D- Density **R**-Ram

M- Main choice for general purpose apps **C**-Compute

G-Graphics **I**-IOPS **F**- FPGA **T**-Cheap general purpose (Think T2 Micro)

P-Graphics (think Pics) **X**-Extreme Memory

What Is Amazon EC2 instance?

An EC2 instance is a virtual server in Amazon's Elastic Compute Cloud (EC2) for running applications on the Amazon Web Services (AWS) infrastructure.

Example

T2 instances are designed to provide moderate baseline performance and the capability to burst to significantly higher performance as required by your workload.

C4 instances are ideal for compute-bound applications that benefit from high performance processors.

What is the purpose of categories in the instances types?

Small

Small instances are used in Development environments, build servers, code repositories, low-traffic web applications, early product experiments, small databases.

Medium

Small and mid-size databases, data processing tasks that require additional memory, caching fleets, and for running backend server for SAP, Microsoft SharePoint, and other enterprise applications

Large

High Performance front-end fleets, web-servers, on demand batch processing, distributed analytics, high performance science and engineering applications, as serving, batch processing, MMO gaming, video encoding and distributed analytics

X Large

We recommend memory-optimized instances for high performance databases, distributed memory caches, in-memory analytics, genome assembly and analysis, larger deployments of SAP, Microsoft SharePoint and other applications.

What is Instance Meta-data?

- Used to get information about an instance (Such as Public IP)
- Curl <http://168.254.168.254/latest/meta-data/>

What are the steps to Create EC2?

We can create EC2 instance using Windows | Linux

Steps to create EC2: Windows

1. Choose an Amazon Machine Image (AMI)
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
8. Launch the Instance to check the server is running
9. Login in to the remote machine, using the public IP (e.g., Public DNS ec2-34-238-82-205.compute-1.amazonaws.com)
user: Administrator
Pwd: decrypted PEM FILE. (e.g. J4N))4vakEie(qHN\$aVqBFmvUHXW35sE)

Steps to create EC2: Linux - Amazon Linux

1. Choose an Amazon Machine Image (AMI)
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
8. Launch the Instance
9. PEM file generation
10. Upload the Pem file and generate a public key using **PUTTYGEN** tool.
11. Now you will have. ppk file.
12. Login in the putty with the public ip address and execute the below commands
Upload the ppk file in the auth.

user: ec2-user

`sudo su - root`

Install the apache: `yum install httpd -y`

start the apache: `service httpd start`

13. Launch the IP address in the browser to check the apache running.

Steps to display the output

1. Next, download the static files from any free templates from google (Photo vibrant template)

2. Go to WinSCP tool

Type Host: `54.85.175.112` Username: `ec2-user`

Advanced: Load ppk file and click login Now WinSCP gets loaded

3. Login to putty as `ec2-user`

4. Enter into the root - `sudo su - root`, Provide the permissions `chmod -Rf 777 /var/www/html`

5. Put the file into the path - `/var/www/html` with full permission for the folder and the files.

6. Hit the ip address of the EC2, our application's page will be available.

Now we have successfully launched our application through EC2 Instances. Great Job.

What is the difference between Amazon S3 and Amazon EC2?

Amazon S3 is storage service in cloud. It is used to store large amount of data files. These files can be image files, pdf etc. like static data or these can be dynamic data that is created during runtime.

Amazon EC2 is a remote computing environment running in cloud. We can install our software and operating system on an EC2 instance. We can use it to run our servers like-Web server, Application server etc. So S3 is a storage system where as EC2 is a computing system in AWS.

How does Amazon EC2 works?

Amazon Elastic Compute Cloud (Amazon EC2) is a computing environment provided by AWS. It supports highly scalable computing capacity in AWS.

Instead of buying hardware for servers we can use Amazon EC2 to deploy our applications. So, there is no need to buy and maintain the hardware within our own datacenter. We can just rent the Amazon EC2 servers. Based on our varying needs we can use as few and as many Amazon EC2 instances.

It even provides auto-scaling options in which the instances scale up or down based on the load and traffic spikes. It is easier to deploy applications on EC2. Even we can configure security and networking in Amazon EC2 much easily than our own custom data center.

Explain can you vertically scale an Amazon instance? How?

Yes. This is an incredible characteristic of cloud virtualization and AWS. Spinup is a huge case when compared to the one which you are running with. Let up the instance and separate the root EBS volume from this server and remove. Next, stop your live instance, remove its root volume. Note down the distinctive device ID and attach root volume to your new server and start it again. This is the way to scaling vertically in place.

Explain storage for Amazon EC2 instance?

Amazon EC2 provides many data storage options for your instances. Each option has a unique combination of performance and durability. These storages can be used independently or in combination to suit your requirements.

There are mainly four types of storage provided by AWS: -

Amazon EBS: Its durable, block-level storage volumes that you can attach to a running Amazon EC2 instance. The Amazon EBS volume persists independently from the running life of an Amazon EC2 instance. After an EBS volume is attached to an instance, you can use it like any other physical hard drive. Amazon EBS encryption feature supports encryption feature.

Amazon EC2 Instance Store: Storage disk that is attached to the host computer is referred to as instance store. Instance storage provides temporary block-level storage for Amazon EC2 instances. The data on an instance store volume persists only during the life of the associated Amazon EC2 instance; if you stop or terminate an instance, any data on instance store volumes is lost.

Amazon S3: Amazon S3 provides access to reliable and inexpensive data storage infrastructure. It is designed to make web-scale computing easier by enabling you to store and retrieve any amount of data, at any time, from within Amazon EC2 or anywhere on the web.

Adding Storage: Every time you launch an instance from an AMI, a root storage device is created for that instance. The root storage device contains all the information necessary to boot the instance. You can specify storage volumes in addition to the root device volume when you create an AMI or launch an instance using block device mapping.

What are the Security Best Practices for Amazon EC2?

There are several best practices for secure Amazon EC2. Following are few of them.

- Use AWS Identity and Access Management (IAM) to control access to your AWS resources.
- Restrict access by only allowing trusted hosts or networks to access ports on your instance.
- Review the rules in your security groups regularly, and ensure that you apply the principle of least Privilege — only open up permissions that you require.
- Disable password-based logins for instances launched from your AMI. Passwords can be found or cracked and are a security risk.

Explain Stopping, Starting, and Terminating an Amazon EC2 instance?

Stopping and Starting an instance: When an instance is stopped, the instance performs a normal shutdown and then transitions to a stopped state. All of its Amazon EBS volumes remain attached, and you can start the instance again at a later time. You are not charged for additional instance hours while the instance is in a stopped state.

Terminating an instance: When an instance is terminated, the instance performs a normal shutdown, then the attached Amazon EBS volumes are deleted unless the volume's `deleteOnTermination` attribute is set to false. The instance itself is also deleted, and you can't start the instance again at a later time.

What are regions and availability zones in Amazon EC2? Explain in brief?

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**.

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.

What is Regions and Endpoints in AWS?

To reduce data latency in your applications, most Amazon Web Services products allow you to select a regional endpoint to make your requests. An endpoint is a URL that is the entry point for a web service. For example, `https://dynamodb.us-west-2.amazonaws.com` is an entry point for the Amazon DynamoDB service.

Some services, such as IAM, do not support regions; their endpoints therefore do not include a region.

A few services, such as Amazon EC2, let you specify an endpoint that does not include a specific region, for example, <https://ec2.amazonaws.com>. In that case, AWS routes the endpoint to us-east-1.

How to find your regions and Availability Zones using the Amazon EC2 CLI?

Use the `ec2-describe-regions` command as follows to describe your regions.

PROMPT> `ec2-describe-regions`

REGION us-east-1 ec2.us-east-1.amazonaws.com

REGION ap-northeast-1 ec2.ap-northeast-1.amazonaws.com

REGION ap-southeast-1 ec2.ap-southeast-1.amazonaws.com

What is a Placement Group in EC2?

AWS provides an option of creating a Placement Group in EC2 to logically group the instances within a single Availability Zone.

We get the benefits of low network latency and high network throughput by using a Placement Group. Placement Group is a free option as of now. When we stop an instance, it will run in the same Placement Group when it restarts at a later point of time. The biggest limitation of Placement Group is that we cannot add instances from multiple availability zones to one Placement Group.

What are the best practices for Amazon EC2?

To get the maximum benefit from and satisfaction with Amazon EC2, there are mainly four best practices.

- Security and Network Best Practices
- Storage
- Resource Management
- Backup and Recovery

What is the underlying Hypervisor for EC2?

Xen

How you're charged in Amazon EC2? Explain in detail?

- Charges vary upon AMIs backed and storage volumes.
- AMIs backed by instance storage charged for: AMI storage + Instance usage
- AMIs backed by Amazon EBS storage charged for: Volume storage + Usage in addition to the AMI + instance usage
- When an Amazon EBS-backed instance is stopped, you are not charged for instance usage, but you are still charged for volume storage.
- AWS charges a full instance hour for every transition from a stopped state to a running state, even if we transition the instance multiple times within a single hour.

- For example: if hourly instance charge for your instance is \$0.10 and if you were to run that instance for one hour without stopping it, you would be charged \$0.10. If you stopped and restarted that instance twice during that hour, then you would be charged \$0.30 for that hour of usage (the initial \$0.10, plus 2 x \$0.10 for each restart).

What is the difference between scalability and elasticity?

Scalability is a characteristic of cloud computing through which increasing workload can be handled by increasing in proportion the amount of resource capacity. It allows the architecture to provide on demand resources if the requirement is being raised by the traffic.

Whereas, **elasticity** is being one of the characteristic provide the concept of commissioning and decommissioning of large amount of resource capacity dynamically. It is measured by the speed by which the resources are coming on demand and the usage of the resources.

What will happen when we reboot an Amazon EC2 instance?

When we reboot an Amazon EC2 instance, it reboots like computer. There is no effect on hard disk. It reboots with the latest configurations settings that were present just before the reboot. Once you terminate the instance, then only billing stops. You can reboot an instance multiple times but the billing will continue.

What are the steps to change the root EBS device of an Amazon EC2 instance?

We can follow these steps to change the root EBS device of an EC2 instance: Stop Amazon EC2 instance, detach root EBS volume, attach another EBS volume, Start Amazon EC2 instance

What is Public Key Credentials and how to install it?

Amazon EC2 uses public-key cryptography to encrypt and decrypt login information. Public-key cryptography uses a public key to encrypt a piece of data, such as a password, then the recipient uses the private key to decrypt the data. The public and private keys are known as a key pair.

After configuring the AMI to prevent logging in using a password, you must make sure users can log in using another mechanism.

How to disable Password-Based Logins for Root in Amazon EC2 Instance?

Using a fixed root password for a public AMI is a security risk that can quickly become known. Even relying on users to change the password after the first login opens a small window of opportunity for potential abuse.

Following are the steps to disable password-based remote logins for the root user.

1. Open the `/etc/ssh/sshd_config` file with a text editor and locate the following line:

`#PermitRootLogin yes`

2. Change the line to:

`PermitRootLogin without-password`

The location of this configuration file might differ for your distribution.

While connecting to your instance what are the possible connection issues one might face?

The possible connection errors one might encounter while connecting instances are

- Connection timed out
- User key not recognized by the server
- Host key not found, permission denied
- Unprotected private key file
- Server refused our key or No supported authentication method available
- Error using MindTerm on Safari Browser
- Error using Mac OS X RDP Client

What are the main uses of Amazon Elastic Compute Cloud (EC2)?

Amazon EC2 provides scalable computing resources for creating a software infrastructure. It is very easy to deploy application in Amazon E2.

Main uses of EC2 are: -

Easy Configuration: We can easily configure our servers in EC2 and manage the capacity. **Control:** EC2 provides complete control of computing resources even to developers. A user can run the EC2 environment according to his/her system's needs.

Fast Reboot: It is very fast to reboot the instances in EC2. It reduces the overall deployment and development time.

Scalability: In EC2 we can create a highly scalable environment based on the load that is expected on our application. **Resilient:** It is very easy to create and terminate servers in EC2. Due to this we can develop resilient applications in EC2.

How you will find out the instance id from within an ec2 machine?

```
wget -q -O - http://instance-data/latest/meta-data/instance-id
```

If you need programmatic access to the instance ID from within a script

```
die() { status=$1; shift; echo "FATAL: $*"; exit $status; }
```

```
EC2_INSTANCE_id=""`wget -q -O - http://instance-data/latest/meta-data/instance-id || die \"wget  
instance-id has failed: $?\"`"
```

Is it possible to use AWS as a web host? What are the way of using AWS as a web host?

Yes, it is completely possible to host websites on AWS in 2 ways:-

- **Easy** – S3 (Simple Storage Solution) is a bucket storage solution that lets you serve static content e.g. images but has recently been upgraded so you can use it to host flat .html files and your site will get served by a default Apache installation with very little configuration on your part (but also little control).
- **Trickier** – You can use EC2 (Elastic Compute Cloud) and create a virtual Linux instance then install Apache/NGinx (or whatever) on that to give you complete control over serving whatever/however you want. You use SecurityGroups to enable/disable ports for individual machines or groups of them.

How to access/ping a server located on AWS?

Using UI:

In your security group:

- Click the inbound tab
Create a custom ICMP rule
Select echo request
Use range 0.0.0.0/0 for everyone or lock it down to specific IPs
Apply the changes
and you'll be able to ping.
- Using cmd: To do this on the command line you can run:
- `ec2-authorize -P icmp -t -1:-1 -s 0.0.0.0/0`

What are the 4 level of AWS premium support?

Basic, Developer, Business & Enterprise

What does the following command do with respect to the Amazon EC2 security groups?

`ec2-create-group CreateSecurityGroup`

- A. Groups the user created security groups into a new group for easy access.
- B. Creates a new security group for use with your account.
- C. Creates a new group inside the security group.
- D. Creates a new rule inside the security group.

Answer B

Explanation: A Security group is just like a firewall, it controls the traffic in and out of your instance. In AWS terms, the inbound and outbound traffic. The command mentioned is pretty straight forward, it says create security group, and does the same. Moving along, once your security group is created, you can add different rules in it. For example, you have an RDS instance, to access it, you have to add the public IP address of the machine from which you want access the instance in its security group.

You have a video trans-coding application. The videos are processed according to a queue. If the processing of a video is interrupted in one instance, it is resumed in another instance. Currently there is a huge back-log of videos which needs to be processed, for this you need to add more instances, but you need these instances only until your backlog is reduced. Which of these would be an efficient way to do it?

You should be using an On-Demand instance for the same. Why? First of all, the workload has to be processed now, meaning it is urgent, secondly you don't need them once your backlog is cleared, therefore Reserved Instance is out of the picture, and since the work is urgent, you cannot stop the work on your instance just because the spot price spiked, therefore Spot Instances shall also not be used. Hence On-Demand instances shall be the right choice in this case.

You have a distributed application that periodically processes large volumes of data across multiple Amazon EC2 Instances. The application is designed to recover gracefully from Amazon EC2 instance failures. You are required to accomplish this task in the most cost-effective way.

Which of the following will meet your requirements?

- A. Spot Instances**
- B. Reserved instances
- C. Dedicated instances
- D. On-Demand instances

Answer: A

Explanation: Since the work we are addressing here is not continuous, a reserved instance shall be idle at times, same goes with On Demand instances. Also it does not make sense to launch an On Demand instance whenever work comes up, since it is expensive. Hence Spot Instances will be the right fit because of their low rates and no long-term commitments.

How is stopping and terminating an instance different from each other?

Starting, stopping and terminating are the three states in an EC2 instance, let's discuss them in detail:

Stopping and Starting an instance: When an instance is stopped, the instance performs a normal shutdown and then transitions to a stopped state. All of its Amazon EBS volumes remain attached, and you can start the instance again at a later time. You are not charged for additional instance hours while the instance is in a stopped state.

Terminating an instance: When an instance is terminated, the instance performs a normal shutdown, then the attached Amazon EBS volumes are deleted unless the volume's `deleteOnTermination` attribute is set to false. The instance itself is also deleted, and you can't start the instance again at a later time.

If I want my instance to run on a single-tenant hardware, which value do I have to set the instance's tenancy attribute to?

- A. Dedicated**
- B. Isolated
- C. One
- D. Reserved

Answer A

Explanation: The Instance tenancy attribute should be set to Dedicated Instance. The rest of the values are invalid.

When will you incur costs with an Elastic IP address (EIP)?

- A. When an EIP is allocated.
- B. When it is allocated and associated with a running instance.
- C. When it is allocated and associated with a stopped instance.**
- D. Costs are incurred regardless of whether the EIP is associated with a running instance.

Answer C

Explanation: You are not charged, if only one Elastic IP address is attached with your running instance.

But you do get charged in the following conditions:

When you use more than one Elastic IPs with your instance.

When your Elastic IP is attached to a stopped instance.

When your Elastic IP is not attached to any instance.

How is a Spot instance different from an On-Demand instance or Reserved Instance?

First of all, let's understand that Spot Instance, On-Demand instance and Reserved Instances are all models for pricing. Moving along, spot instances provide the ability for customers to purchase compute capacity with no upfront commitment, at hourly rates usually lower than the On-Demand rate in each region.

Spot instances are just like bidding, the bidding price is called Spot Price. The Spot Price fluctuates based on supply and demand for instances, but customers will never pay more than the maximum price they have specified. If the Spot Price moves higher than a customer's maximum price, the customer's EC2 instance will be shut down automatically.

But the reverse is not true, if the Spot prices come down again, your EC2 instance will not be launched automatically, one has to do that manually. In Spot and On demand instance, there is no commitment for the duration from the user side, however in reserved instances one has to stick to the time period that he has chosen.

Are the Reserved Instances available for Multi-AZ Deployments?

A. Multi-AZ Deployments are only available for Cluster Compute instances types

B. Available for all instance types

C. Only available for M3 instance types

D. Not Available for Reserved Instances

Answer B

Explanation: Reserved Instances is a pricing model, which is available for all instance types in EC2.

How to use the processor state control feature available on the c4.8xlarge instance?

The processor state control consists of 2 states:

The **C state** – Sleep state varying from c0 to c6. C6 being the deepest sleep state for a processor

The **P state** – Performance state p0 being the highest and p15 being the lowest possible frequency.

Now, why the C state and P state. Processors have cores, these cores need thermal headroom to boost their performance. Now since all the cores are on the processor the temperature should be kept at an optimal state so that all the cores can perform at the highest performance.

Now how will these states help in that? If a core is put into sleep state it will reduce the overall temperature of the processor and hence other cores can perform better. Now the same can be synchronized with other cores, so that the processor can boost as many cores it can by timely putting other cores to sleep, and thus get an overall performance boost.

Concluding, the C and P state can be customized in some EC2 instances like the c4.8xlarge instance and thus you can customize the processor according to your workload.

How to do it? You can refer this tutorial for the same.

What kind of network performance parameters can you expect when you launch instances in cluster placement group?

The network performance depends on the instance type and network performance specification, if launched in a placement group you can expect up to

10 Gbps in a single-flow,

20 Gbps in multiframe i.e., full duplex

Network traffic outside the placement group will be limited to 5 Gbps(full duplex).

To deploy a 4-node cluster of Hadoop in AWS which instance type can be used?

First let's understand what actually happens in a Hadoop cluster, the Hadoop cluster follows a master slave concept. The master machine processes all the data, slave machines store the data and act as data nodes.

Since all the storage happens at the slave, a higher capacity hard disk would be recommended and since master does all the processing, a higher RAM and a much better CPU is required. Therefore, you can select the configuration of your machine depending on your workload.

For e.g. – In this case c4.8xlarge will be preferred for master machine whereas for slave machine we can select i2.large instance. If you don't want to deal with configuring your instance and installing hadoop cluster manually, you can straight away launch an Amazon EMR (Elastic Map Reduce) instance which automatically configures the servers for you. You dump your data to be processed in S3, EMR picks it from there, processes it, and dumps it back into S3.

How do you choose an Availability Zone?

Let's understand this through an example, consider there's a company which has user base in India as well as in the US.

Let us see how we will choose the region for this use case:

Regions	<ul style="list-style-type: none"> • Mumbai/N Virginia
Instance Type (Reserved Instance)	<ul style="list-style-type: none"> • e.g. amazon ec2- m4.4xlarge 16(vCPU), 64 GB RAM
Pricing (1 Year)	<ul style="list-style-type: none"> • Mumbai - \$691/monthly - \$0.9 hourly • N Virginia - \$480/monthly - \$0.6 hourly
Latency	<ul style="list-style-type: none"> • From USA to India - Low • From India to USA - High

So, with reference to the above figure the regions to choose between are, Mumbai and North Virginia. Now let us first compare the pricing, you have hourly prices, which can be converted to your per month figure. Here North Virginia emerges as a winner. But, pricing cannot be the only parameter to consider. Performance should also be kept in mind hence, let's look at latency as well. Latency basically is the time that a server takes to respond to your requests i.e., the response time. North Virginia wins again! So, concluding, North Virginia should be chosen for this use case.

Is one Elastic IP address enough for every instance that I have running?

Depends! Every instance comes with its own private and public address. The private address is associated exclusively with the instance and is returned to Amazon EC2 only when it is stopped or terminated. Similarly, the public address is associated exclusively with the instance until it is stopped or terminated. However, this can be replaced by the Elastic IP address, which stays with the instance as long as the user doesn't manually detach it. But what if you are hosting multiple websites on your EC2 server, in that case you may require more than one Elastic IP address.

Elastic IP

Elastic IP Address Highlights

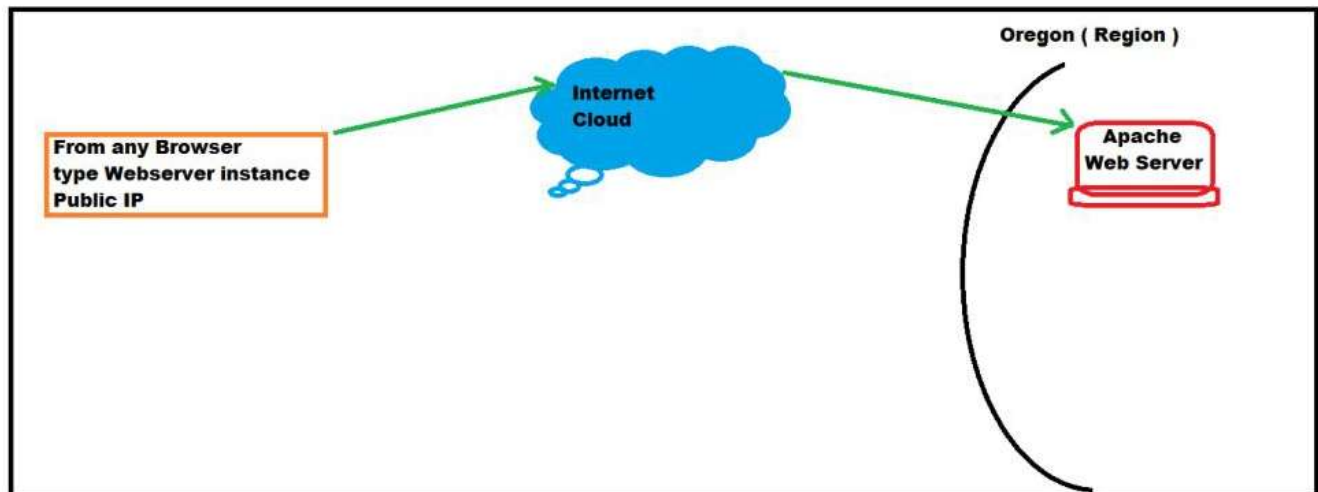
Amazon provides an **Elastic IP Address** with an AWS account. An Elastic IP address is a public and static IP address based on IPv4 protocol. It is designed for dynamic cloud computing. This IP address is reachable from the Internet.

If we do not have a specific IP address for our EC2 instance, then we can associate our instance to the Elastic IP address of our AWS account. Now our instance can communicate on the Internet with this Elastic IP Address.

Share the Linux Instance with Elastic IP Configuration Step by Step?

To Configure Webserver on Amazon Linux instance with Elastic IP

Topology



Pre-requisites

User should have AWS account, or IAM user with EC2fullaccess

Task:

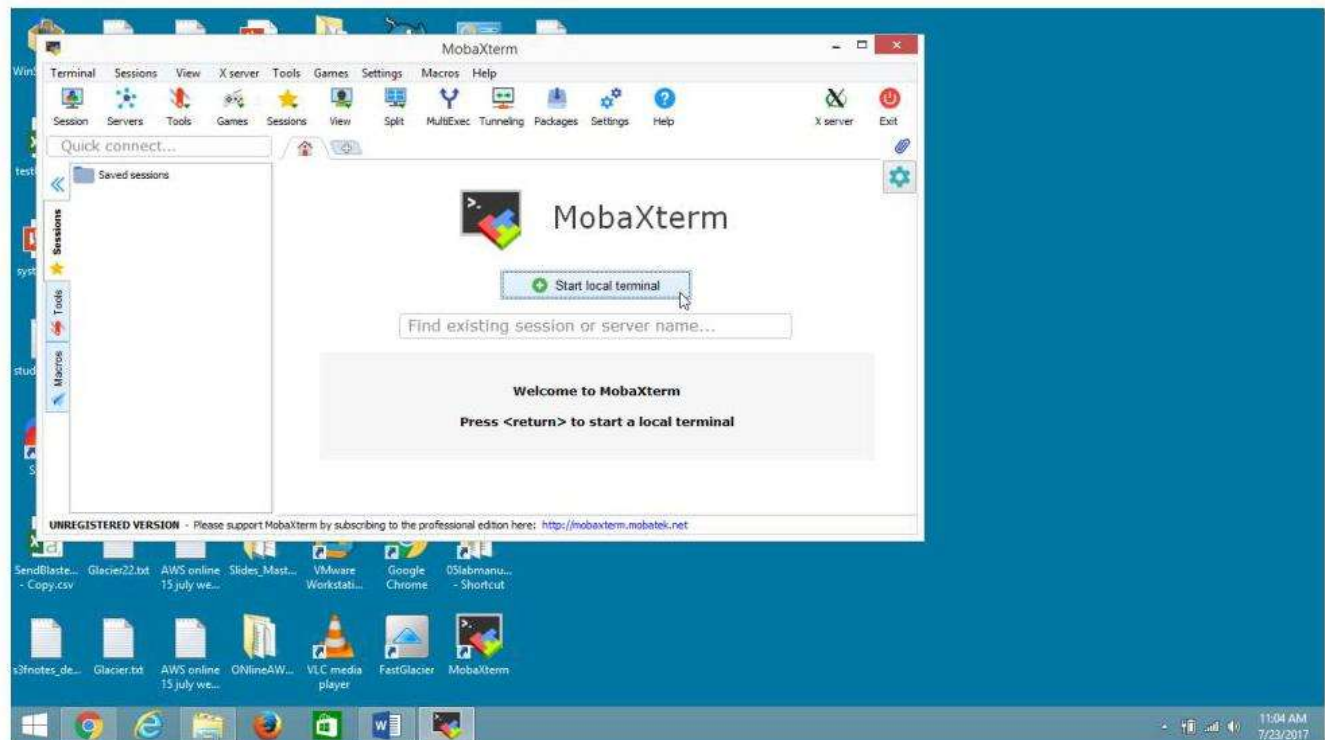
- Launch Linux instance in AWS
- Switch to the root user
- Configure Apache Webserver
- Enable HTTP port in Security Group
- Open the Browser and provide the Public IP or DNS_name of Webserver
- Assign an Elastic IP
- Releasing an Elastic IP

1) Launch Amazon Linux instance and login to your instance

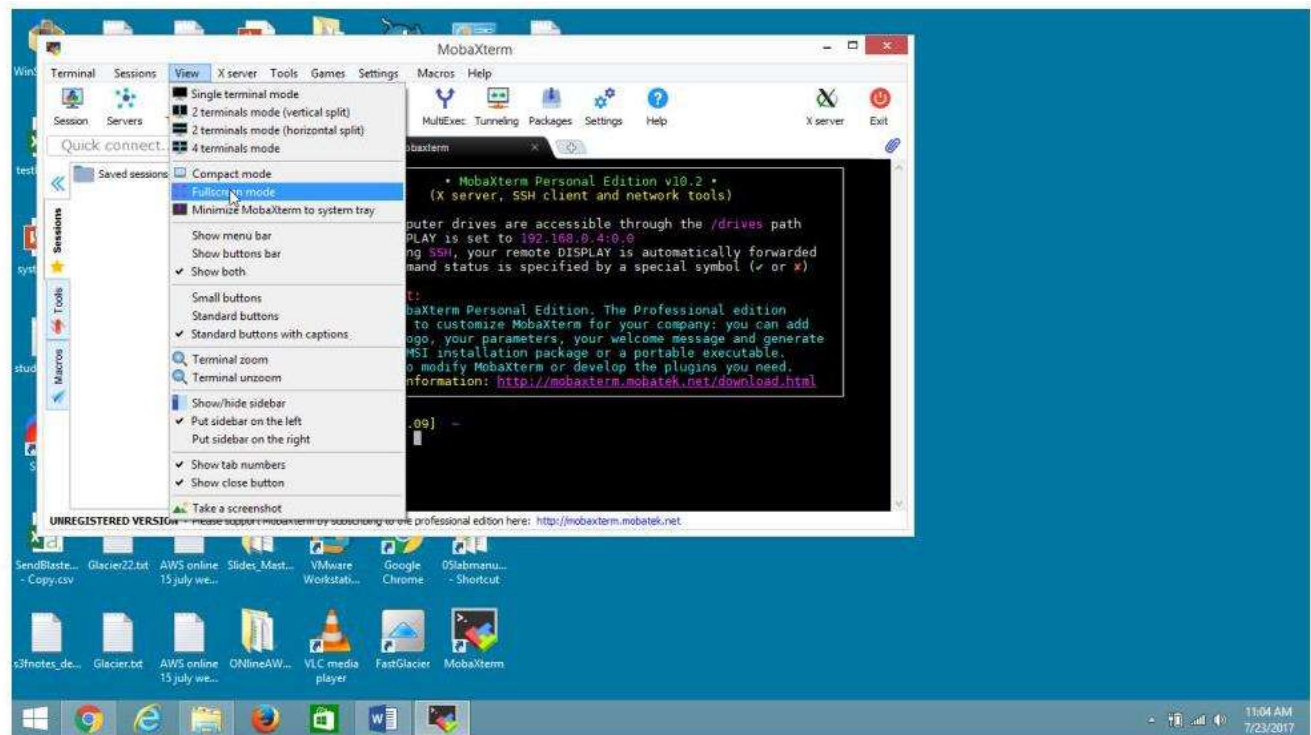
Hopefully, just you have created or Please refer (How to configure amazon Linux EC2 instance)

2) Connect to Linux instance from window using MobaXterm

- **Open MobaXterm**
- **Click on Start local terminal**

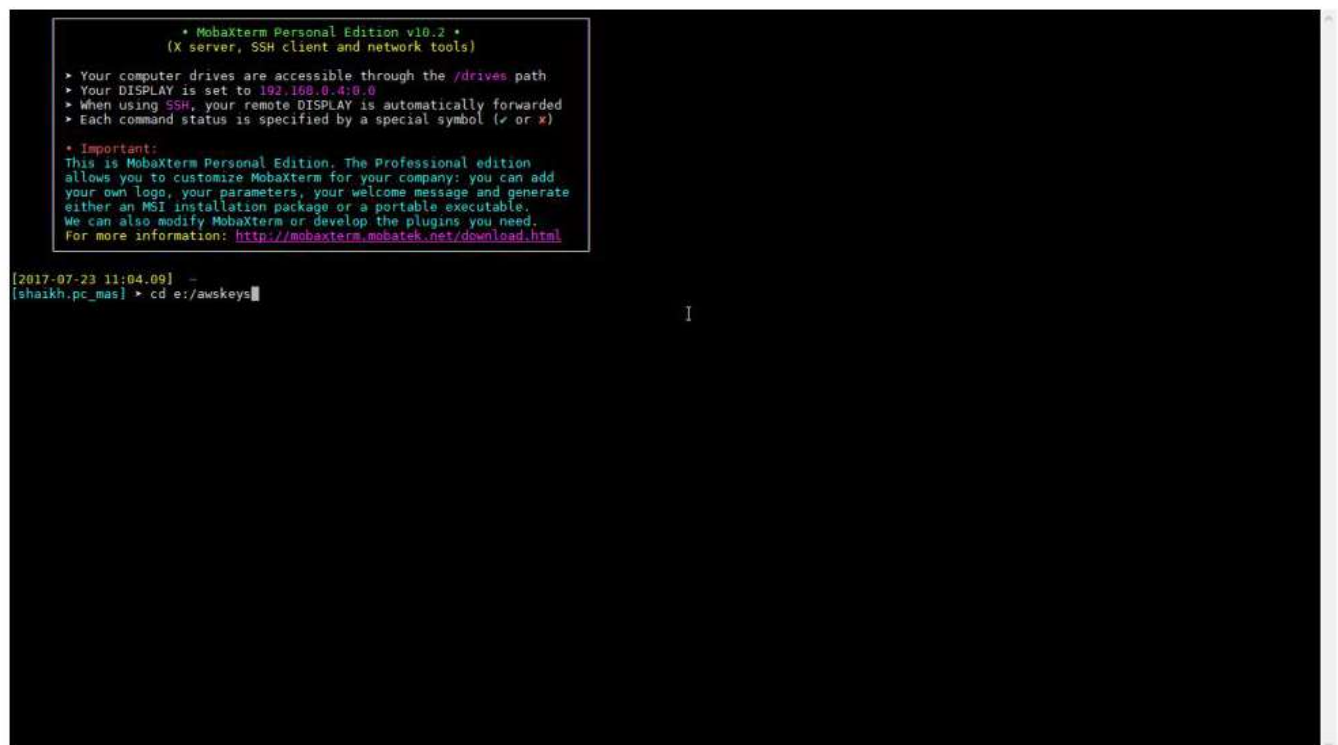


Go to Full Screen Mode



Navigate to the folder where the key*.pem file is stored

Eg: `cd e:/awskeys`



Login to Linux instance by typing the following command

`ssh -i "keyorg123.pem" ec2-user@ec2-54-186-150-140.us-west-2.compute.amazonaws.com`

```
[2017-07-23 09:34.47] /drives/e/awskeys
[shaikh.pc_mas] > ssh -i "keyorg123.pem" ec2-user@ec2-54-186-150-140.us-west-2.compute.amazonaws.com
Warning: Permanently added 'ec2-54-186-150-140.us-west-2.compute.amazonaws.com' (RSA) to the list of known hosts.
X11 forwarding request failed on channel 0

 _ | _ | _ )
 _ | ( _ - /   Amazon Linux AMI
 _ | \ _ | _ |

https://aws.amazon.com/amazon-linux-ami/2017.03-release-notes/
1 package(s) needed for security, out of 1 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-10-246 ~]$
```

Switch to the root user

Type "**sudo su**"

```
[ec2-user@ip-172-31-10-246 ~]$ sudo su
[root@ip-172-31-10-246 ec2-user]#
```

Configure Apache Webserver run the following commands as below

```
[root@ip-172-31-10-246 ec2-user]# yum install httpd -y
[root@ip-172-31-10-246 ec2-user]# chkconfig httpd on
[root@ip-172-31-10-246 ec2-user]# service httpd restart
[root@ip-172-31-10-246 ec2-user]# vi /var/www/html/index.html
```

To use vi editor

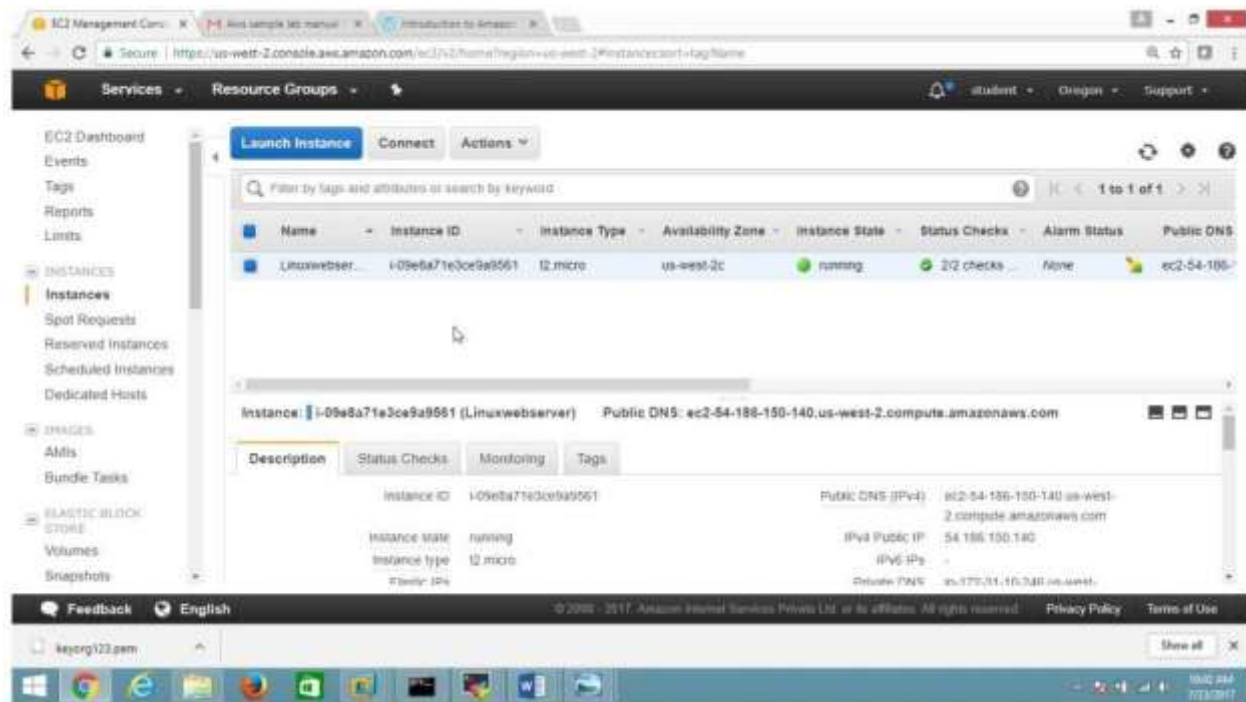
Go to insert mode by typing 'i' and add following code in index.html file

Note: [esc+shift+colon -> :wq!] (Save & Quit in Vi Editor)

```
<html>
<body bgcolor=black>
<marquee>
  <font color=gold>
    <h1> Welcome to Apache Webserver in AWS instance </h1>
  </font>
</marquee>
</body>
</html>
```

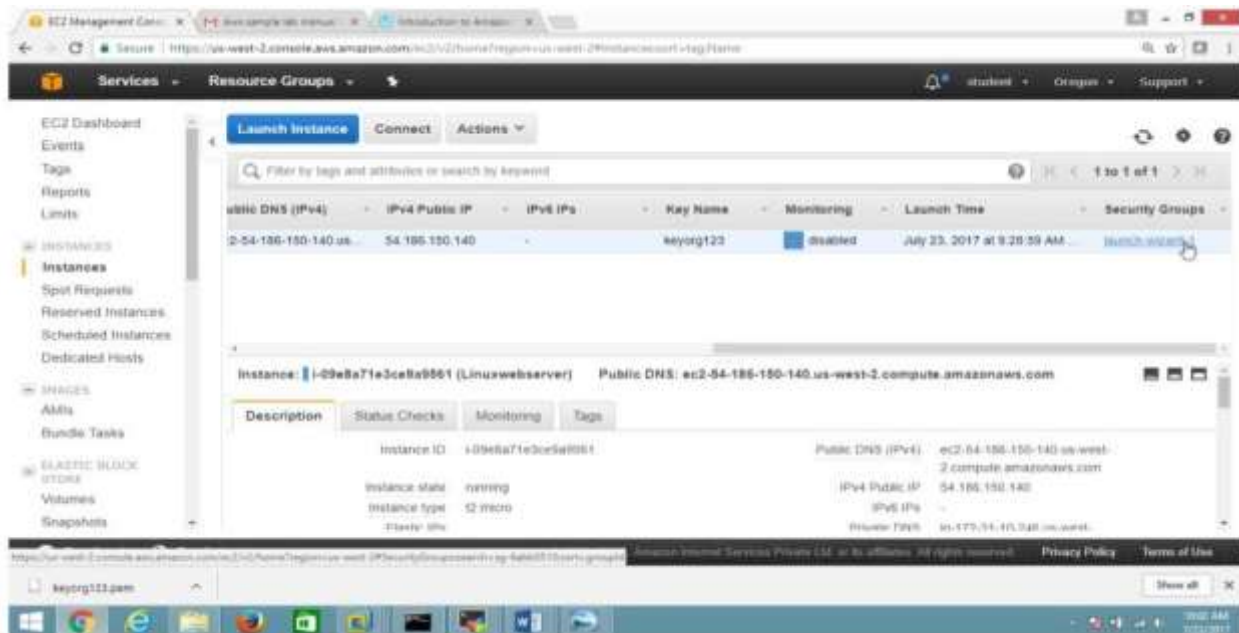

3) Create an inbound Rule to Allow http traffic on port 80

- Open the AWS console
- On the EC2 Dashboard panel
- Select the Linux instance



Go to the right end

Select "Security Groups" Click on "Launch-Wizard-1"



Click on "Inbound" (tab)button

The screenshot shows the AWS Management Console interface. The left sidebar contains navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES, IMAGES, and ELASTIC BLOCK STORE. The main content area displays the 'Security Group: sg-6abb0510' page. The 'Inbound' tab is selected, showing a table with one rule: SSH (Type), TCP (Protocol), Port Range 22, and Source 0.0.0.0/0. The 'Edit' button is visible above the table. The top navigation bar includes 'Services', 'Resource Groups', and user information. The bottom status bar shows the date and time as 10:02 AM 7/23/2017.

Name	Group ID	Group Name	VPC ID	Description
sg-6abb0510	sg-6abb0510	launch-wizard-1	vpc-89c341ee	launch-wizard-1 created 2017-07-23T09:27

Security Group: sg-6abb0510

Description Inbound Outbound Tags

Group name launch-wizard-1 Group description launch-wizard-1 created 2017-07-23T09:27 59:347+05:30

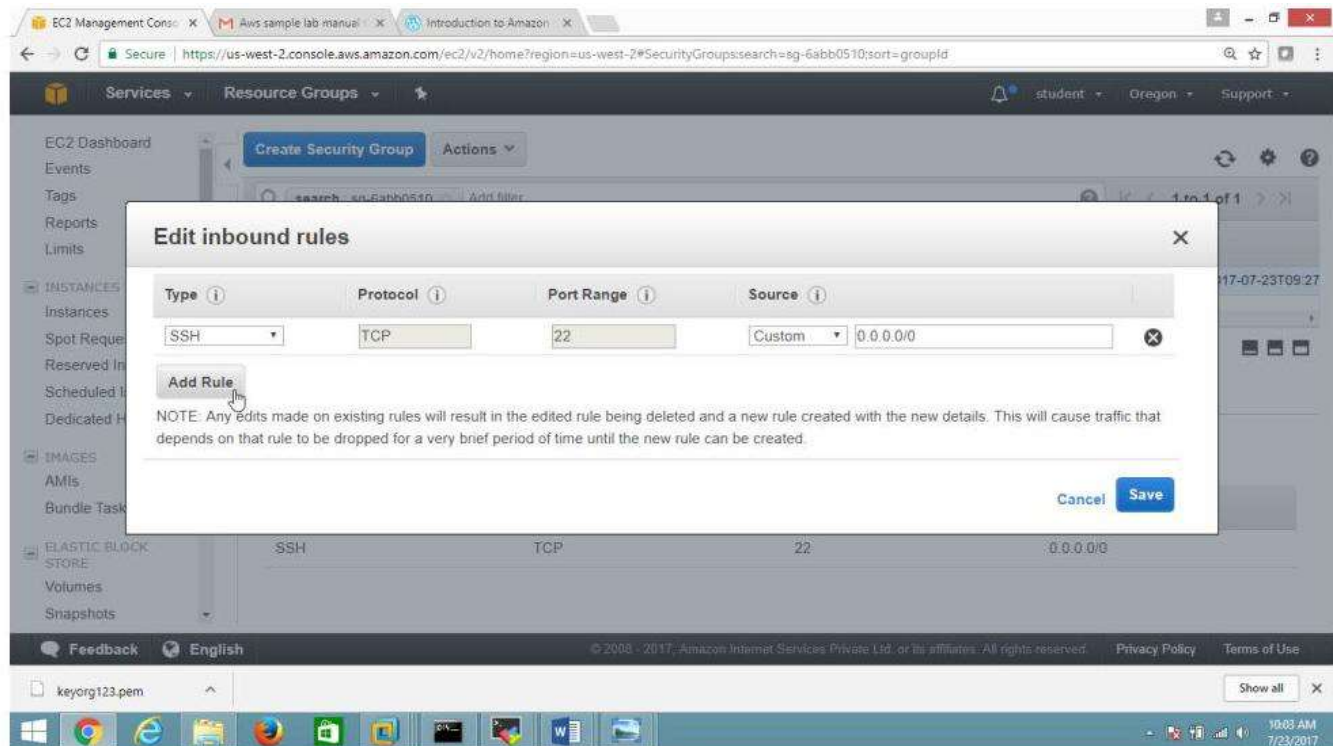
Group ID sg-6abb0510 VPC ID vpc-89c341ee

Click on "Edit" (tab) Button

This screenshot shows the same AWS Management Console page as the previous one, but with the 'Edit' button highlighted by a mouse cursor. The 'Inbound' tab is still selected, and the table below it shows the rule details: SSH (Type), TCP (Protocol), Port Range 22, and Source 0.0.0.0/0. The 'Edit' button is located above the table. The top navigation bar and bottom status bar are consistent with the previous screenshot.

Type	Protocol	Port Range	Source
SSH	TCP	22	0.0.0.0/0

Click on "Add Rule" Button

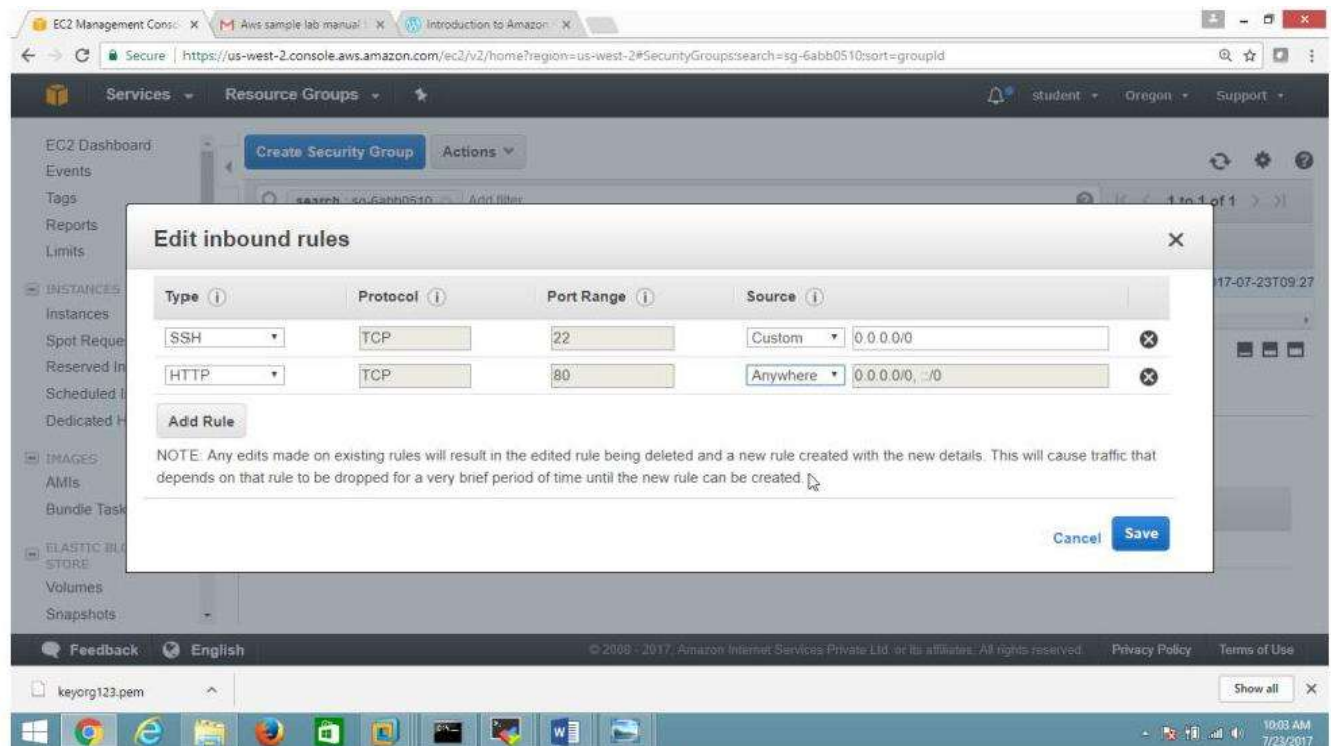


Add HTTP Rule

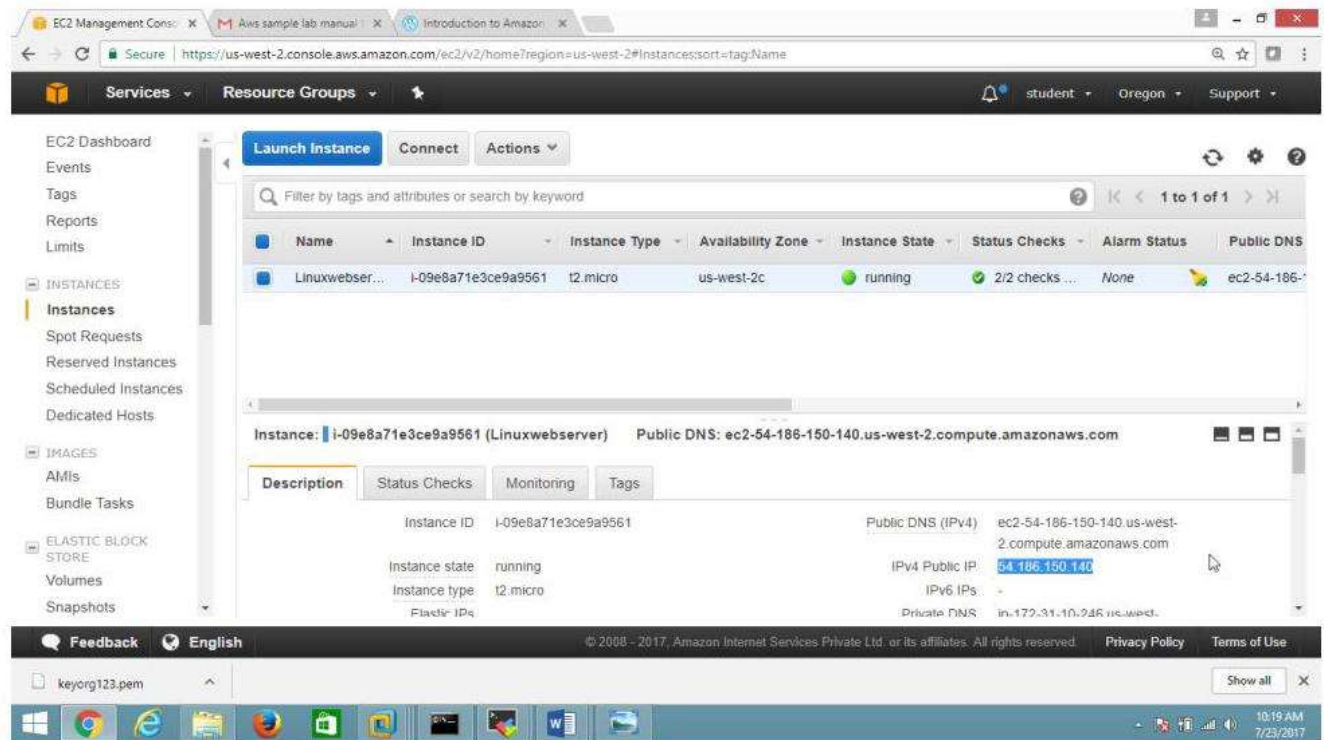
Under Type column select "**HTTP**"

Under Source column select "**Anywhere**"

Click "**Save**" Button

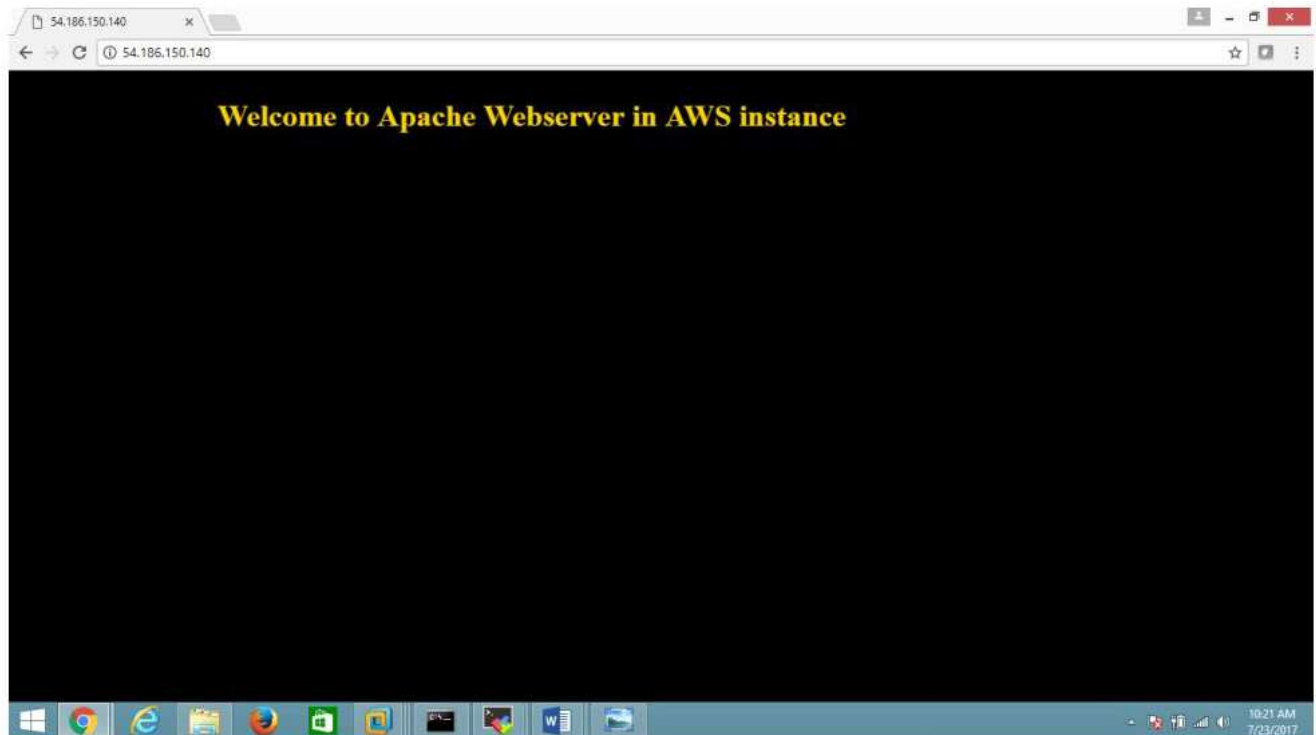


4) Open Browser and provide Webserver instance DNS_name or Pubic IP



Verify

Website is running



Share how to assign Elastic IP Address Step by Step
To Assign Elastic IP Address

Since Public IP given by AWS is not permanent, if the instance is stopped or started again, public IP released by the instance, in this case across internet again cannot visit the same website, so to have permanent public IP, assign Elastic IP

Note: If your instance is terminated or not in use, and Elastic IP is not released then in this case it will be charged, so be careful if you are using and running under free tier usage.

Best practice is launch an instance assign Elastic IP, and before terminating release Elastic IP then terminate the instance

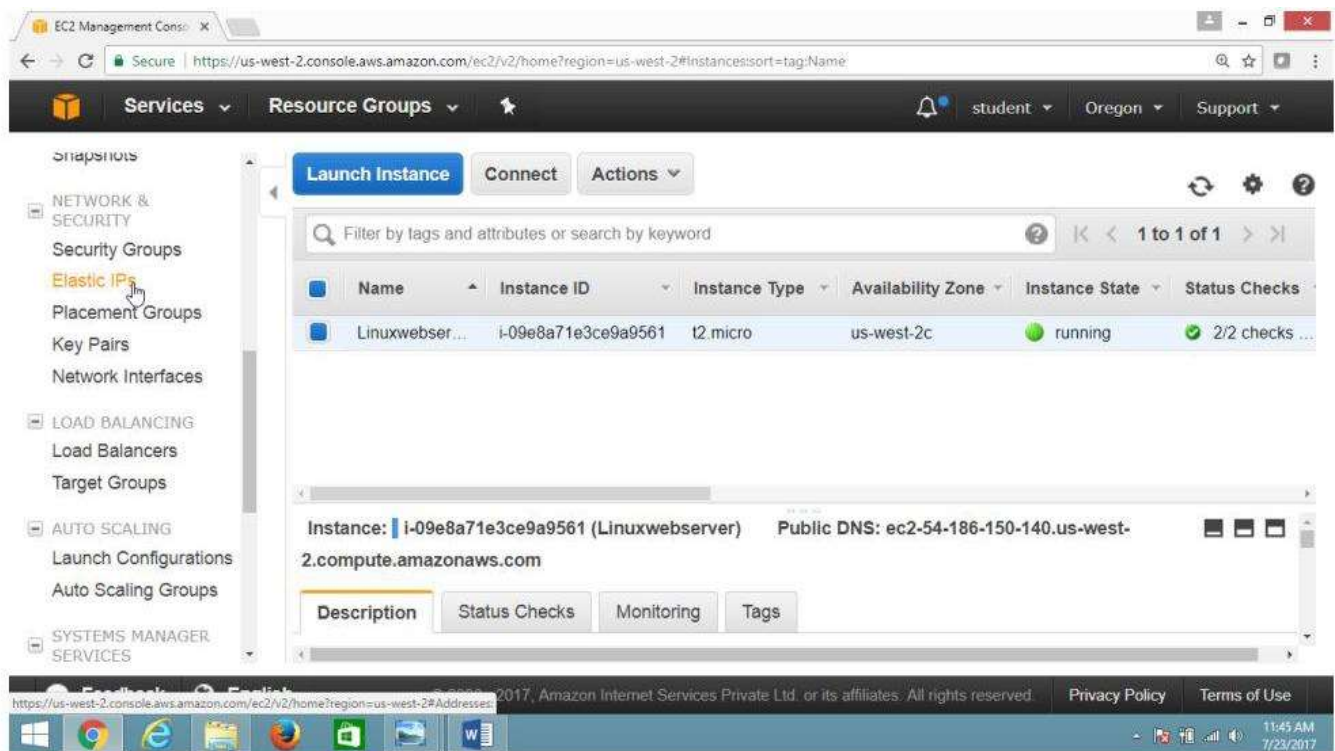
"To assigning Elastic IP to an instance"

Open AWS console

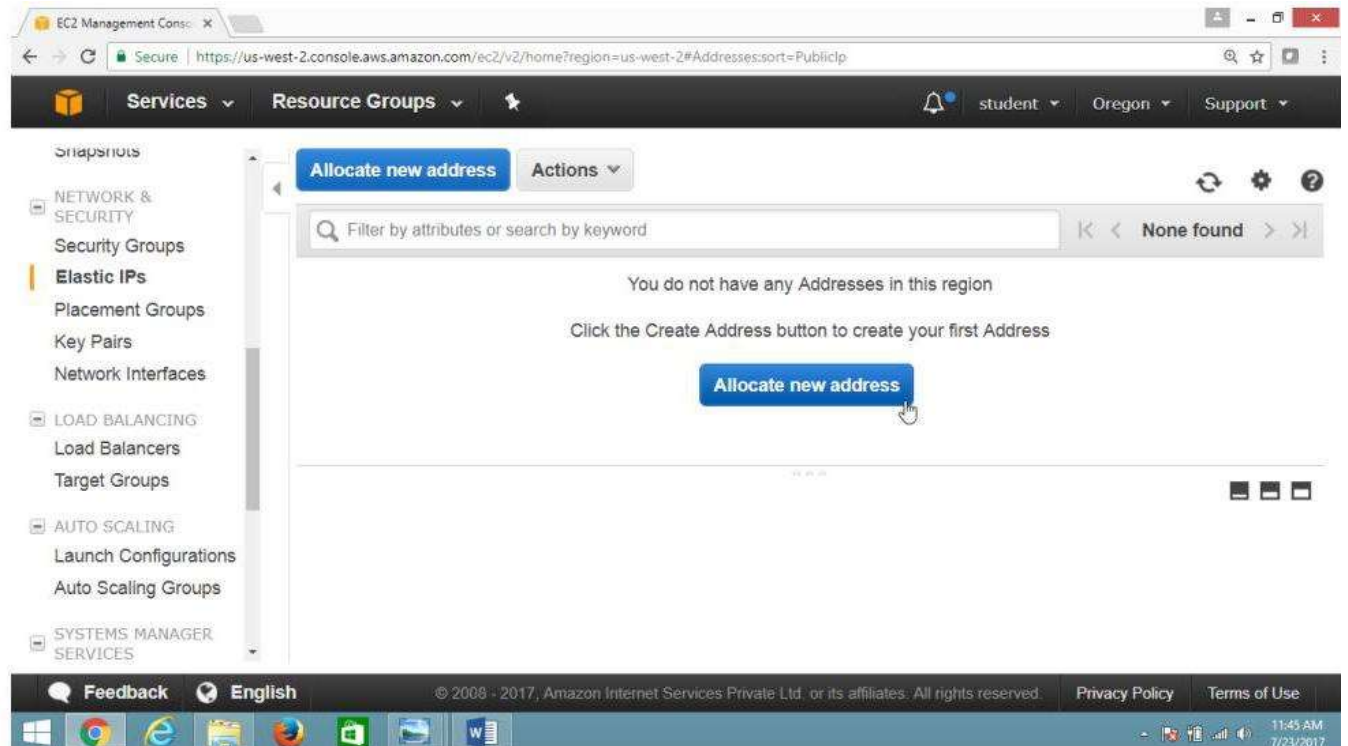
On the EC2 Dashboard Panel

Select "**Network Security**"

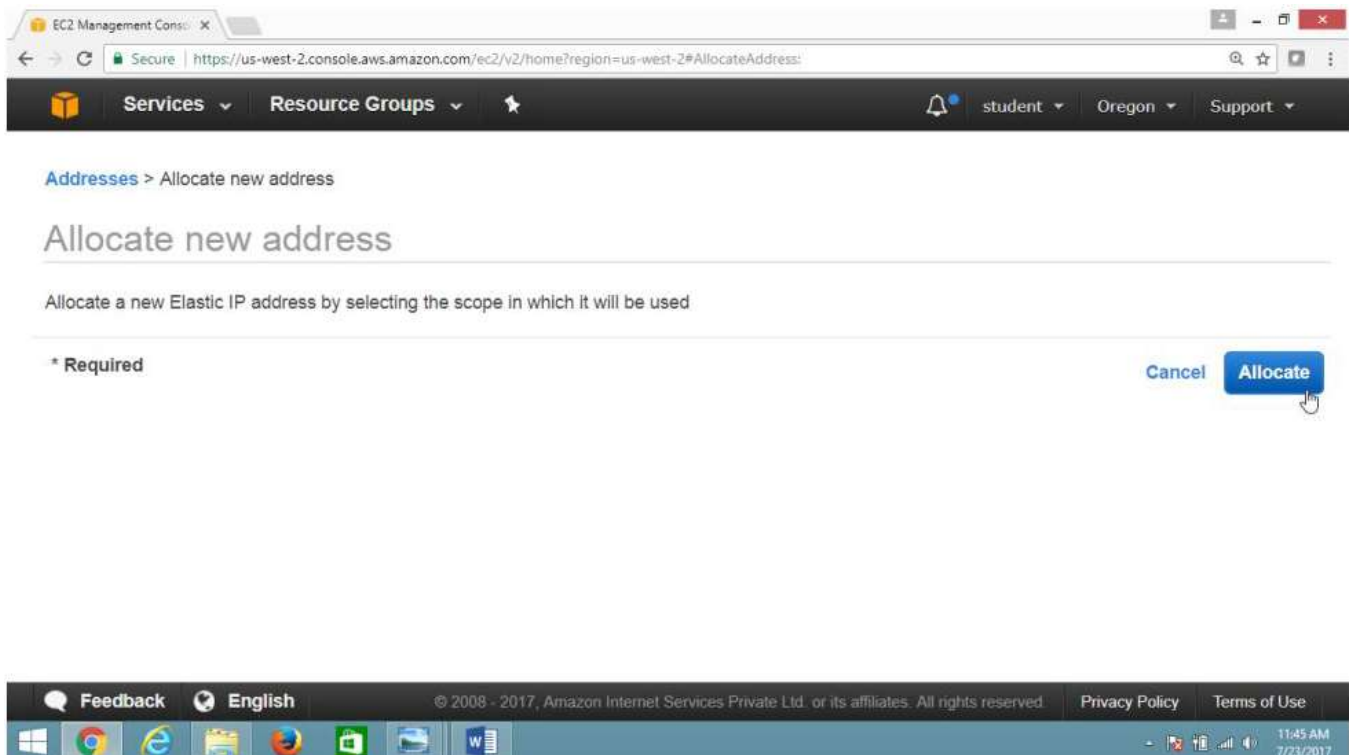
Click on "**Elastic IP**"



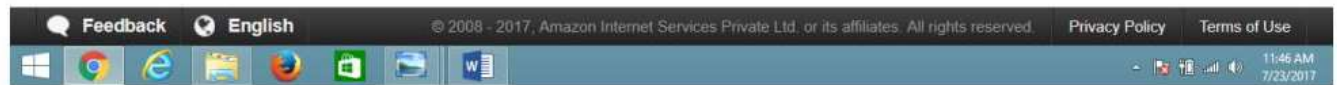
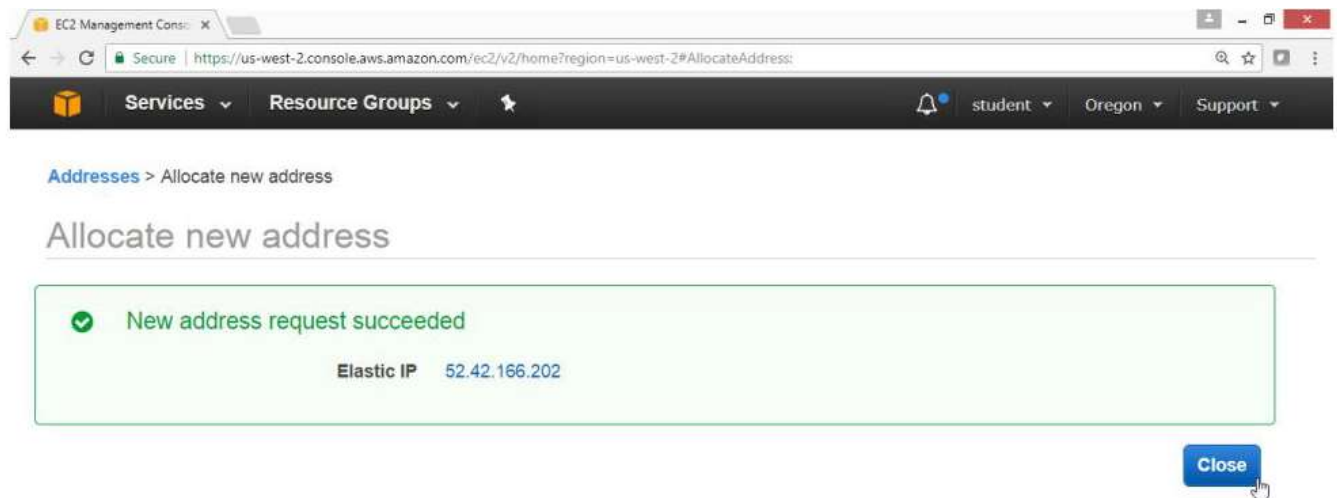
Click on "Allocate new address" Button



Click "Allocate" Button



Click on "Close" Button



Open your Browser and provide your instance DNS name or Elastic Public IP

Verify website is running with elastic IP



To releasing Elastic IP

Open the console "EC2 Dashboard"

Expand "Network Security"

Select "Elastic IP"

Click "Action" Button

Select "Disassociate Address"

The screenshot shows the AWS Management Console interface for Elastic IP addresses. The left sidebar contains navigation links for IMAGES, ELASTIC BLOCK STORE, NETWORK & SECURITY, and LOAD BALANCING. The main content area displays a table of Elastic IP addresses. The first row shows the address 52.42.166.202, which is associated with an instance (i-09e8a71e3ce9a95...). The 'Actions' menu is open, and the 'Disassociate address' option is highlighted. Below the table, the details for the selected address are shown, including the 'Description' tab and the 'Allocation ID' (eipalloc-1723c12a).

Instance	Private IP address	Scope
i-09e8a71e3ce9a95...	172.31.10.246	vpc

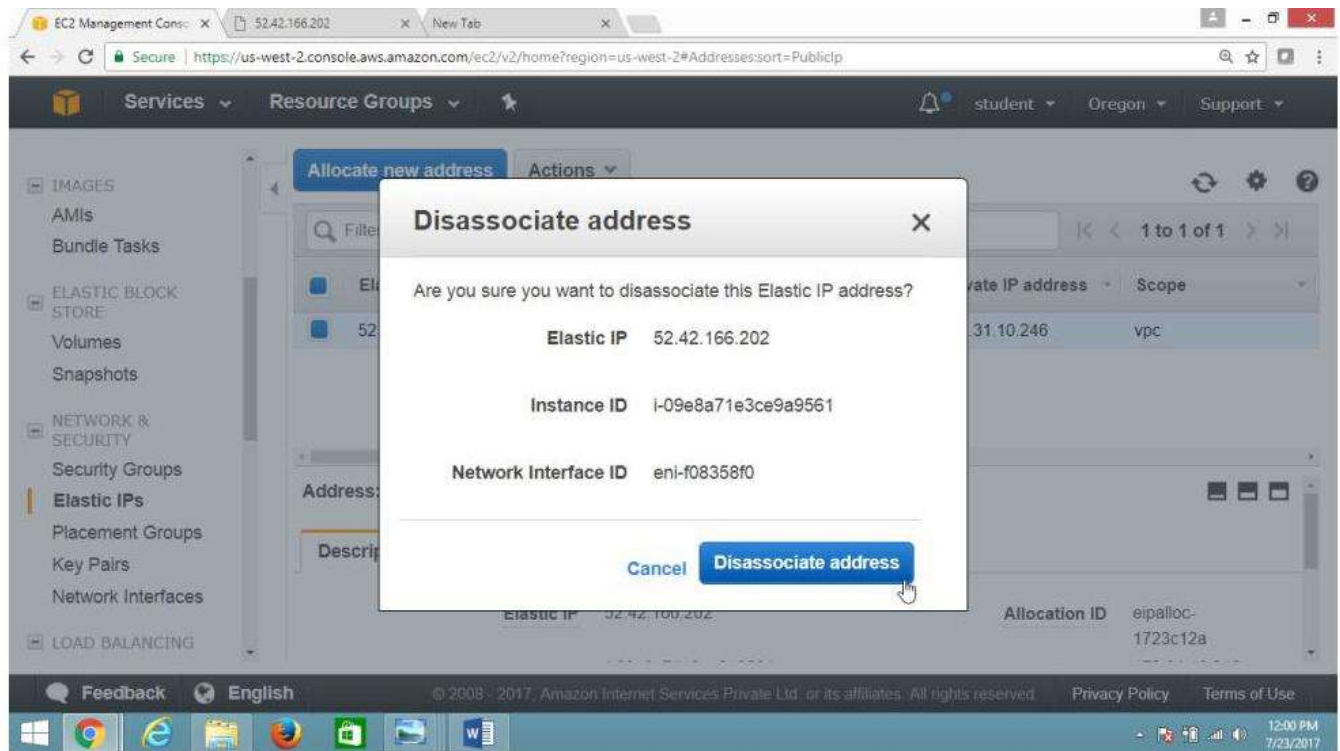
Address: 52.42.166.202

Description

Elastic IP: 52.42.166.202

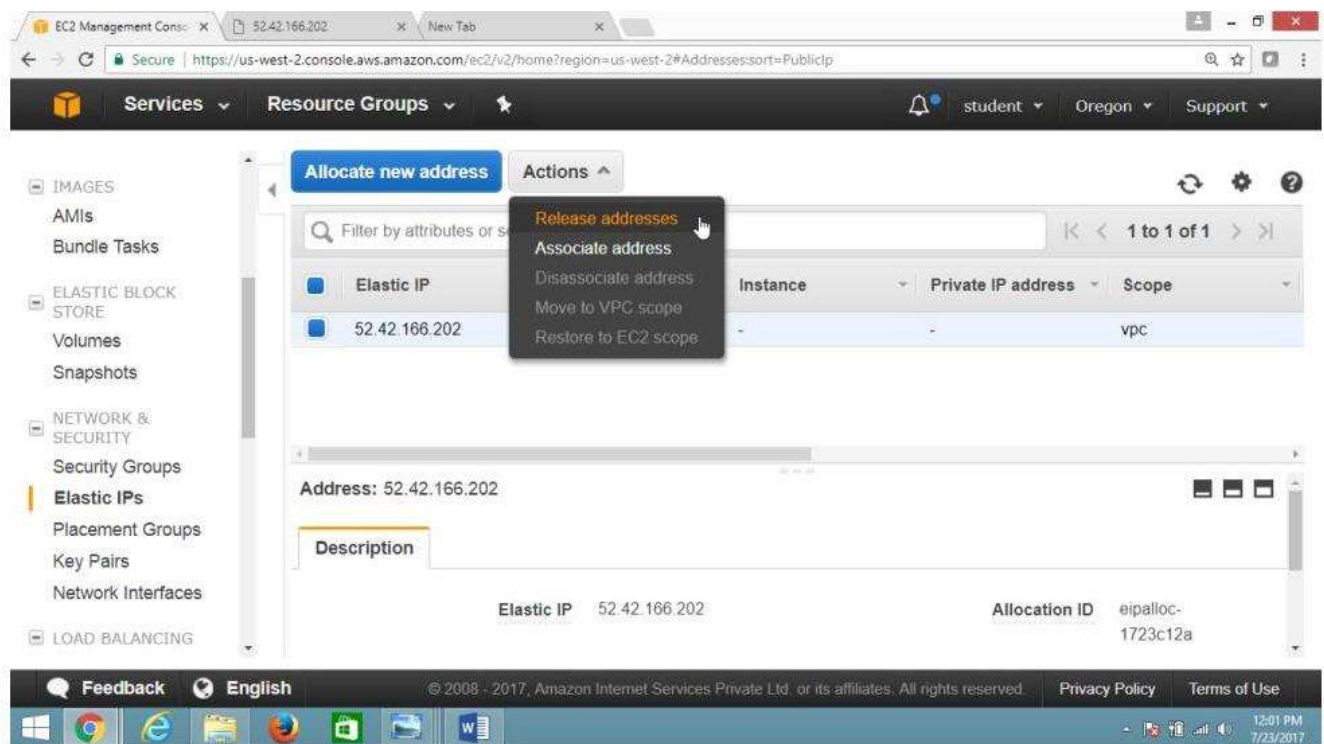
Allocation ID: eipalloc-1723c12a

Click "**Disassociate Address**" Button

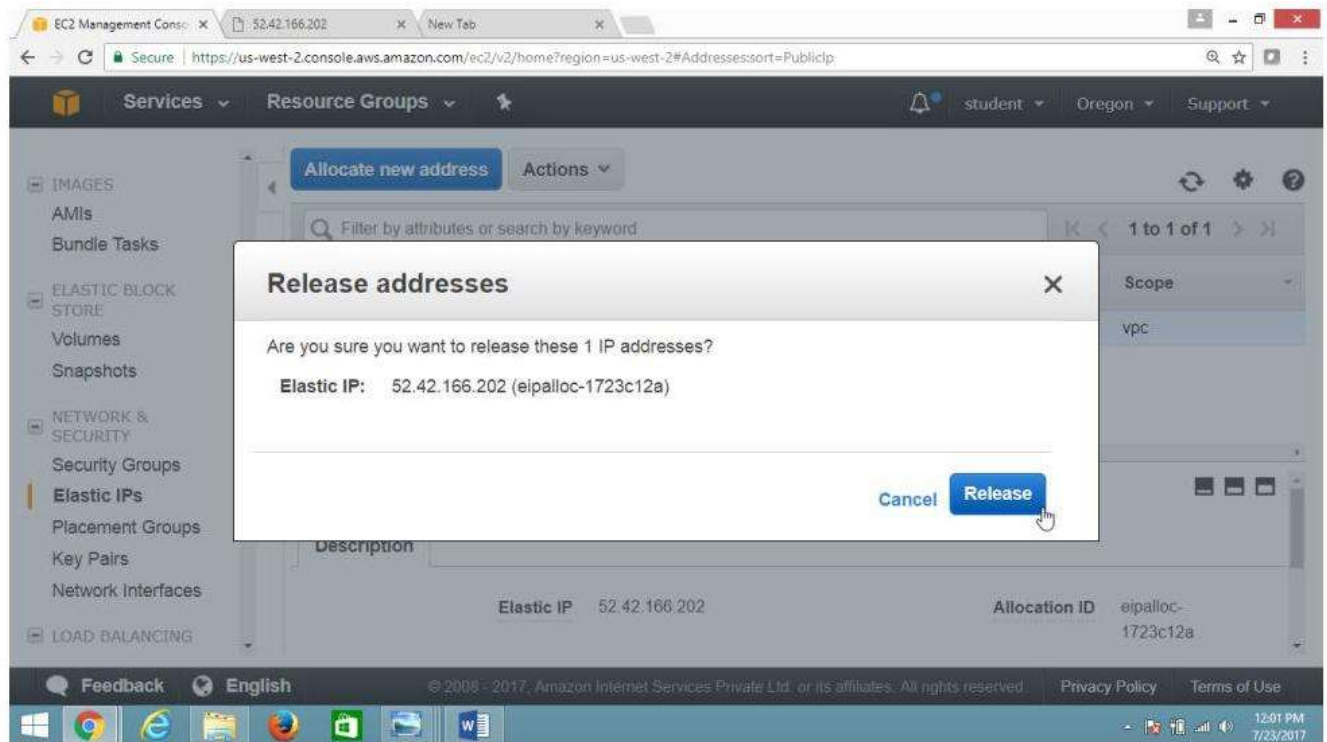


Click "**Action**" Button

Select Release Address



Click "**Release**" button



What is ElastiCache?

ElastiCache is a web service that makes it easy to set up, manage, and scale distributed in-memory cache environments in the cloud.

What is the use of Amazon ElastiCache?

Amazon ElastiCache is mainly used for improving the performance of web applications by caching the information that is frequently accessed. ElastiCache webservice provides very fast access to the information by using in-memory caching.

Behind the scenes, ElastiCache supports open source caching platforms like-Memcached and Redis. We do not have to manage separate caching servers with ElastiCache. We can just add critical pieces of data in ElastiCache to provide very low latency access to applications that need this data very frequently.

Elastic File System

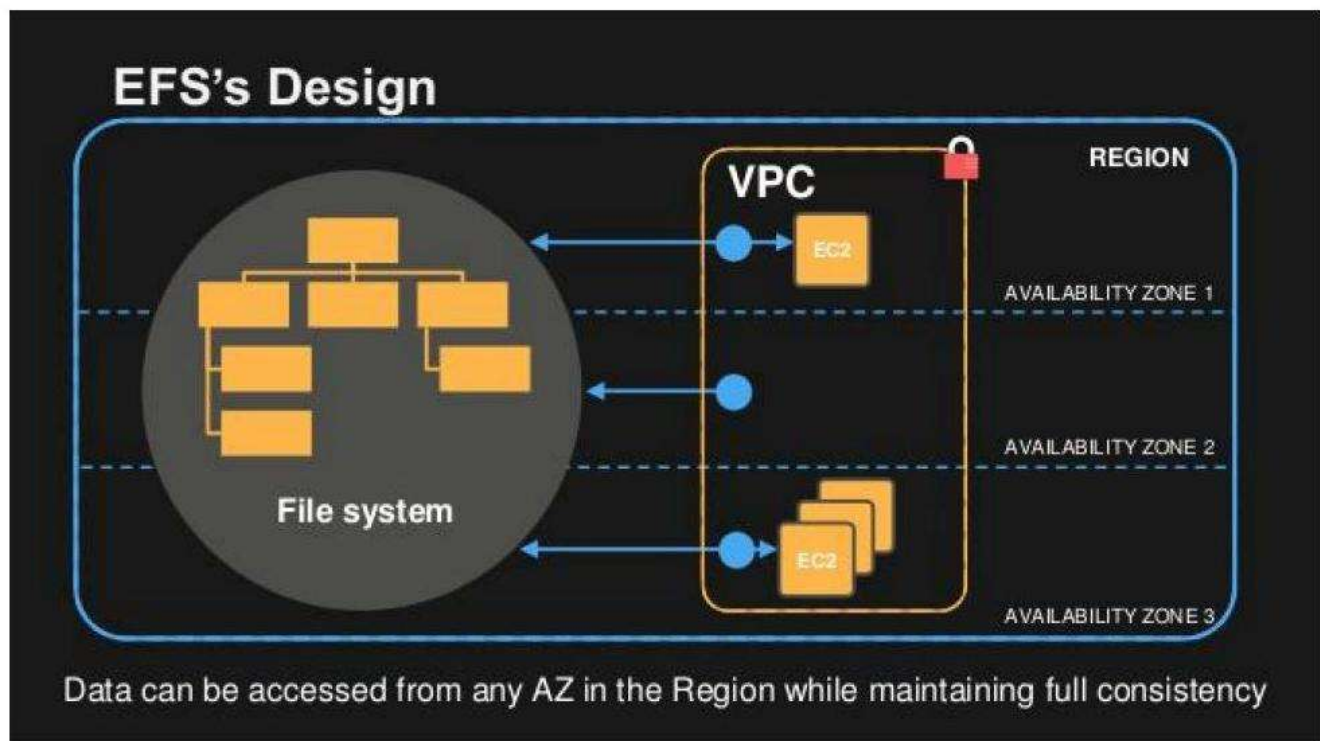
Elastic File System Highlights

- Supports the Network File System Version 4 (NFSv4) protocol
- You only pay for the storage you use (No Pre-Provisioning is required)
- Can Scale up to the Petabytes
- Can support thousands of concurrent NFS connections
- Data is stored across Multiple AZ's within a region
- Read After Write Consistency

Share the Elastic File System Configuration Step by Step?

To configure and use AWS EFS Service

Topology



Pre-requisites

User should have AWS account, or IAM user with Amazon ElasticFullAccess policy.

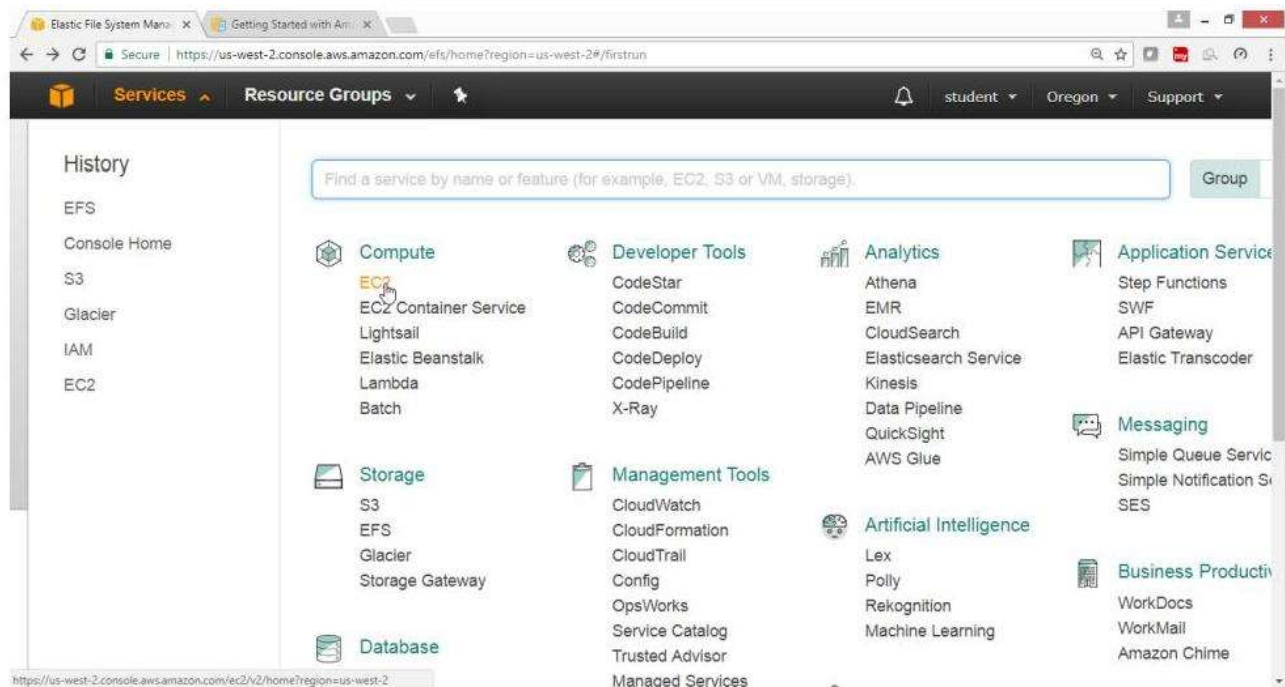
To configure EFS with following task: -

- Step 1 - Create a security group for EFS access
- Step 2 - Create your Amazon EFS File System
- Step 3 - Launch your EC2 Instance
- Step 4 - Create your Amazon EFS File System
- Step 5 - Mount the Amazon EFS File System in your Linux Launch Instance

1) Create a security group for EFS access

Open AWS Console go for [EC2 service](#)

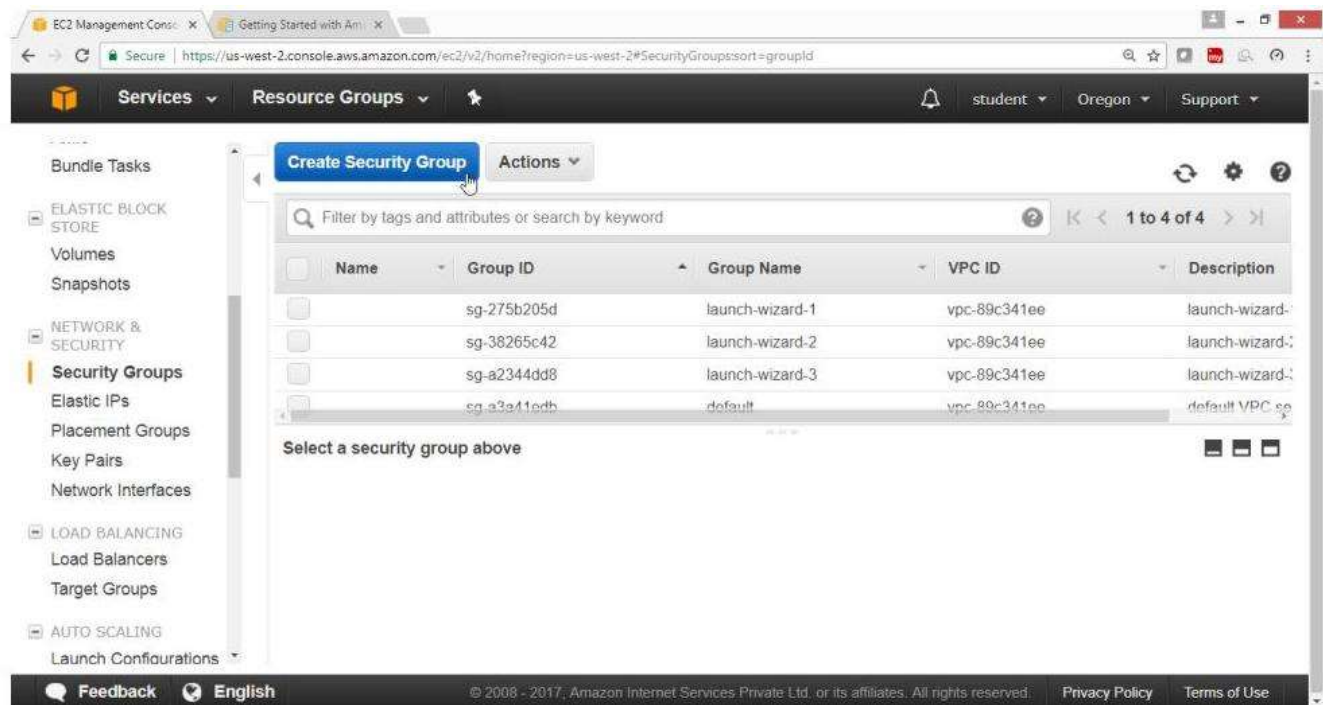
Click on **EC2**



Under EC2 Dashboard go for **Network & Security**

Select **Security Groups**

Click on **Create Security Group**



Under "**Create Security Group**" wizard

Give Follow values: -

Security group name -> NFSsecurity2

Description -> NFSrule2

VPC -> take default

Select Inbound

Type -> All Traffic

Source -> Anywhere

Click on [Create button](#)

The screenshot shows the 'Create Security Group' dialog in the AWS Management Console. The dialog has a title bar with a close button (X) and a header area with navigation links: Services, Resource Groups, and a search icon. The main content area is divided into sections for defining the security group. The 'Security group name' field is set to 'NFSsecurity2', the 'Description' field is set to 'NFSrule2', and the 'VPC' dropdown is set to 'vpc-89c341ee | default-vpc-oregon (default)'. Below these fields, there are tabs for 'Inbound' and 'Outbound' rules, with 'Inbound' currently selected. A table for 'Security group rules' is displayed with columns for 'Type', 'Protocol', 'Port Range', and 'Source'. The first rule is configured with 'All traffic' for Type, 'All' for Protocol, '0 - 65535' for Port Range, and 'Anywhere' for Source. The 'Source' field is expanded to show the IP range '0.0.0.0/0, ::/0'. An 'Add Rule' button is located below the table. At the bottom right of the dialog, there are 'Cancel' and 'Create' buttons. The footer of the console shows 'Feedback', 'English', and copyright information for Amazon Internet Services Private Ltd.

Services Resource Groups

student Oregon Support

Create Security Group

Security group name: NFSsecurity2

Description: NFSrule2

VPC: vpc-89c341ee | default-vpc-oregon (default)

Security group rules:

Inbound Outbound

Type	Protocol	Port Range	Source
All traffic	All	0 - 65535	Anywhere

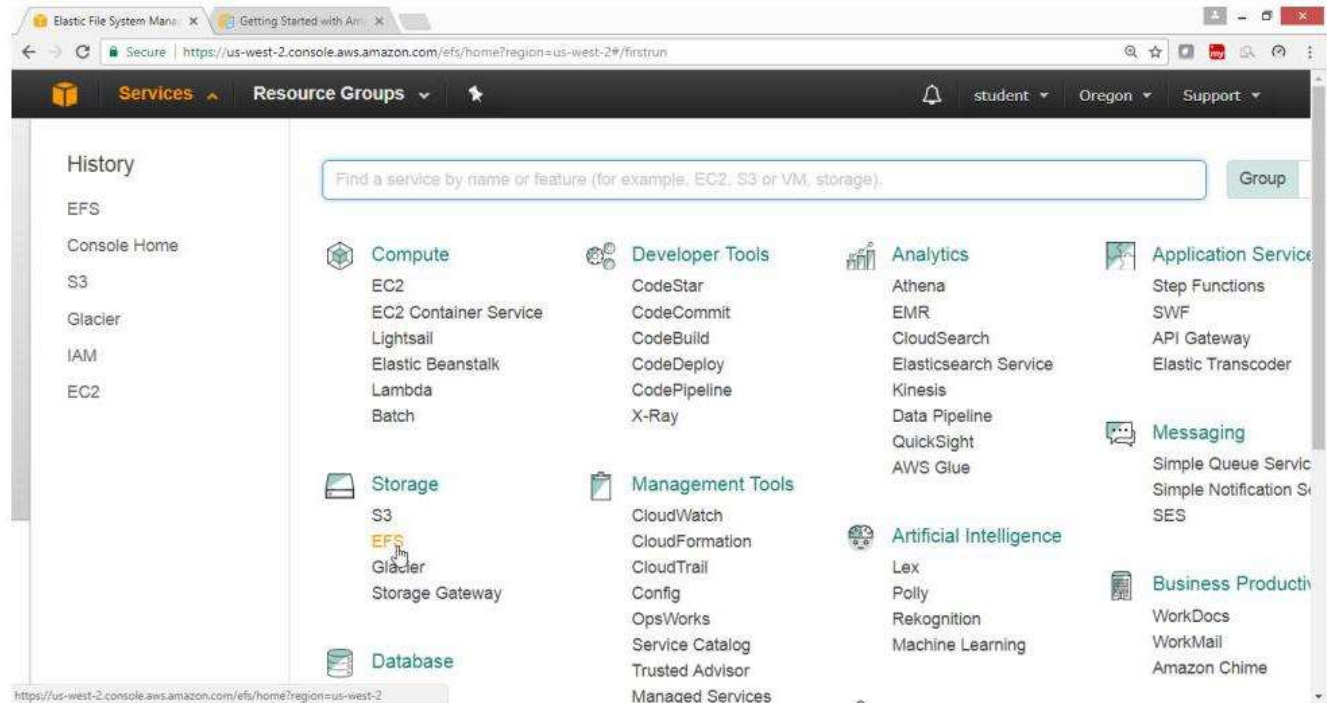
0.0.0.0/0, ::/0

Add Rule

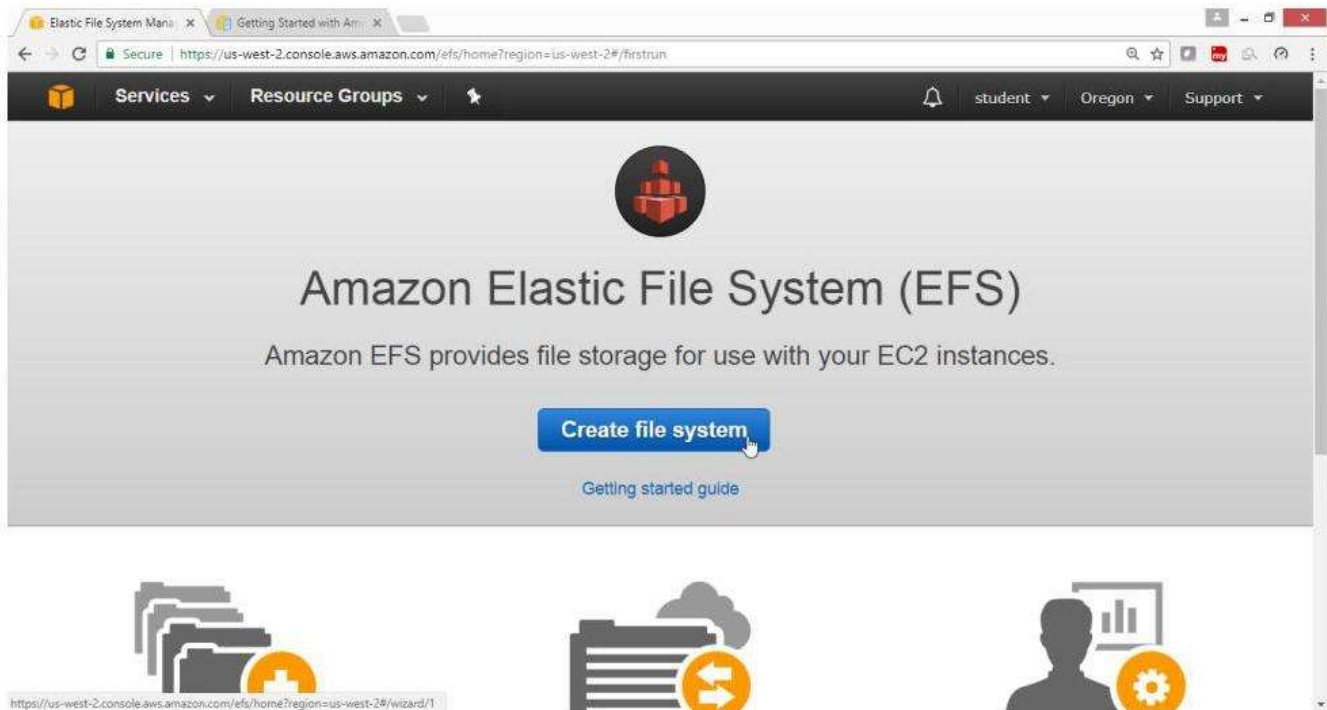
Cancel Create

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

2) Create your Amazon EFS File System



Click on **"Create File System"** button



Select Default VPC

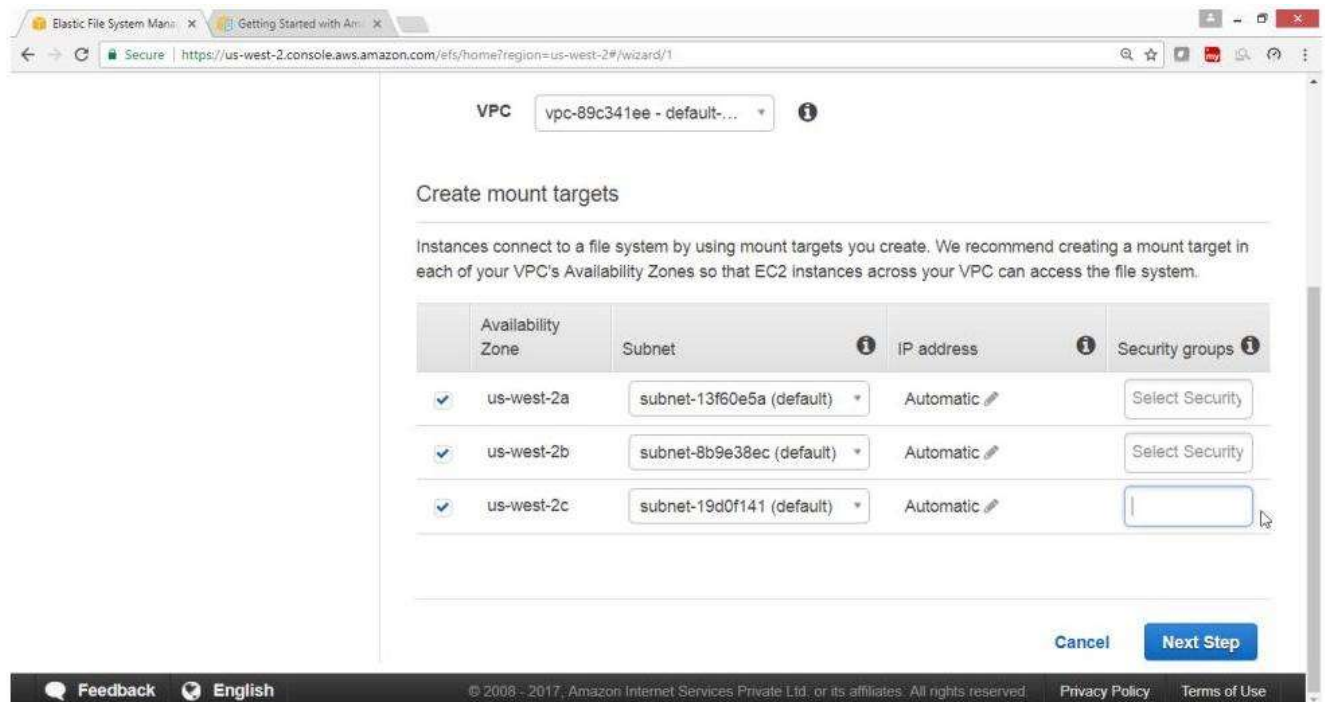
The screenshot shows the AWS Elastic File System console. The top navigation bar includes 'Services', 'Resource Groups', and a user profile 'student' in the 'Oregon' region. The main heading is 'Create file system'. On the left, a sidebar lists three steps: 'Step 1: Configure file system access' (active), 'Step 2: Configure optional settings', and 'Step 3: Review and create'. The main content area is titled 'Configure file system access' and contains the following text: 'An Amazon EFS file system is accessed by EC2 instances running inside one of your VPCs. Instances connect to a file system by using a network interface called a mount target. Each mount target has an IP address, which we assign automatically or you can specify.' Below this text, there is a 'VPC' dropdown menu showing 'vpc-89c341ee - default-...'. Further down, a section titled 'Create mount targets' explains that instances connect to a file system using mount targets and recommends creating a mount target in each of the VPC's Availability Zones. Below this explanation is a table with columns: 'Availability Zone', 'Subnet', 'IP address', and 'Security groups'. The table is currently empty.

Remove all security Groups

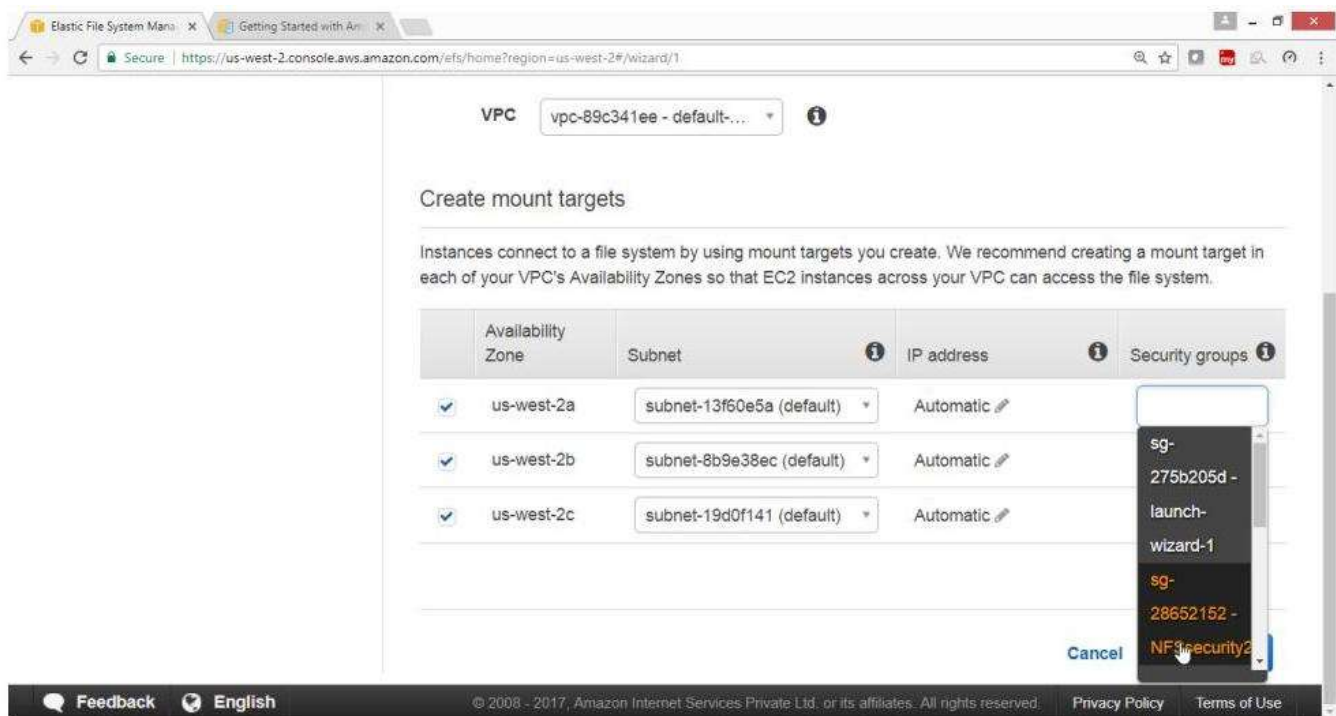
This screenshot shows the 'Create mount targets' section of the AWS Elastic File System console. The explanatory text is the same as in the previous screenshot. The table below now contains three rows, each representing an Availability Zone. Each row has a checked checkbox in the 'Availability Zone' column, a 'Subnet' dropdown, an 'IP address' field set to 'Automatic', and a 'Security groups' dropdown. The 'Security groups' dropdown for each row is open, showing a list with 'sg-a3a41edb - default' selected. The rows are for 'us-west-2a', 'us-west-2b', and 'us-west-2c'.

Availability Zone	Subnet	IP address	Security groups
<input checked="" type="checkbox"/> us-west-2a	subnet-13f60e5a (default)	Automatic	sg-a3a41edb - default
<input checked="" type="checkbox"/> us-west-2b	subnet-8b9e38ec (default)	Automatic	sg-a3a41edb - default
<input checked="" type="checkbox"/> us-west-2c	subnet-19d0f141 (default)	Automatic	sg-a3a41edb - default

Verify that **all security groups** go deleted



Now add **NFSsecurity2** group in A.Z



Verify that all Security Groups are added

Click on Next Step

Create mount targets

Instances connect to a file system by using mount targets you create. We recommend creating a mount target in each of your VPC's Availability Zones so that EC2 instances across your VPC can access the file system.

	Availability Zone	Subnet	IP address	Security groups
<input checked="" type="checkbox"/>	us-west-2a	subnet-13f60e5a (default)	Automatic	sg-28652152 NFSsecurity2
<input checked="" type="checkbox"/>	us-west-2b	subnet-8b9e38ec (default)	Automatic	sg-28652152 NFSsecurity2
<input checked="" type="checkbox"/>	us-west-2c	subnet-19d0f141 (default)	Automatic	sg-28652152 NFSsecurity2

Cancel

Next Step

Provide tags

Key -> Name

Value -> NFSHyd1

Drag Down

The screenshot shows the 'Add tags' step in the AWS Elastic File System console. The browser address bar indicates the URL: <https://us-west-2.console.aws.amazon.com/efs/home?region=us-west-2#/wizard/2>. The page title is 'Step 3: Review and create'. The 'Add tags' section explains that tags are case-sensitive key-value pairs and recommends a tag with key = Name. A table with two columns, 'Key' and 'Value', is shown. The 'Key' column has a text input field containing 'Name'. The 'Value' column has a text input field containing 'NFSHyd1'. To the right of the 'Value' field is a 'Remove' button. Below the table is an 'Add New Key' button. The 'Choose performance mode' section follows, recommending 'General Purpose' performance mode for most file systems. It states that 'Max I/O' performance mode is optimized for applications where tens, hundreds, or thousands of EC2 instances are accessing the file system. Two radio buttons are present: 'General Purpose (default)' (selected) and 'Max I/O'.

Key	Value	Remove
Name	NFSHyd1	

Choose performance mode

We recommend **General Purpose** performance mode for most file systems. **Max I/O** performance mode is optimized for applications where tens, hundreds, or thousands of EC2 instances are accessing the file system — it scales to higher levels of aggregate throughput and operations per second with a tradeoff of slightly higher latencies for file operations.

☒ General Purpose (default)

☐ Max I/O

Select General Purpose

Click on **Next Step**

The screenshot shows the 'Enable encryption' step in the AWS Elastic File System console. The browser address bar indicates the URL: <https://us-west-2.console.aws.amazon.com/efs/home?region=us-west-2#/wizard/2>. The page title is 'Step 3: Review and create'. The 'Enable encryption' section explains that if encryption is enabled, all data on the file system will be encrypted at rest. It states that you can select a KMS key from your account to protect your file system, or you can provide the ARN of a key from a different account. Encryption can only be enabled during file system creation. A link 'Learn more' is provided. A checkbox labeled 'Enable encryption' is shown and is checked. At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Next Step' (highlighted with a mouse cursor).

Enable encryption

If you enable encryption for your file system, all data on your file system will be encrypted at rest. You can select a KMS key from your account to protect your file system, or you can provide the ARN of a key from a different account. Encryption can only be enabled during file system creation. [Learn more](#)

☒ Enable encryption

[Cancel](#) [Previous](#) [Next Step](#)

NFSHyd1 filesystem got selected

Click on **Create File System**

Elastic File System Manag...Getting Started with Am...

Secure | https://us-west-2.console.aws.amazon.com/efs/home?region=us-west-2#/wizard/3

VPC

Zone

Subnet

IP address

Security groups

vpc-89c341ee - default-vpc-oregon (default)

us-west-2a

subnet-13f60e5a (default)

Automatic

sg-28652152 - NFSsecurity2

us-west-2b

subnet-8b9e38ec (default)

Automatic

sg-28652152 - NFSsecurity2

us-west-2c

subnet-19d0f141 (default)

Automatic

sg-28652152 - NFSsecurity2

Optional settings

Tags

Name: NFSShyd1

Performance mode

General Purpose (default)

Encrypted

No

Cancel

Previous

Create File System

FeedbackEnglish

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

Verify

Elastic File System Manag...Getting Started with Am...

Secure | https://us-west-2.console.aws.amazon.com/efs/home?region=us-west-2#/filesystems/fs-53f822fa

ServicesResource GroupsstudentOregonSupport

File systems

File systems

Success!

You have created a file system. You can mount your file system from an EC2 instance with an NFSv4.1 client installed. You can also mount your file system from an on-premises server over an AWS Direct Connect connection. Click [here](#) for EC2 mount instructions, and [here](#) for on-premises mount instructions.

Create file systemActions

	Name	File system ID	Metered size	Number of mount targets	Creation date
	NFSShyd1	fs-53f822fa	6.0 KiB	3	2017-08-15T06:16:55Z

Other details

Owner ID 523251683217

Life cycle state Available

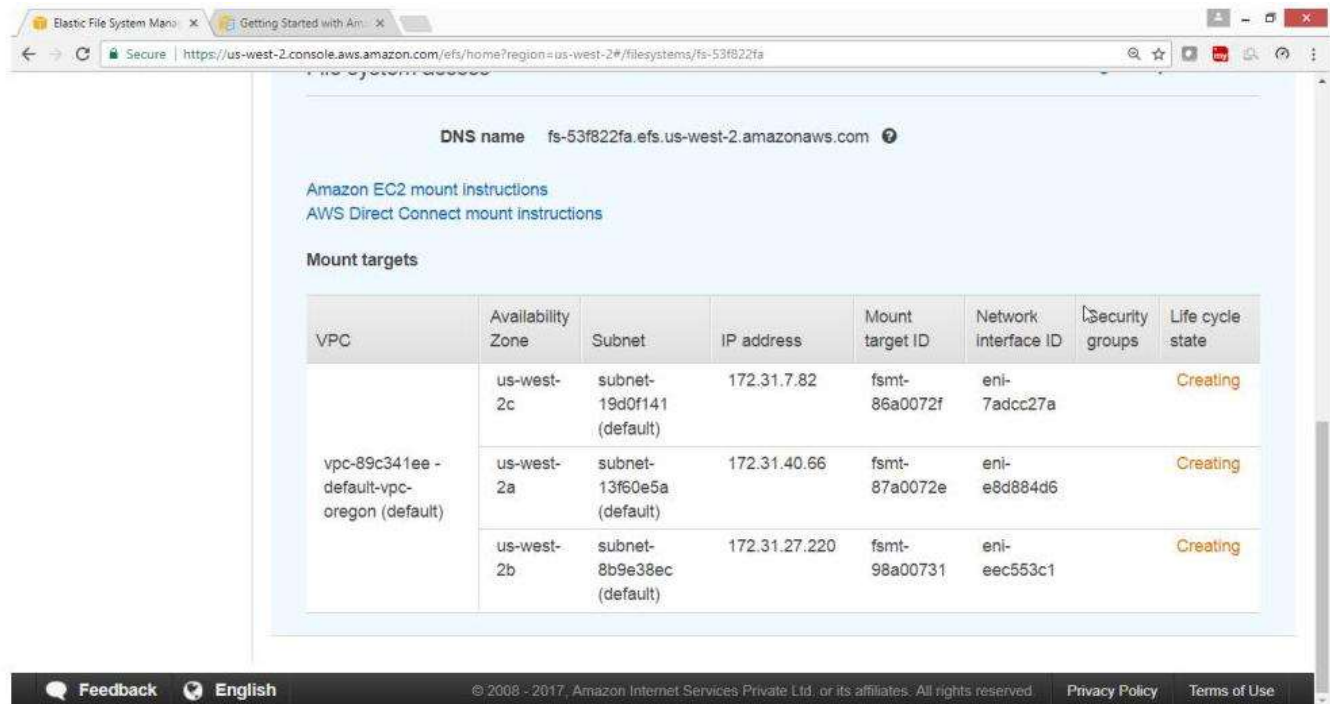
Tags

Name: NFSShyd1

Manage tags

Drag Down

Verify that Life Cycle state is **Creating**, it takes few minutes.

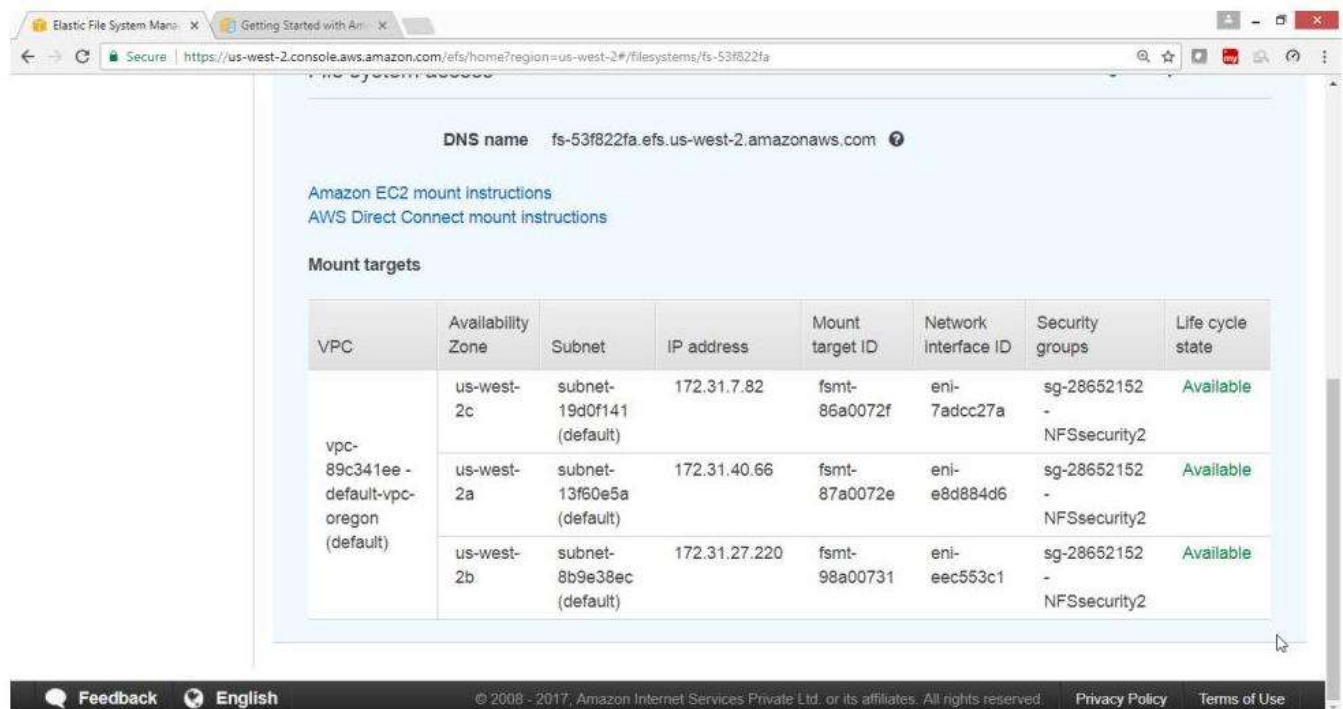


The screenshot shows the AWS Elastic File System console for file system fs-53f822fa. The DNS name is fs-53f822fa.efs.us-west-2.amazonaws.com. Below the DNS name are links for Amazon EC2 mount instructions and AWS Direct Connect mount instructions. The 'Mount targets' section contains a table with three rows, all showing a 'Creating' life cycle state.

VPC	Availability Zone	Subnet	IP address	Mount target ID	Network interface ID	Security groups	Life cycle state
vpc-89c341ee - default-vpc-oregon (default)	us-west-2c	subnet-19d0f141 (default)	172.31.7.82	fsmt-86a0072f	eni-7adcc27a		Creating
	us-west-2a	subnet-13f60e5a (default)	172.31.40.66	fsmt-87a0072e	eni-e8d884d6		Creating
	us-west-2b	subnet-8b9e38ec (default)	172.31.27.220	fsmt-98a00731	eni-eec553c1		Creating

At the bottom of the console, there is a footer with 'Feedback', 'English', '© 2008 - 2017, Amazon Internet Services Private Ltd, or its affiliates. All rights reserved.', 'Privacy Policy', and 'Terms of Use'.

Verify that Life cycle state is Available



DNS name fs-53f822fa.efs.us-west-2.amazonaws.com ⓘ

[Amazon EC2 mount instructions](#)
[AWS Direct Connect mount instructions](#)

Mount targets

VPC	Availability Zone	Subnet	IP address	Mount target ID	Network interface ID	Security groups	Life cycle state
vpc-89c341ee - default-vpc-oregon (default)	us-west-2c	subnet-19d0f141 (default)	172.31.7.82	fsmt-86a0072f	eni-7adcc27a	sg-28652152 - NFSsecurity2	Available
	us-west-2a	subnet-13f50e5a (default)	172.31.40.66	fsmt-87a0072e	eni-e8d884d6	sg-28652152 - NFSsecurity2	Available
	us-west-2b	subnet-8b9e38ec (default)	172.31.27.220	fsmt-98a00731	eni-eec553c1	sg-28652152 - NFSsecurity2	Available

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Step 3. Now launch Linux instance & Mount the Amazon EFS File System

Log to Linux instance by using mobaxterm client

```
[2017-08-15 12:01.25] /drives/e/awskeys  
[shaikf.pc_mas] > ssh -i "studentorg.pem" ec2-user@ec2-54-213-7-42.us-west-2.compute.amazonaws.com
```

Run the following commands

```
[ec2-user@ip-172-31-45-138 ~]$ sudo su  
[root@ip-172-31-45-138 ec2-user]#  
[root@ip-172-31-45-138 ec2-user]# yum install nfs-utils  
[root@ip-172-31-45-138 ec2-user]#  
[root@ip-172-31-45-138 ec2-user]# mkdir /opt/oracledata  
[root@ip-172-31-45-138 ec2-user]# mount -t nfs4 fs-53f822fa.efs.us-west-2.amazonaws.com:/ /opt/oracledata  
[root@ip-172-31-45-138 ec2-user]#
```

Verify it is mounted & Check the last line

```
proc on /proc type proc (rw,relatime)  
sysfs on /sys type sysfs (rw,relatime)  
devtmpfs on /dev type devtmpfs (rw,relatime,size=499756k,nr_inodes=124939,mode=755)  
devpts on /dev/pts type devpts (rw,relatime,gid=5,mode=620,ptmxmode=000)  
tmpfs on /dev/shm type tmpfs (rw,relatime)  
/dev/xvda1 on / type ext4 (rw,noatime,data=ordered)  
devpts on /dev/pts type devpts (rw,relatime,gid=5,mode=620,ptmxmode=000)  
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw,relatime)  
fs-53f822fa.efs.us-west-2.amazonaws.com:/ on /opt/oracledata type nfs4 (rw,relatime,vers=4.0,rsize=1048576,wsiz  
e=1048576,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientaddr=172.31.45.138,local_lock=none,addr=172  
.31.40.66)  
[root@ip-172-31-45-138 ec2-user]#
```


Elastic Block Store

EBS Highlights

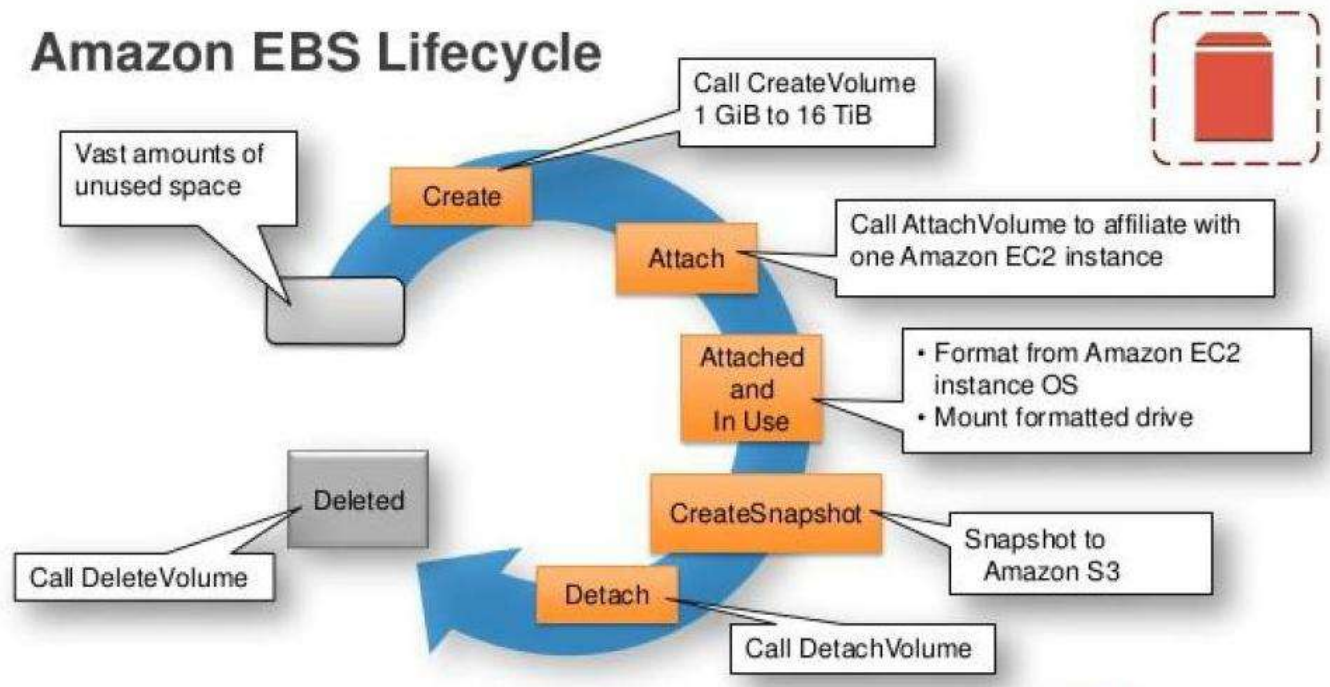
Amazon EBS allows you to create storage volumes and attach them to Amazon EC2 instances. Once attached, you can create a file system on top of these volumes, run a database, or use them in any other way you would use a block device. Amazon EBS volumes are replicated in a specific availability zone, where they are automatically replicated to protect you from the failure of a single component.

- EBS consists of five volume types: -
 - **SSD, General Purpose** - GP2 (Up to 10,000 IOPS)
 - **SSD, Provisioned IOPS** - IO1 (More than 10,000 IOPS)
 - **HDD, Throughput Optimized** - ST1 - Frequently accessed workloads
 - **HDD, Cold** - SC1 - Less frequently accessed data
 - **HDD, Magnetic** - Standard - cheap, infrequently accessed storage
- You cannot mount 1 EBS volume to multiple EC2 instances instead use EFS
- Termination Protection is turned off by default, you must turn it on
- On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated
- Root Volumes cannot be encrypted by default, you need a third-party tool (such as bit locker etc.,) to encrypt the root volume

Share the Elastic File System Configuration Step by Step?

To configure and use EBS Service

Topology



Pre-requisites

User should have AWS account, or IAM user with EC2FullAccess

User should have basic knowledge of managing partitions in Windows or Linux

To configure EBS with following task: -

- Create EBS Volume
- Attaching and Detaching EBS volume
- Expanding the size of EBS volume
- Taking the snapshot of EBS volume

1. To create an EBS Volume

Open the Amazon Console

Select Compute, Choose EC2 Service

On the EC2 Dashboard panel

Choose "Elastic Block Store" Click on Volumes

The screenshot displays the AWS Management Console interface for the EC2 service in the US West (Oregon) region. The left-hand navigation pane shows the 'EC2 Dashboard' with various links. Under the 'ELASTIC BLOCK STORE' section, the 'Volumes' link is highlighted with a mouse cursor. The main content area shows the 'Resources' section, indicating that there are 0 Volumes. Below this, the 'Create Instance' section is visible, featuring a 'Launch Instance' button and a note about the region. The right-hand sidebar contains 'Account Attributes' and 'Additional Information' sections. The footer of the console shows the URL and copyright information.

aws Services Resource Groups student Oregon Support

EC2 Dashboard

- Events
- Tags
- Reports
- Limits
- INSTANCES
 - Instances
 - Spot Requests
 - Reserved Instances
 - Scheduled Instances
 - Dedicated Hosts
- IMAGES
 - AMIs
 - Bundle Tasks
- ELASTIC BLOCK STORE
 - Volumes**
 - Snapshots
- NETWORK & SECURITY

Resources

You are using the following Amazon EC2 resources in the US West (Oregon) 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 US West (Oregon) region.

Account Attributes

- Supported Platforms
 - VPC
- Default VPC
 - vpc-89c341ee
- Resource ID length management

Additional Information

- Getting Started Guide
- Documentation
- All EC2 Resources
- Forums
- Pricing
- Contact Us

AWS Marketplace

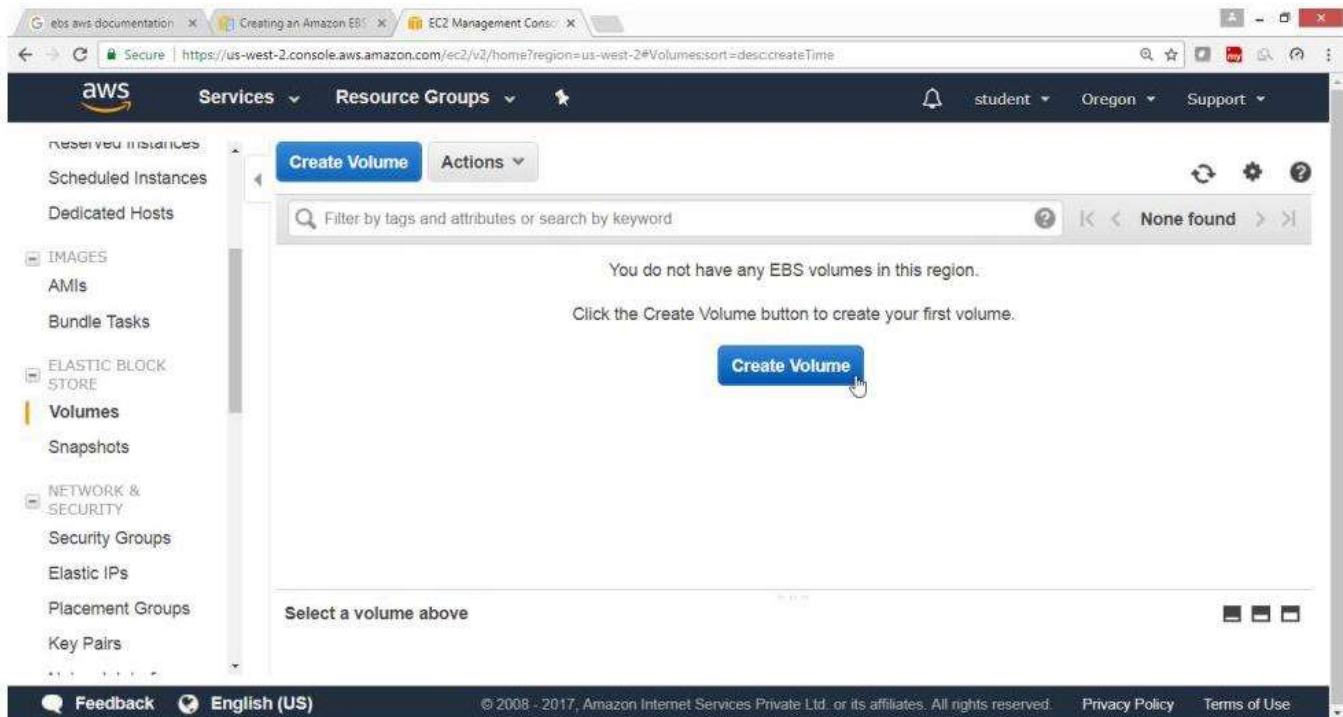
Find free software trial products in the AWS Marketplace from the [EC2 Launch Wizard](#). Or try these popular AMIs:

Service Health Scheduled Events

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

<https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#Volumes>

Click on "Create Volumes" Button



In the “Create Volume” dialog box,

Volume Type -> General purpose SSD(GP2)

Size (GiB) -> 2 GiB

IOPS -> 100/300

Throughput (MB/s) -> Not Applicable

Availability Zone -> us-west-2a (As per your requirement)

Leave the remaining as defaults

Click on “Create Volume” Button

The screenshot shows the AWS 'Create Volume' console. At the top is the AWS navigation bar with the logo, 'Services', 'Resource Groups', and user account information. The main heading is 'Create Volume'. The form includes the following fields:

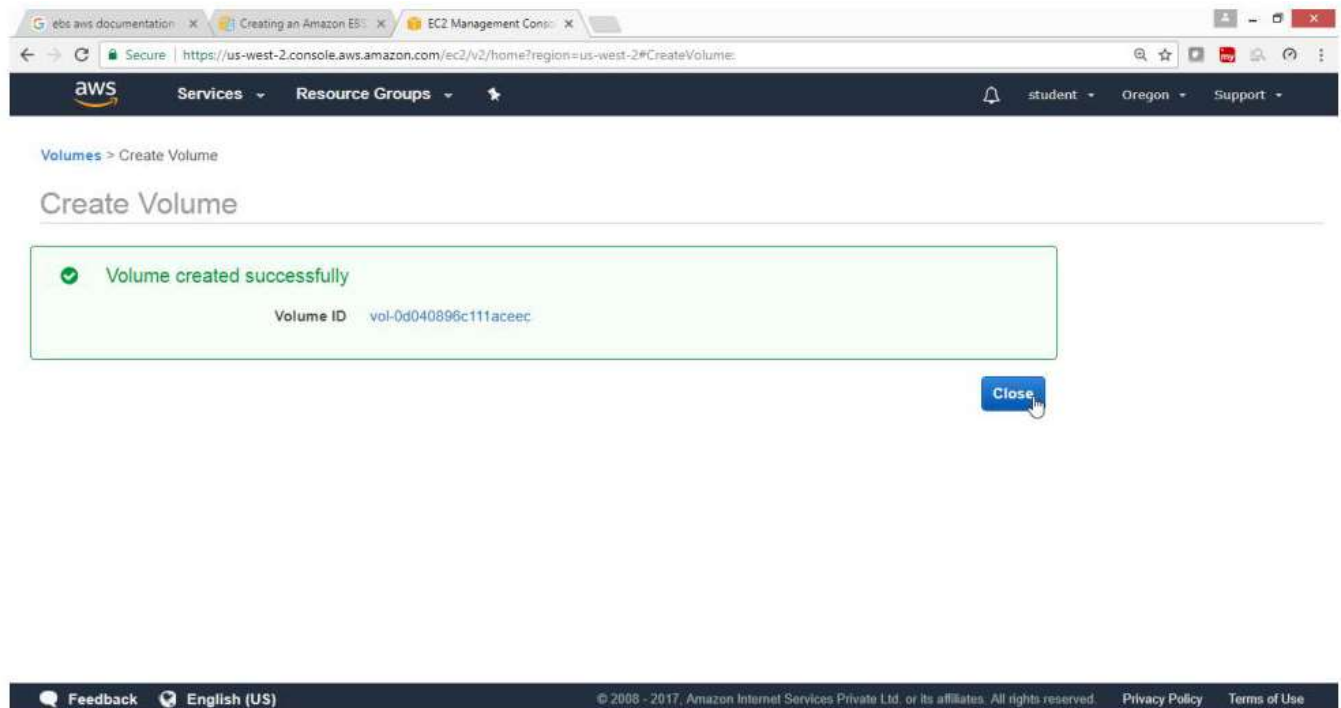
- Volume Type:** A dropdown menu set to 'General Purpose SSD (GP2)'.
- Size (GiB):** A text input field containing '2', with a note '(Min: 1 GiB, Max: 16384 GiB)'.
- IOPS:** A text input field containing '300 / 3000', with a note '(Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS)'.
- Availability Zone:** A dropdown menu set to 'us-west-2a'.
- Throughput (MB/s):** A text input field containing 'Not applicable'.
- Snapshot ID:** A dropdown menu set to 'Select a snapshot', with a refresh icon and a note.
- Encryption:** A checkbox labeled 'Encrypt this volume'.
- Tags:** A checkbox labeled 'Add tags to your volume'.

At the bottom left, there is a note '* Required'. At the bottom right, there are two buttons: 'Cancel' and 'Create Volume'.

Feedback English (US) © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

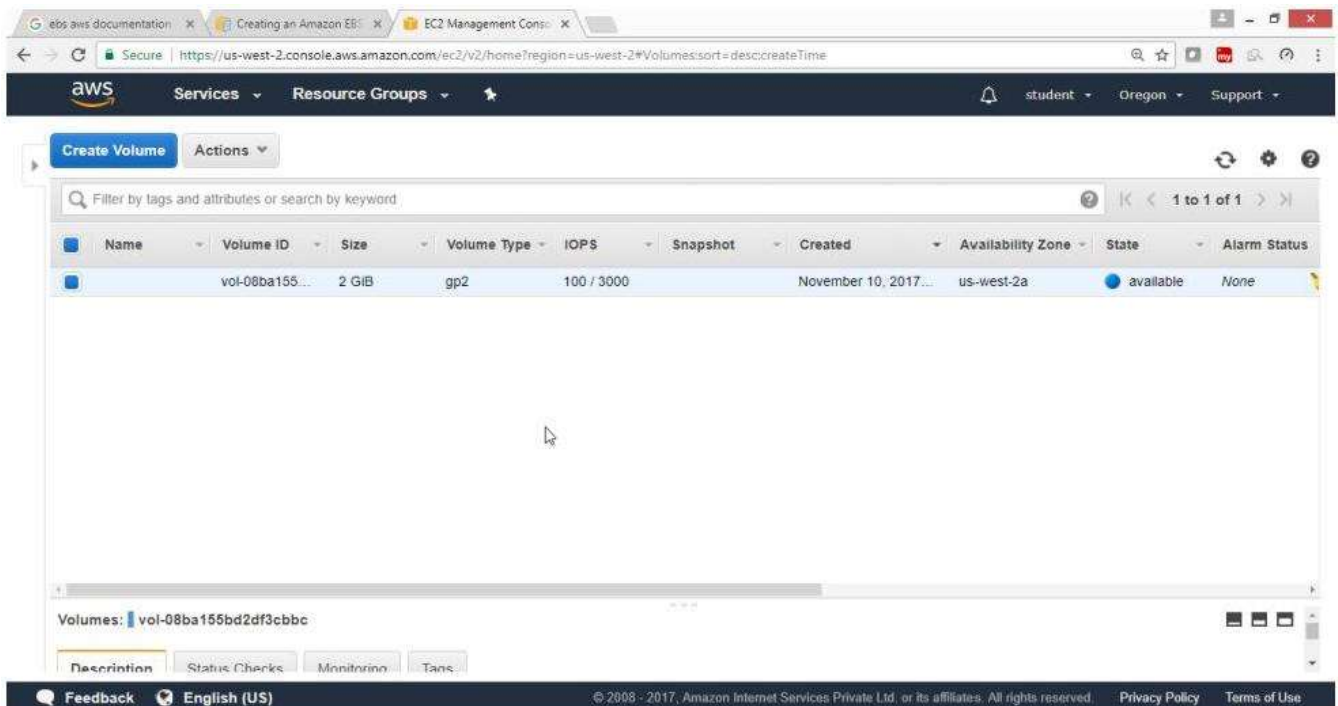
Verify Volume successfully created

Click "Close" Button

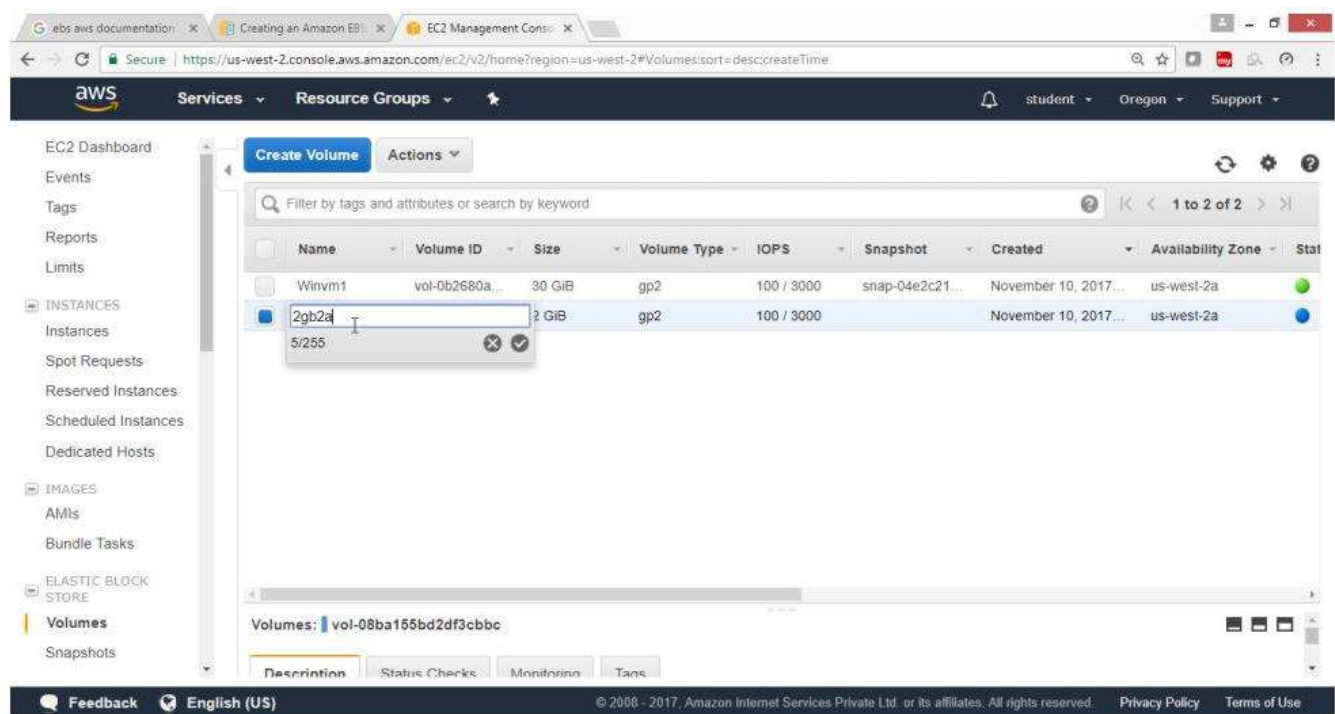


To monitoring the State of your Volumes

Select Volume check state ->available



In the Name Column give name for your volume -> 2gb2a



2) To Attaching and Detaching EBS volume in Windows Instance

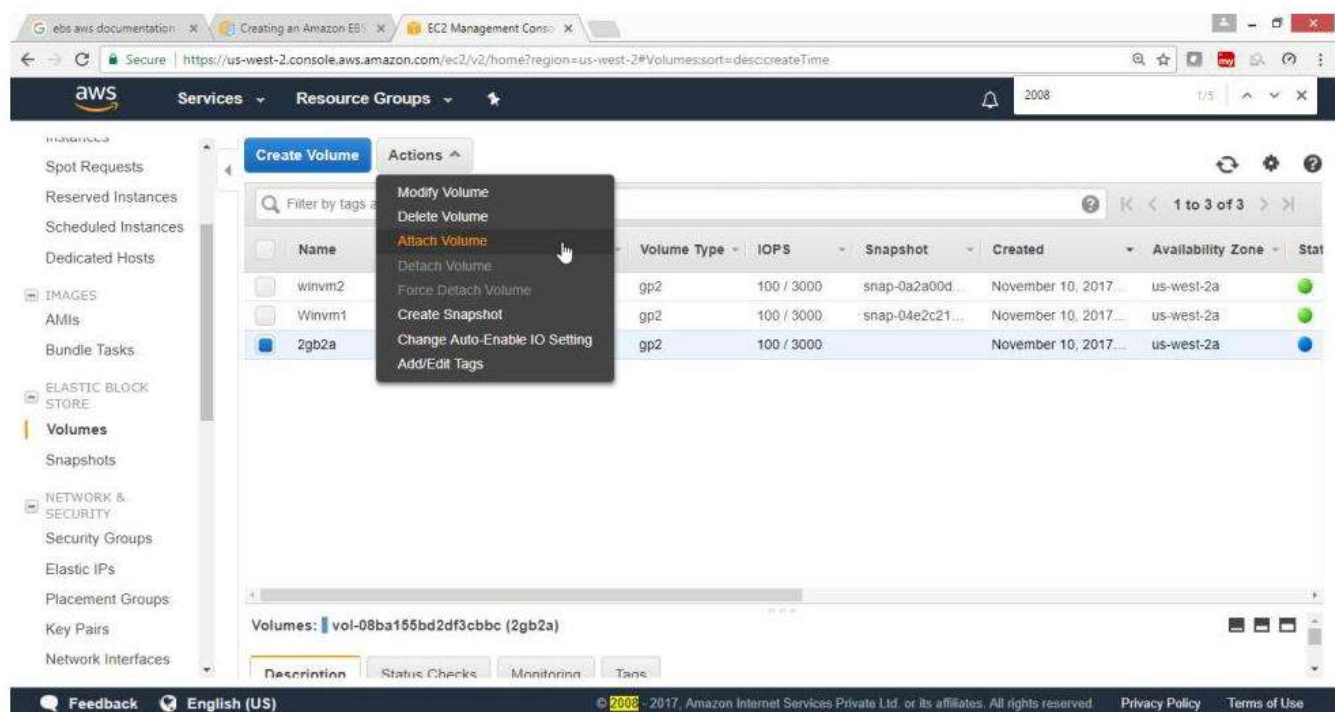
On the EC2 Dashboard Panel

Choose "Elastic Block Store" Click on Volume

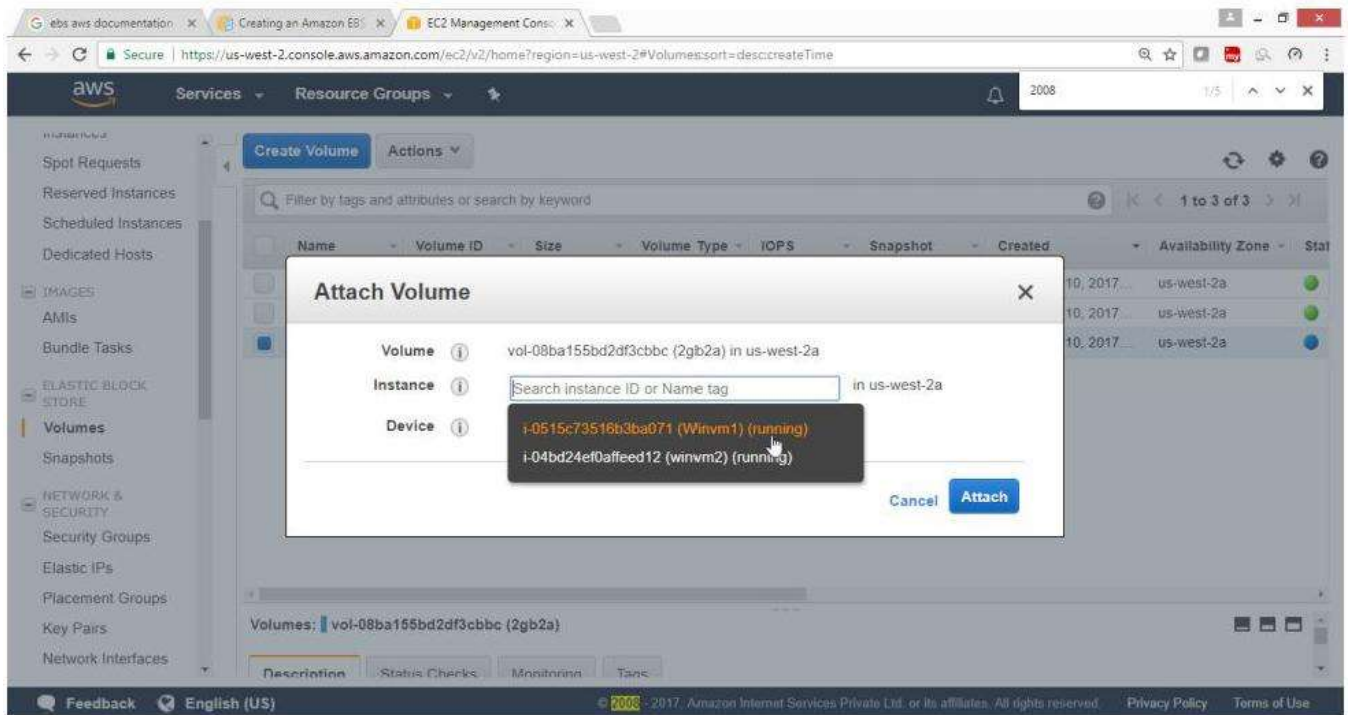
Note: The volume which you want to attach to an instance should be in same availability zone

Drop Down "Action" Button,

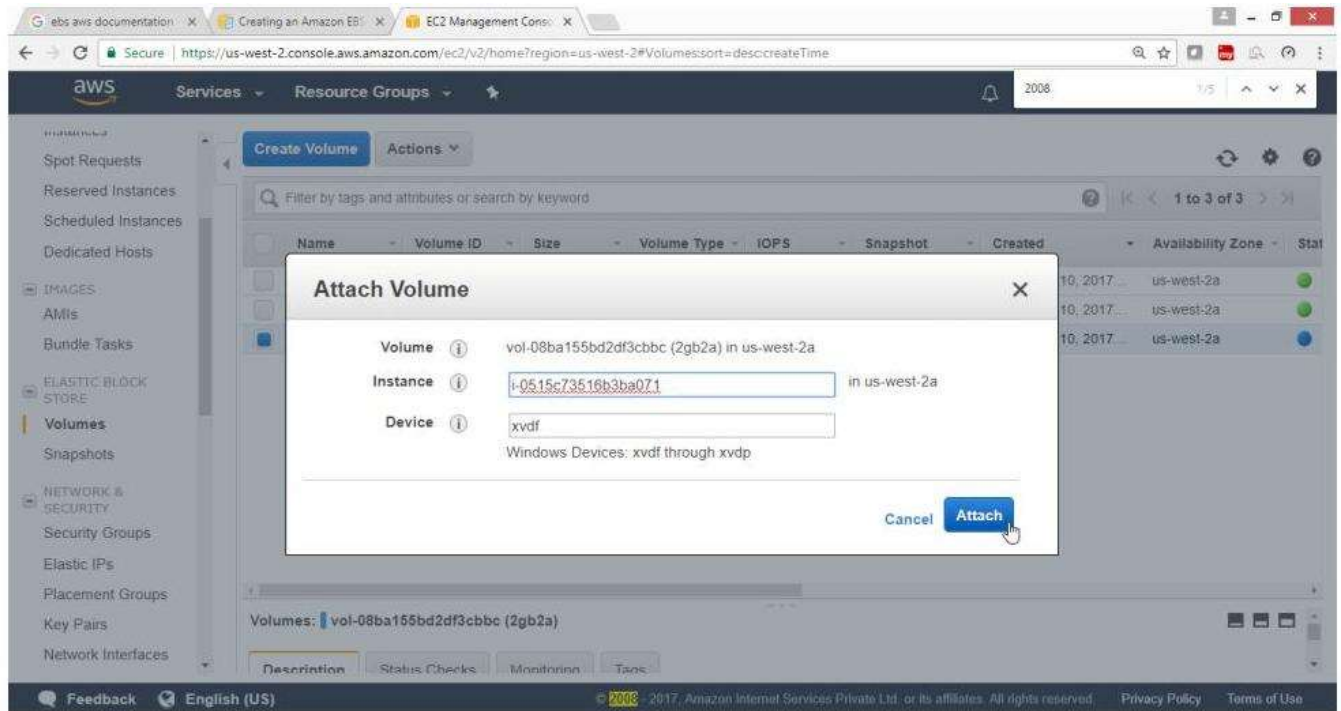
Select "Attach Volume"



Select Instance -> **Winvm1**



Click on **Attach**

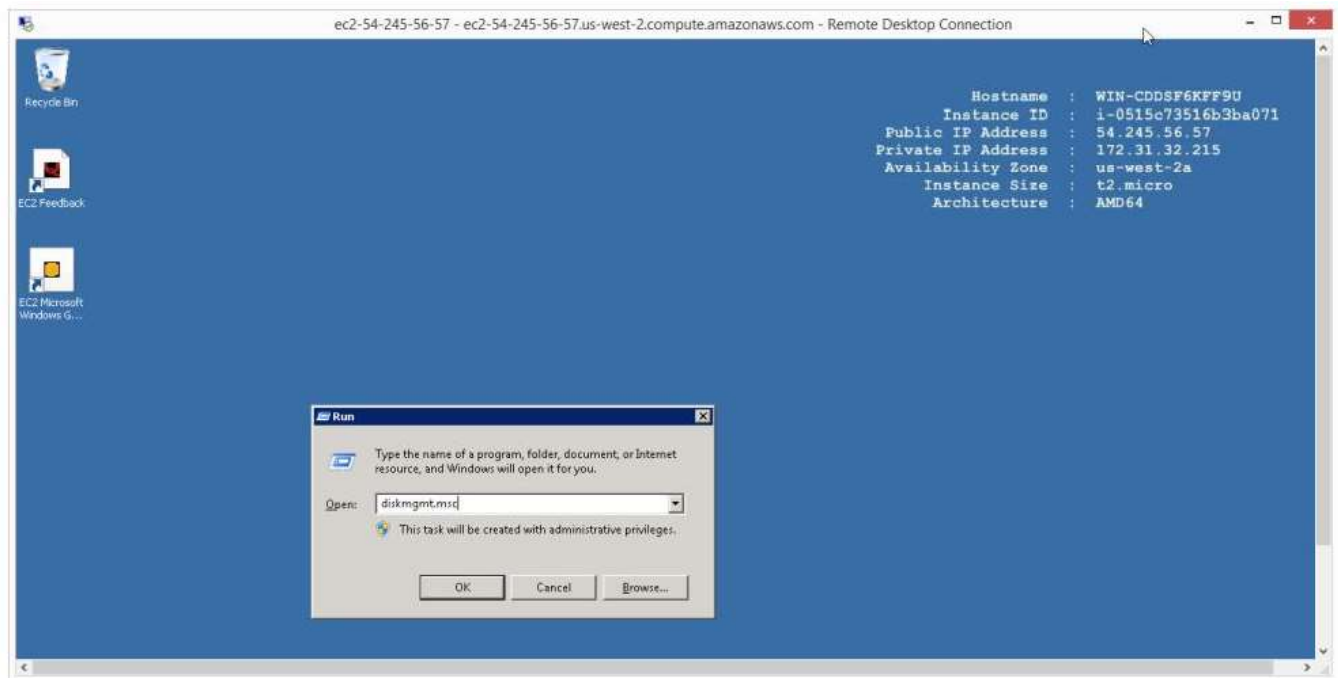


Verify the availability on new volume

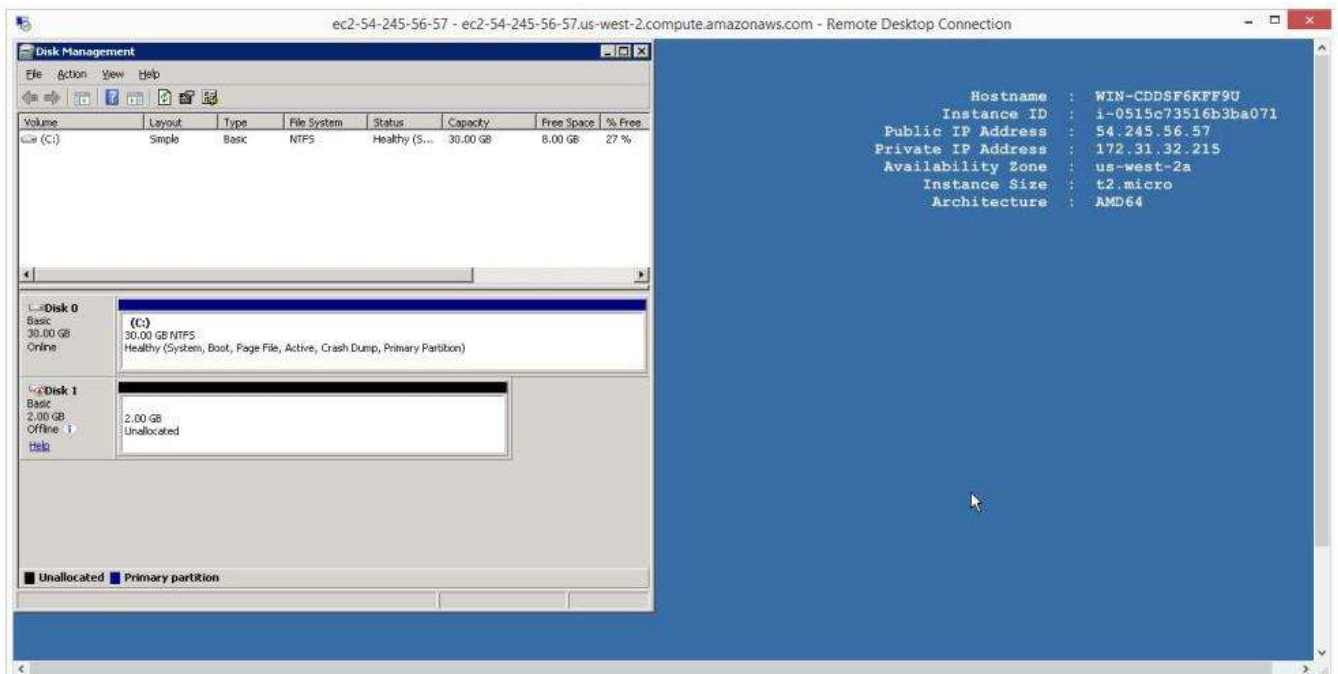
3. To check availability of new drive login to your Windows Instance

Login to Windows Instance

Run->diskmgmt.msc

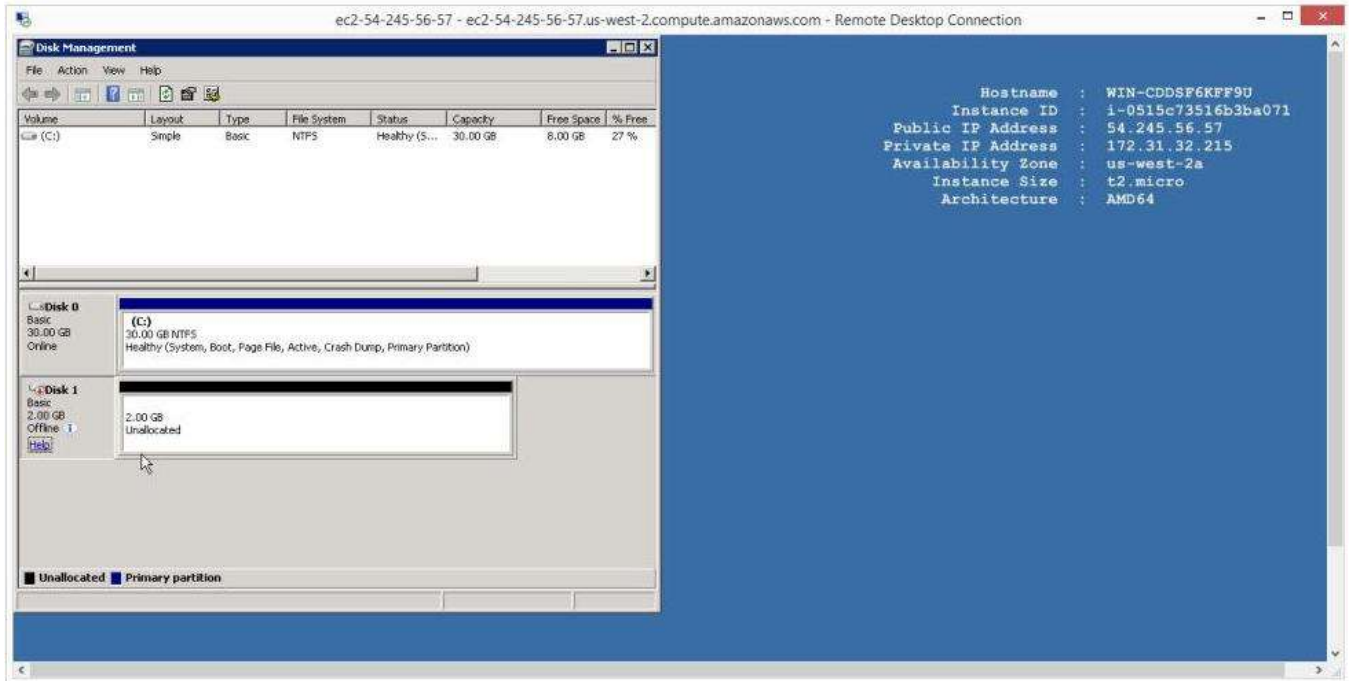


Verify that 2 GB volume available as unallocated space

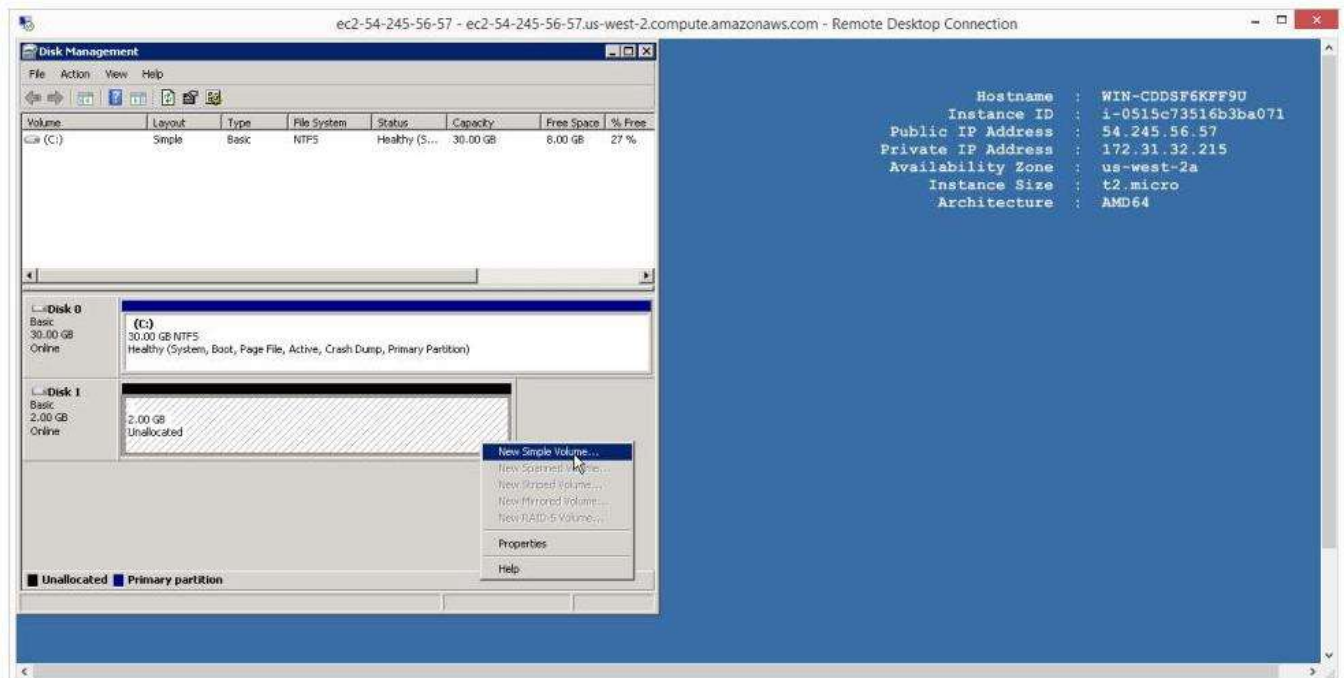


New disk is offline,

So, turn it to online by right clicking and select online

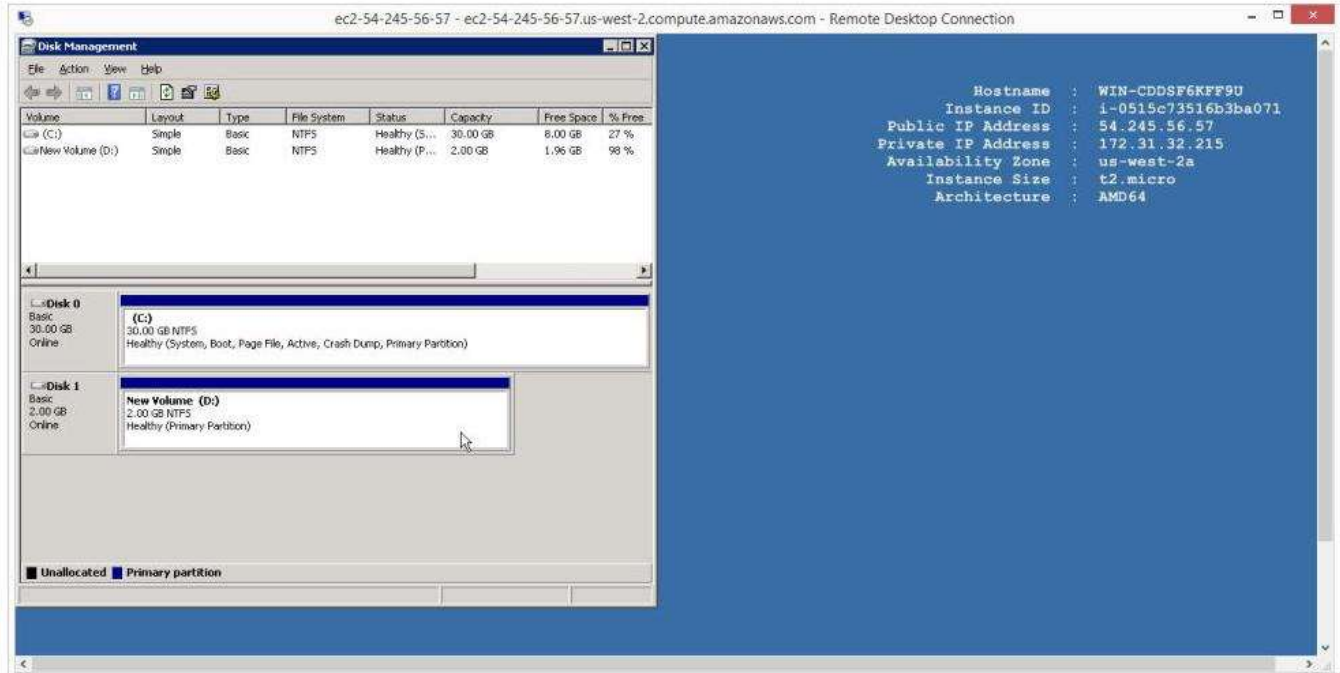


Format the unallocated disk



Verify

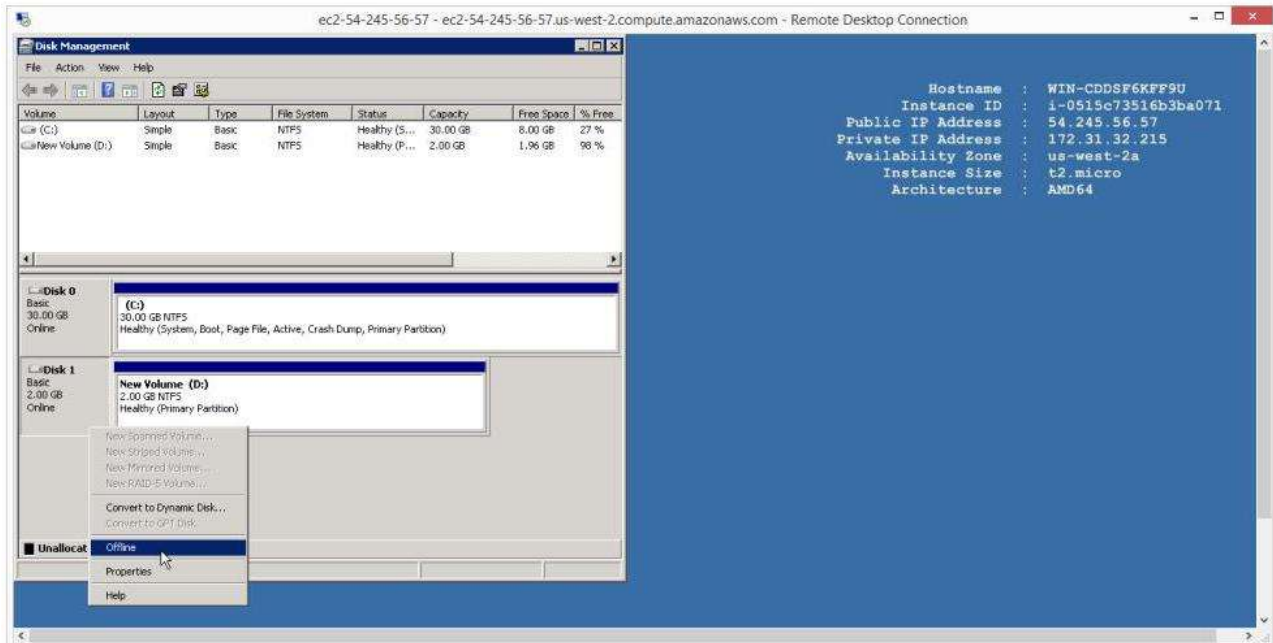
New Volume to 2 GB is available to use



4. To Detach the volume

In Windows Select Disk 1

Right click select offline



On the EC2 Dashboard panel

Choose "Elastic Block Store" click on Volumes

Select Volume to be detached under Name column

Drop Down "Action" Button

Select "Detach Volume"

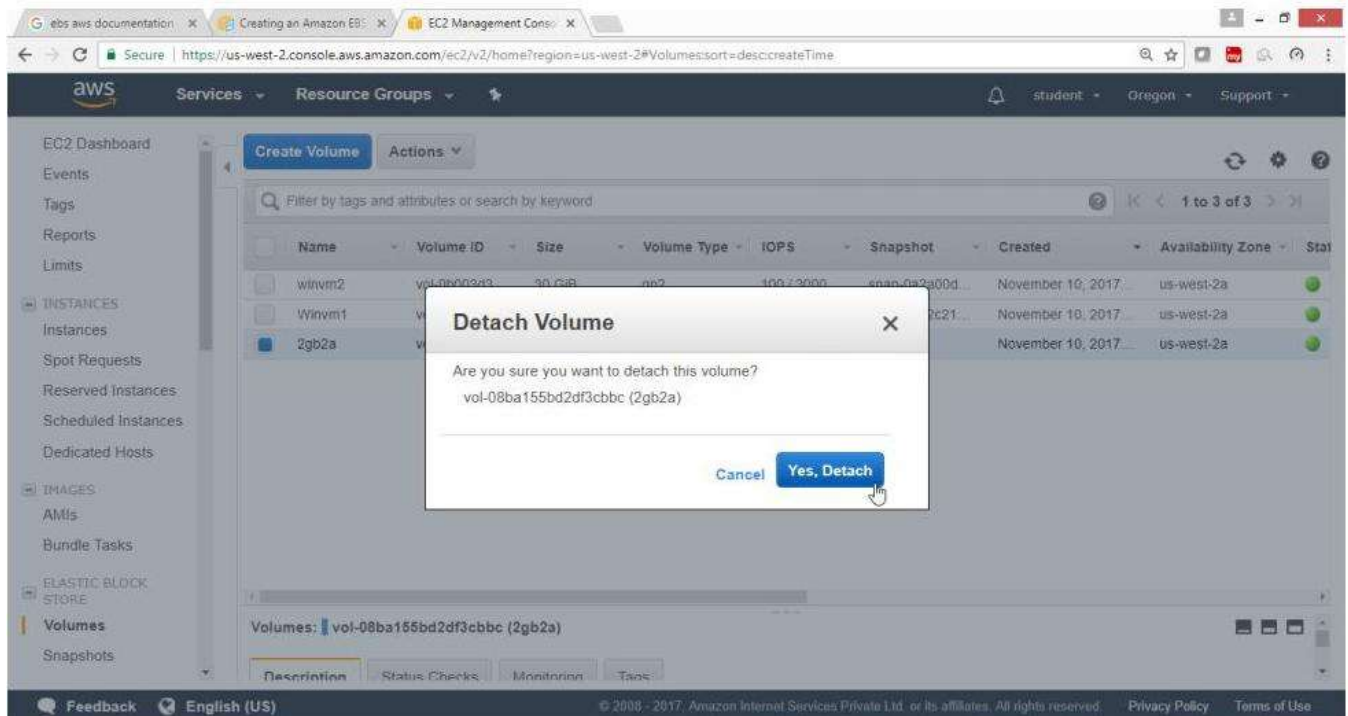
The screenshot shows the AWS Management Console interface. On the left, the navigation pane includes sections for EC2 Dashboard, INSTANCES, IMAGES, and ELASTIC BLOCK STORE. The 'Volumes' section under ELASTIC BLOCK STORE is selected. The main content area displays a table of volumes. The volume '2gb2a' is selected, and the 'Actions' dropdown menu is open, showing options like 'Detach Volume', 'Force Detach Volume', and 'Create Snapshot'. The 'Detach Volume' option is highlighted.

Name	Volume Type	IOPS	Snapshot	Created	Availability Zone	Status
winvm2	gp2	100 / 3000	snap-0a2a00d...	November 10, 2017...	us-west-2a	Available
Winvm1	gp2	100 / 3000	snap-04e2c21...	November 10, 2017...	us-west-2a	Available
2gb2a	gp2	100 / 3000		November 10, 2017...	us-west-2a	Available

Volumes: vol-08ba155bd2df3cbbc (2gb2a)

Feedback English (US) © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

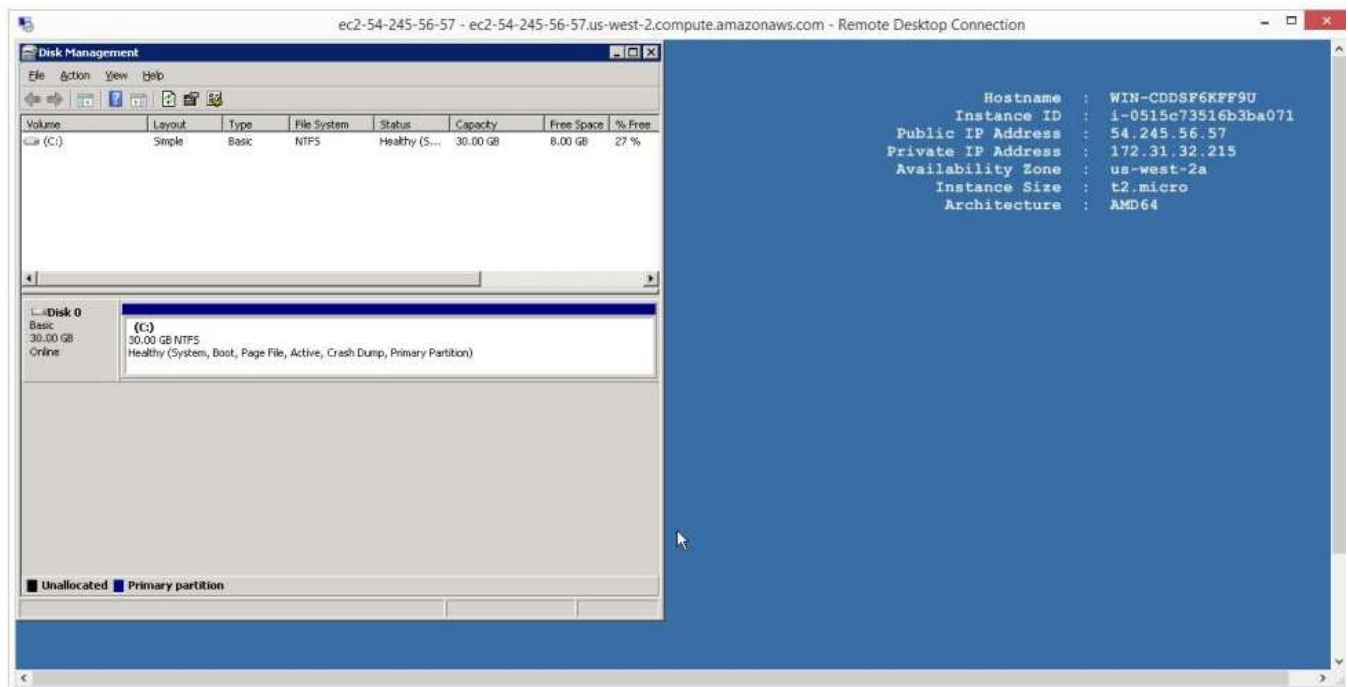
Click on "Yes, Detach" Button



Verification

Login to windows instance

Check that D: drive is removed

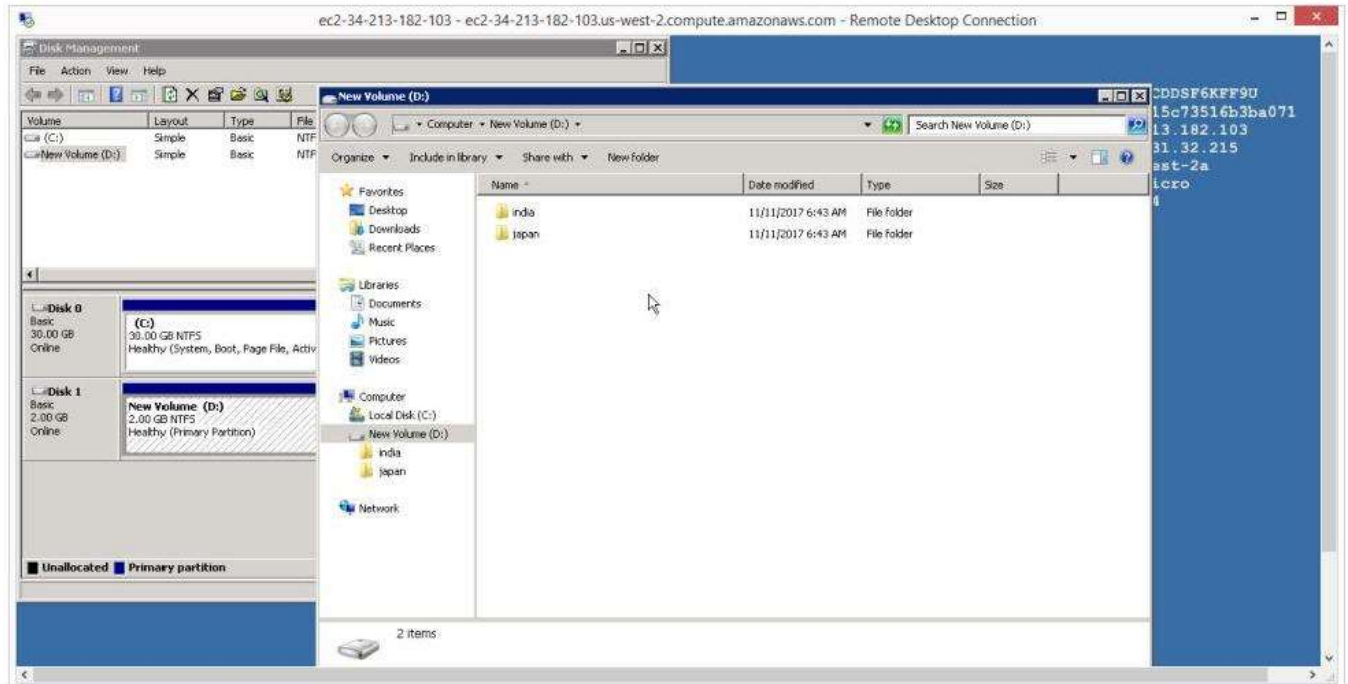


5. To Create Snapshot and Restore EBS Volumes

To create a snapshot

In the current D drive two folders are available

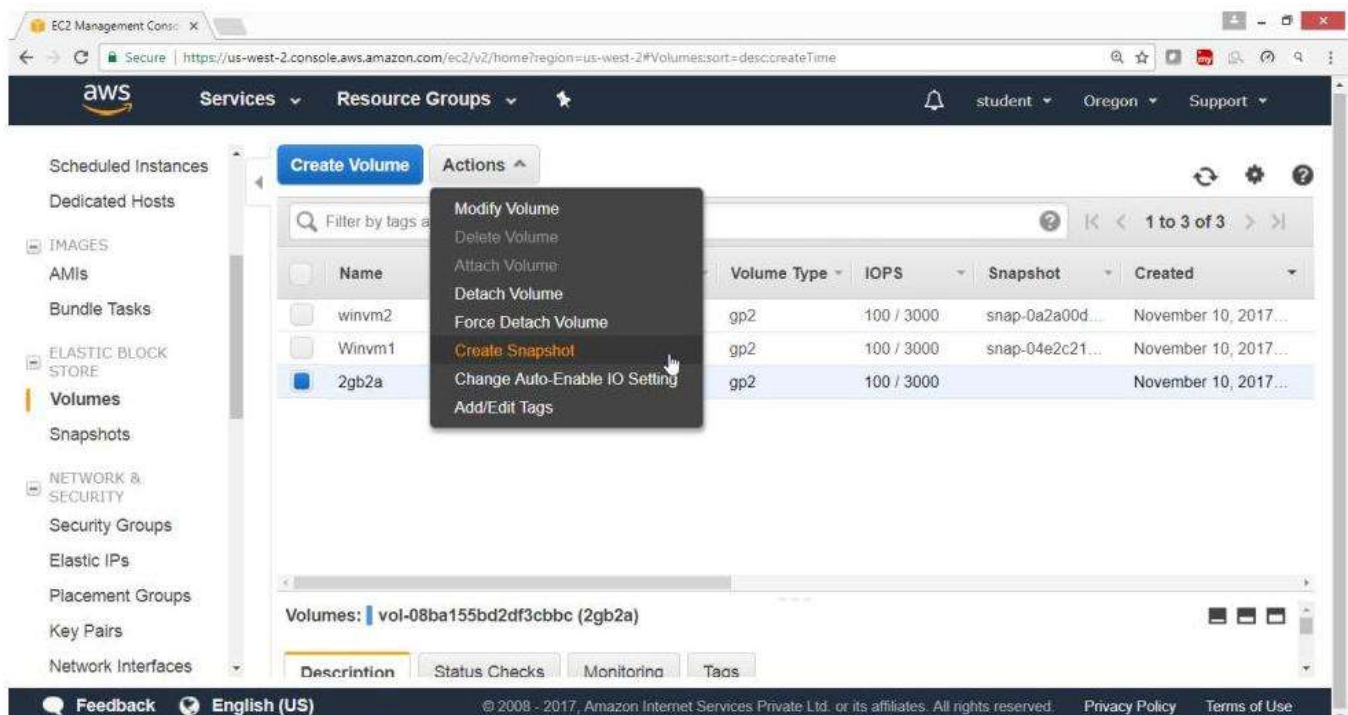
No create a snapshot of this volume



On the "EC2 Dashboard" Panel

Click on "Elastic Block Store" choose Volumes

Drop Down "Action" Button select Create Snapshot



Provide snapshot details

Click "Create" button

Create Snapshot

Volume vol-08ba155bd2df3cbbc (2gb2a)

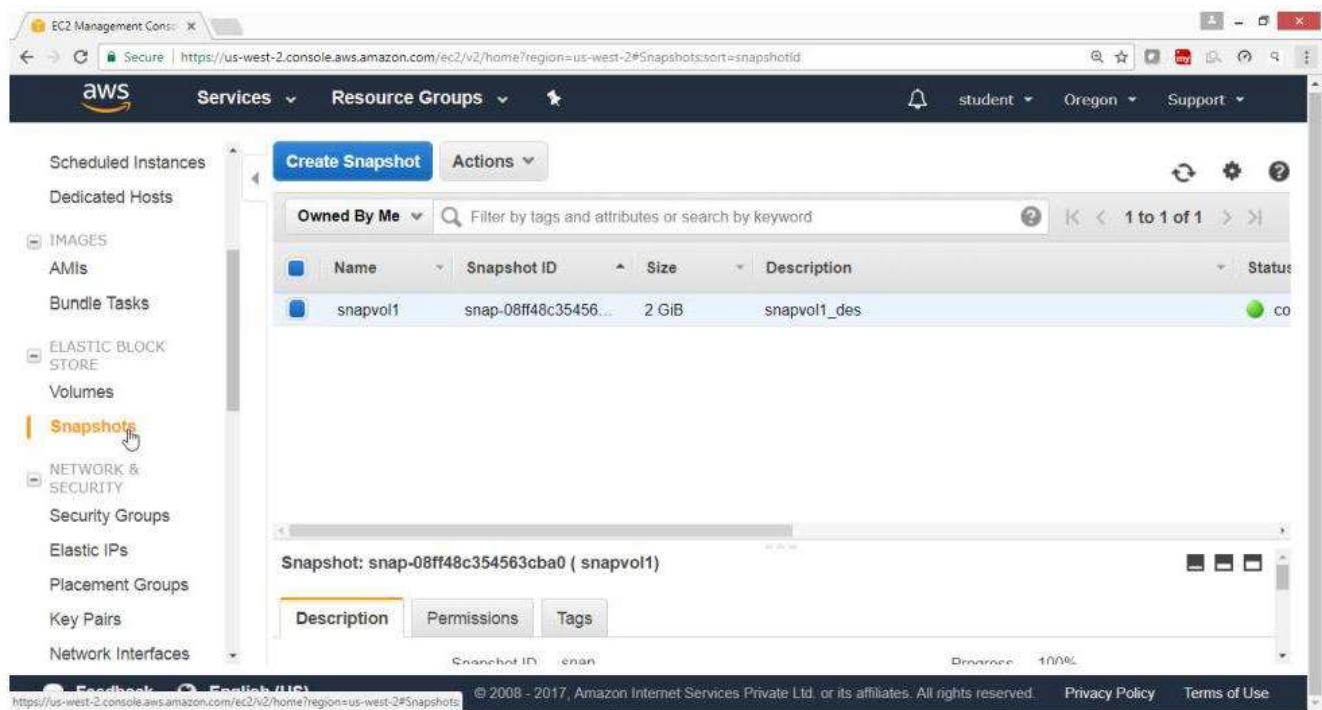
Name

Description

Encrypted No

[Cancel](#) [Create](#)

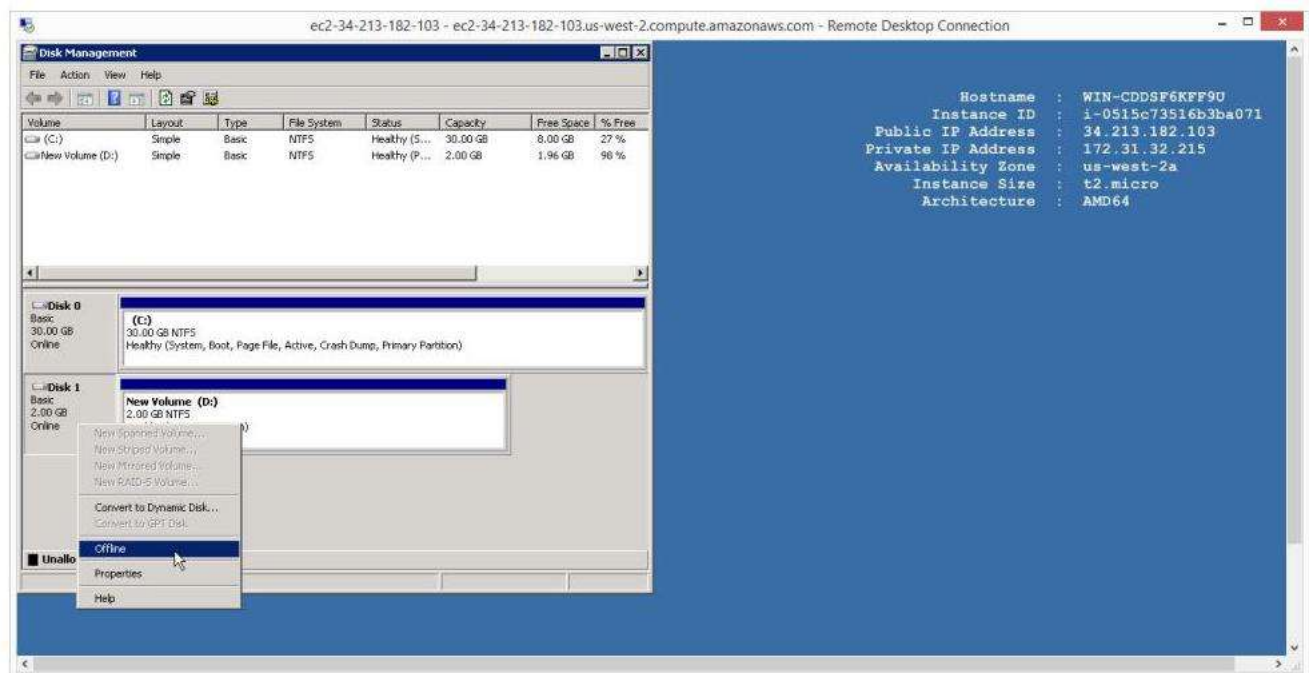
Verify that snapshot is created



6) To Delete the Volume

First select the disk1 from Disk Management

Right click select offline



On the EC2 Dashboard panel

Expand "Elastic Block Store", choose volumes

Select volume to be detached under the Name column

Drop Down "Action" Button, Select "Delete Volume"

The screenshot shows the AWS Management Console interface for the EC2 Volumes page. The left sidebar contains navigation links for various AWS services, including S3, IAM, CloudFormation, and Elastic Block Store. The main content area displays a table of volumes. The 'Actions' dropdown menu is open, showing options like 'Modify Volume', 'Delete Volume', 'Attach Volume', 'Detach Volume', 'Force Detach Volume', 'Create Snapshot', 'Change Auto-Enable IO Setting', and 'Add/Edit Tags'. The 'Delete Volume' option is highlighted. Below the table, there is a section for the selected volume 'vol-08ba155bd2df3cbbc (2gb2a)' with tabs for 'Description', 'Status Checks', 'Monitoring', and 'Tags'.

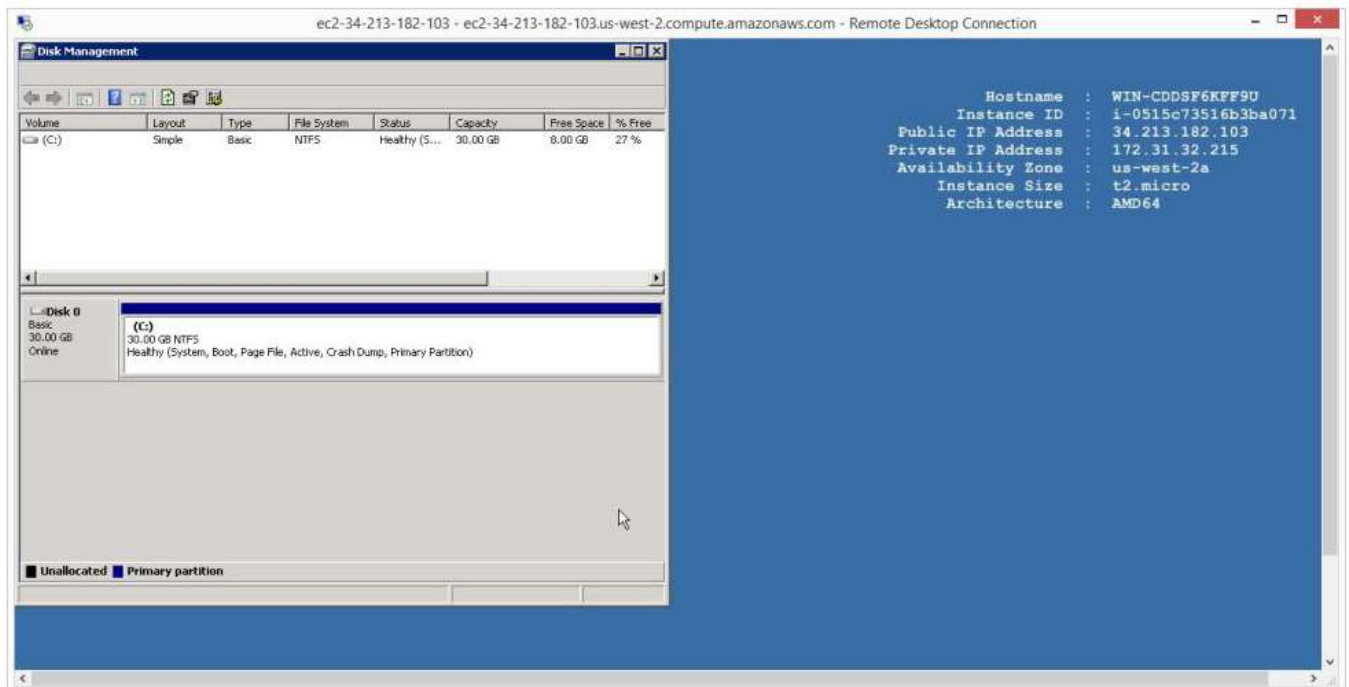
Name	Volume Type	IOPS	Snapshot	Created
winvm2	gp2	100 / 3000	snap-0a2a00d...	November 10, 2017...
Winvm1	gp2	100 / 3000	snap-04e2c21...	November 10, 2017...
2gb2a	gp2	100 / 3000		November 10, 2017...

Volumes: vol-08ba155bd2df3cbbc (2gb2a)

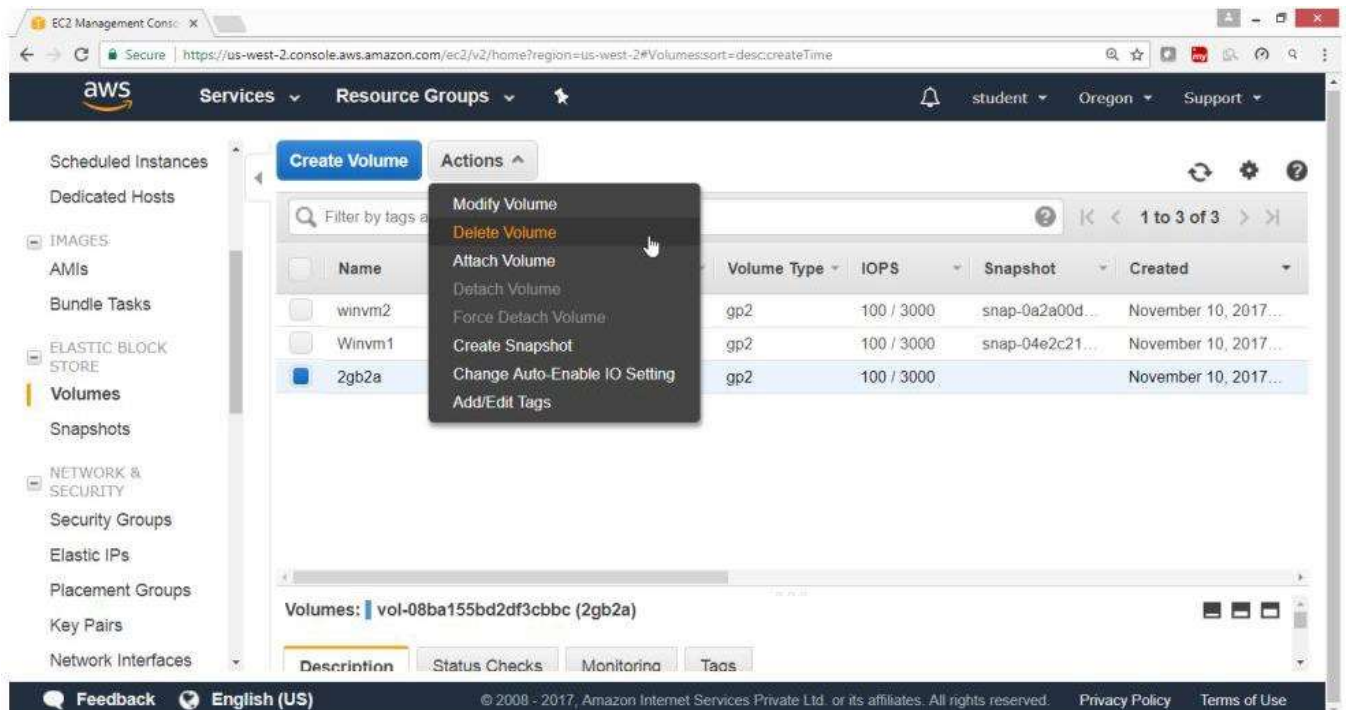
Feedback English (US) © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Verify from windows instance open disk management tool

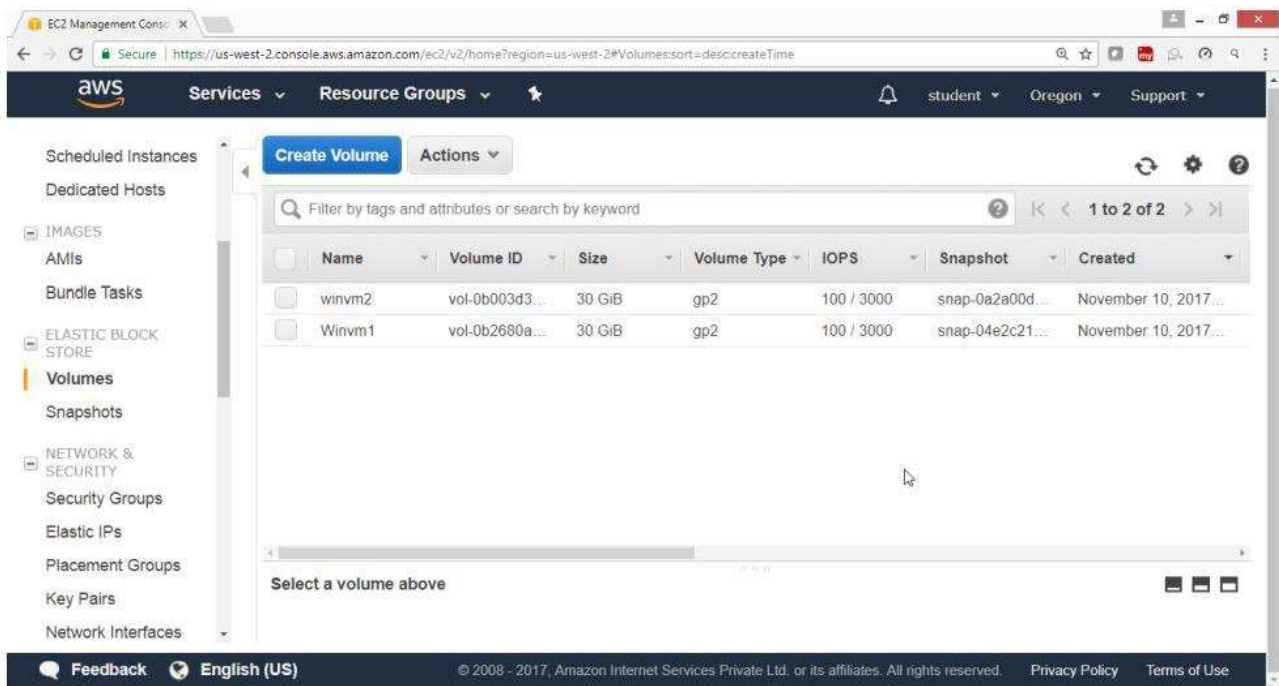
Now D drive is detached



Now delete the volume



Verify volume is deleted



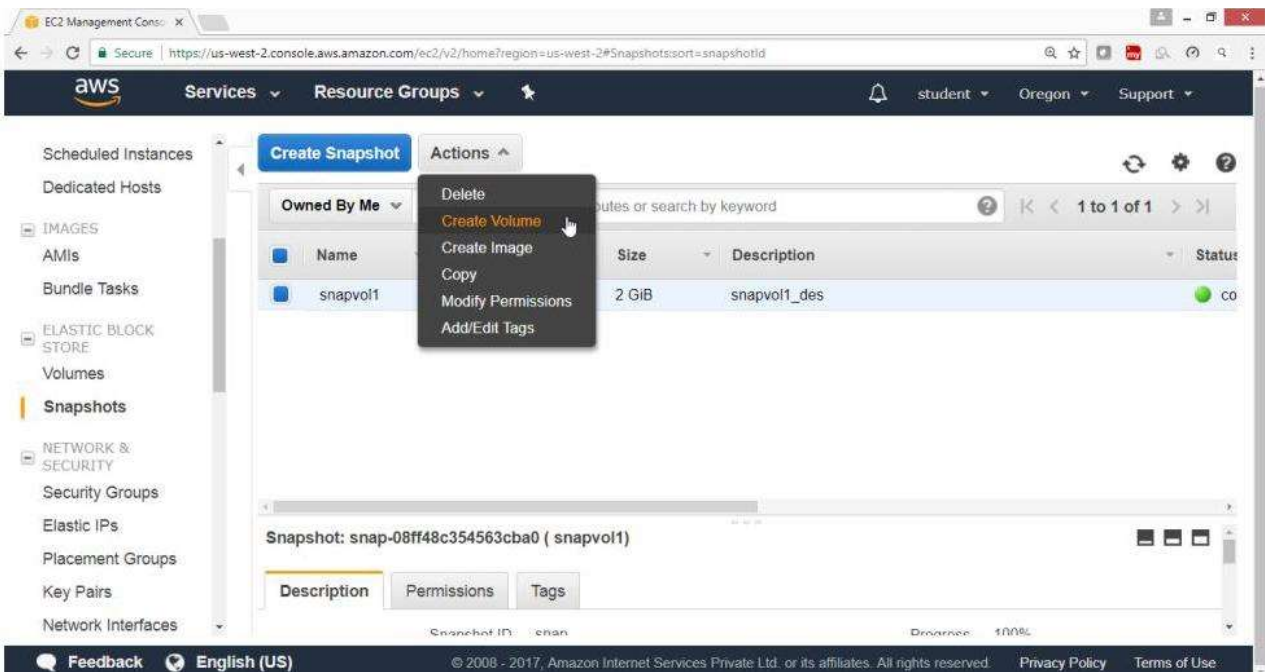
7.To Restore the volume

From the console EC2 Dashboard

Expand "Elastic Block Store", choose Snapshots

Select the snapshot

Drop Down "Action" button, Select Create Volume



Accept the defaults values in wizard

Note: Check the right availability zone

aws Services Resource Groups

student Oregon Support

Snapshot ID snap-08ff48c354563cba0 (snapvol1)

Volume Type General Purpose SSD (GP2) ⓘ

Size (GiB) 2 (Min: 1 GiB, Max: 16384 GiB) ⓘ

IOPS 100 / 3000 (Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS) ⓘ

Availability Zone* us-west-2a ⓘ

Throughput (MB/s) Not applicable ⓘ

Encryption Not Encrypted

Tags ☐ Add tags to your volume

* Required

Cancel Create Volume

Feedback English (US) © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Verify the volume is created

EC2 Management Console

https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#Volumes:sort=descCreateTime

aws Services Resource Groups

student Oregon Support

Scheduled Instances

Dedicated Hosts

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

NETWORK & SECURITY

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

Create Volume Actions

Filter by tags and attributes or search by keyword ⓘ

Name	Volume ID	Size	Volume Type	IOPS	Snapshot	Created
winvm2	vol-0cd6da3c...	2 GiB	gp2	100 / 3000	snap-08ff48c3...	November 11, 2017 ...
winvm1	vol-0b003d3...	30 GiB	gp2	100 / 3000	snap-0a2a00d...	November 10, 2017...
Winvm1	vol-0b2680a...	30 GiB	gp2	100 / 3000	snap-04e2c21...	November 10, 2017...

Volumes: vol-0cd6da3c73f3a3881

Description Status Checks Monitoring Tags

Feedback English (US) © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

8) To expanding the size of EBS volume

To expand EBS volume first take snapshot, now select the snapshot

On the EC2 Dashboard panel

Expand "Elastic Block Store", Choose Snapshots

Drop Down "Action" Button

Select "Create Volume"

The screenshot shows the AWS Management Console interface. On the left, the navigation pane lists various services, with 'Snapshots' highlighted under 'ELASTIC BLOCK STORE'. The main content area displays a table of snapshots. A dropdown menu is open, showing options: 'Delete', 'Create Volume' (highlighted), 'Create Image', 'Copy', 'Modify Permissions', and 'Add/Edit Tags'. Below the table, the details for a specific snapshot are shown, including its ID and a progress bar at 100%.

EC2 Management Console

Services Resource Groups

student Oregon Support

Scheduled Instances
Dedicated Hosts

IMAGES
AMIs
Bundle Tasks

ELASTIC BLOCK STORE
Volumes
Snapshots

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

Create Snapshot Actions

Owned By Me

Routes or search by keyword

Name	Size	Description	Status
snapvol1	2 GiB	snapvol1_des	Completed

Snapshot: snap-08ff48c354563cba0 (snapvol1)

Description Permissions Tags

Snapshot ID: snap-08ff48c354563cba0 Progress: 100%

Feedback English (US) © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Give the required size ->4GB

Check the right availability zone

Click "Create Volume" Button

The screenshot shows the 'Create Volume' form in the AWS console. The form is for creating a new EBS volume from a snapshot. The 'Snapshot ID' is 'snap-08ff48c354563cba0 (snapvol1)'. The 'Volume Type' is 'General Purpose SSD (GP2)'. The 'Size (GiB)' is set to '4', with a note '(Min: 1 GiB, Max: 16384 GiB)'. The 'IOPS' are '100 / 3000', with a note '(Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS)'. The 'Availability Zone' is 'us-west-2a'. The 'Throughput (MB/s)' is 'Not applicable'. The 'Encryption' is 'Not Encrypted'. There is a 'Tags' section with a checkbox 'Add tags to your volume'. At the bottom right, there are 'Cancel' and 'Create Volume' buttons. The footer includes 'Feedback', 'English (US)', and copyright information.

Snapshot ID: snap-08ff48c354563cba0 (snapvol1)

Volume Type: General Purpose SSD (GP2)

Size (GiB): 4 (Min: 1 GiB, Max: 16384 GiB)

IOPS: 100 / 3000 (Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS)

Availability Zone*: us-west-2a

Throughput (MB/s): Not applicable

Encryption: Not Encrypted

Tags: ☐ Add tags to your volume

* Required

Cancel Create Volume

Verify that 4 GB is created

The screenshot shows the 'Volumes' page in the AWS console. The left sidebar contains navigation links for 'Scheduled Instances', 'Dedicated Hosts', 'IMAGES', 'AMIs', 'Bundle Tasks', 'ELASTIC BLOCK STORE', 'Volumes', 'Snapshots', 'NETWORK & SECURITY', 'Security Groups', 'Elastic IPs', 'Placement Groups', 'Key Pairs', and 'Network Interfaces'. The main area shows a table of volumes. The first volume is highlighted with a blue selection box. The table has columns: Name, Volume ID, Size, Volume Type, IOPS, Snapshot, and Created. Below the table, there is a search bar and a list of volumes: 'vol-034d7007ffcef5949', 'vol-0cd6da3c...', 'winvm2', and 'Winvm1'. At the bottom, there are tabs for 'Description', 'Status Checks', 'Monitoring', and 'Tags'. The footer includes 'Feedback', 'English (US)', and copyright information.

Name	Volume ID	Size	Volume Type	IOPS	Snapshot	Created
vol-034d700...	vol-034d700...	4 GiB	gp2	100 / 3000	snap-08ff48c3...	November 11, 2017 ...
vol-0cd6da3c...	vol-0cd6da3c...	2 GiB	gp2	100 / 3000	snap-08ff48c3...	November 11, 2017 ...
winvm2	vol-0b003d3...	30 GiB	gp2	100 / 3000	snap-0a2a00d...	November 10, 2017 ...
Winvm1	vol-0b2680a...	30 GiB	gp2	100 / 3000	snap-04e2c21...	November 10, 2017 ...

Volumes: vol-034d7007ffcef5949

Description Status Checks Monitoring Tags

Now attach this expanded volume to your instance

EC2 Management Console

Services Resource Groups

Scheduled Instances
Dedicated Hosts

IMAGES
AMIs
Bundle Tasks

ELASTIC BLOCK STORE
Volumes
Snapshots

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

Create Volume Actions

- Modify Volume
- Delete Volume
- Attach Volume
- Detach Volume
- Force Detach Volume
- Create Snapshot
- Change Auto-Enable IO Setting
- Add/Edit Tags

Name	Volume Type	IOPS	Snapshot	Created
	gp2	100 / 3000	snap-08ff48c3...	November 11, 2017 ...
winvm2	gp2	100 / 3000	snap-08ff48c3...	November 11, 2017 ...
winvm1	gp2	100 / 3000	snap-0a2a00d...	November 10, 2017 ...
Winvm1	gp2	100 / 3000	snap-04e2c21...	November 10, 2017 ...

Volumes: vol-034d7007ffcef5949

Description Status Checks Monitoring Tags

Feedback English (US) © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Select Instance

Attach Volume

Volume ⓘ vol-034d7007ffcef5949 in us-west-2a

Instance ⓘ in us-west-2a

Device ⓘ

- i-0515c73516b3ba071 (Winvm1) (running)
- i-04bd24ef0affeed12 (winvm2) (running)

Cancel Attach

Click Attach Button

Attach Volume

Volume

vol-034d7007ffcef5949 in us-west-2a

Instance

i-0515c73516b3ba071

in us-west-2a

Device

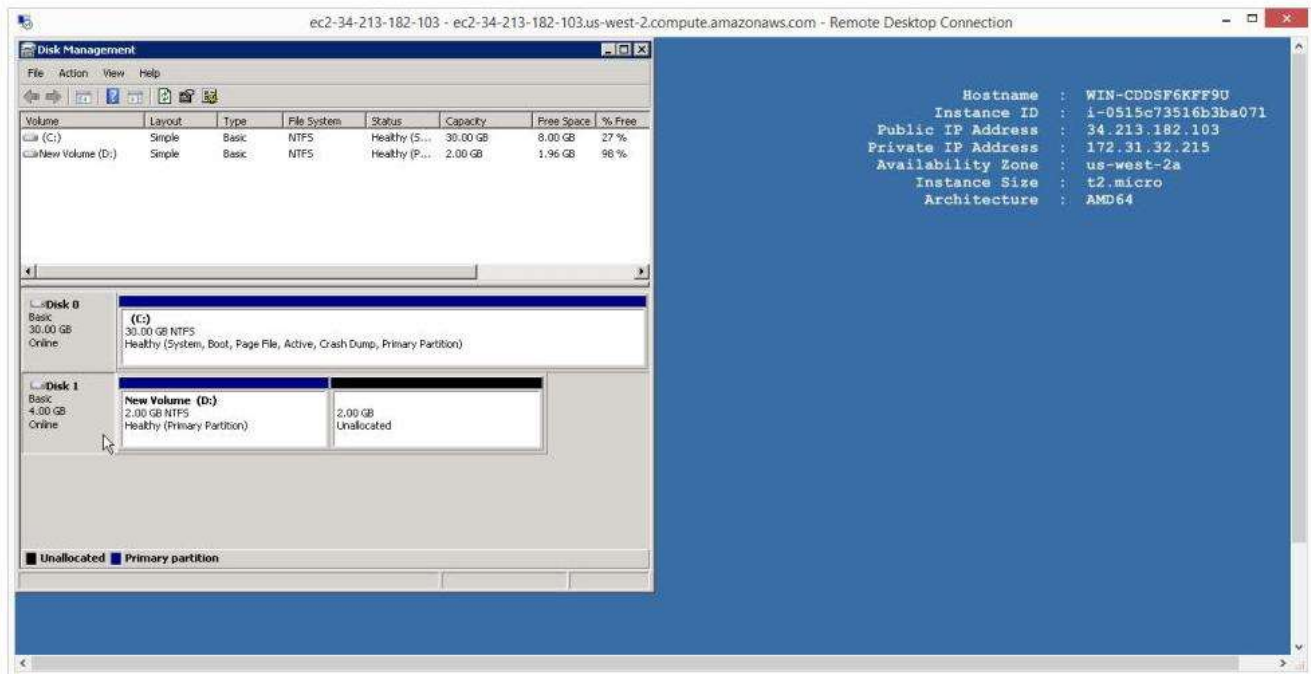
xvdf

Windows Devices: xvdf through xvdp

Cancel

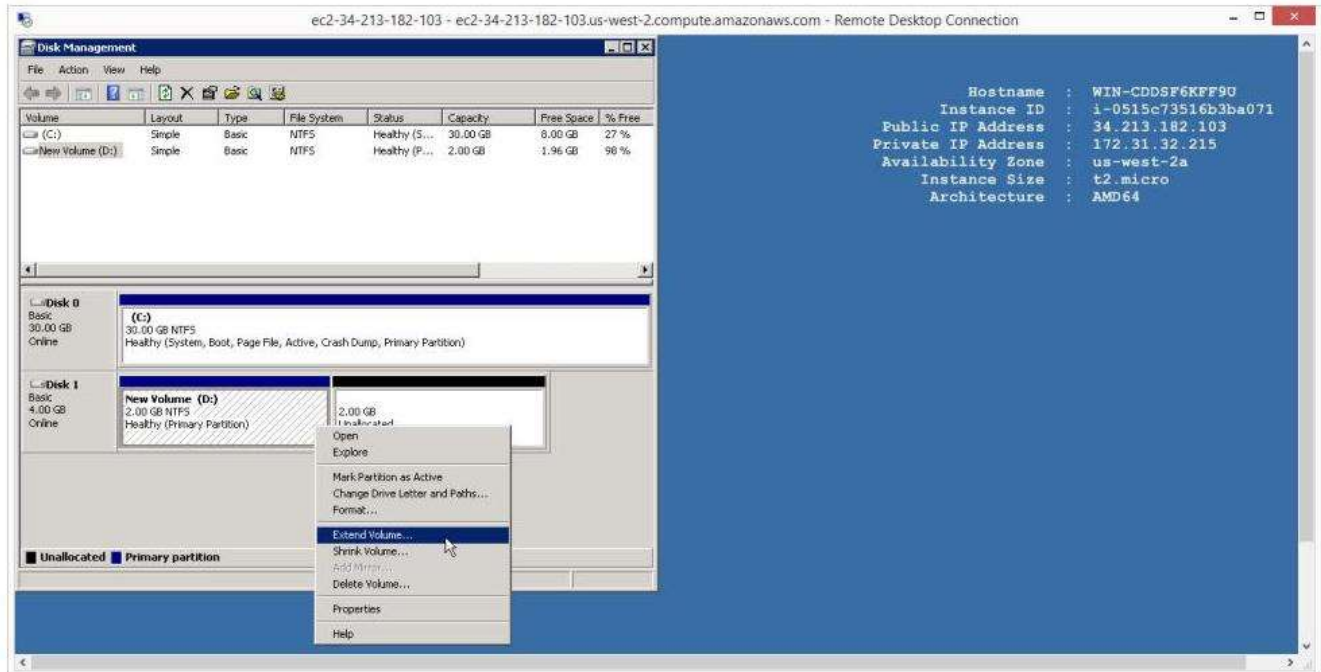
Attach

Verify 4 GB drive is available

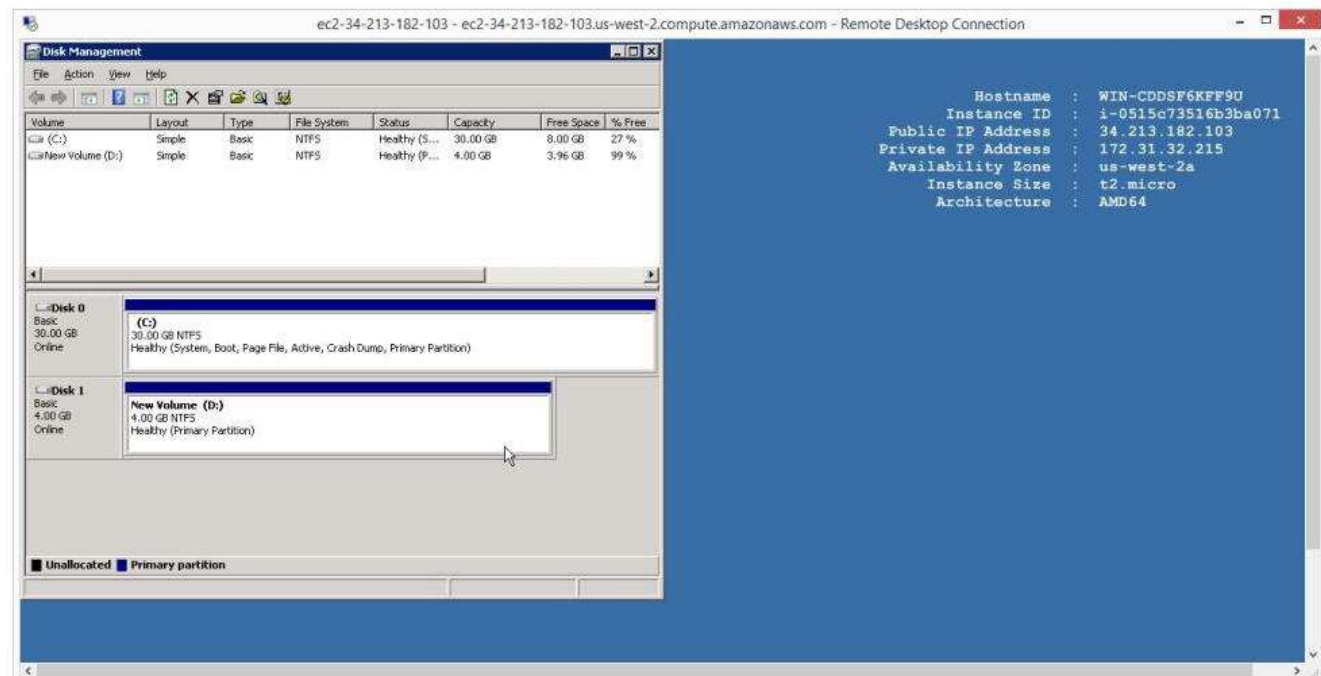


Now with respect to Window Operating System

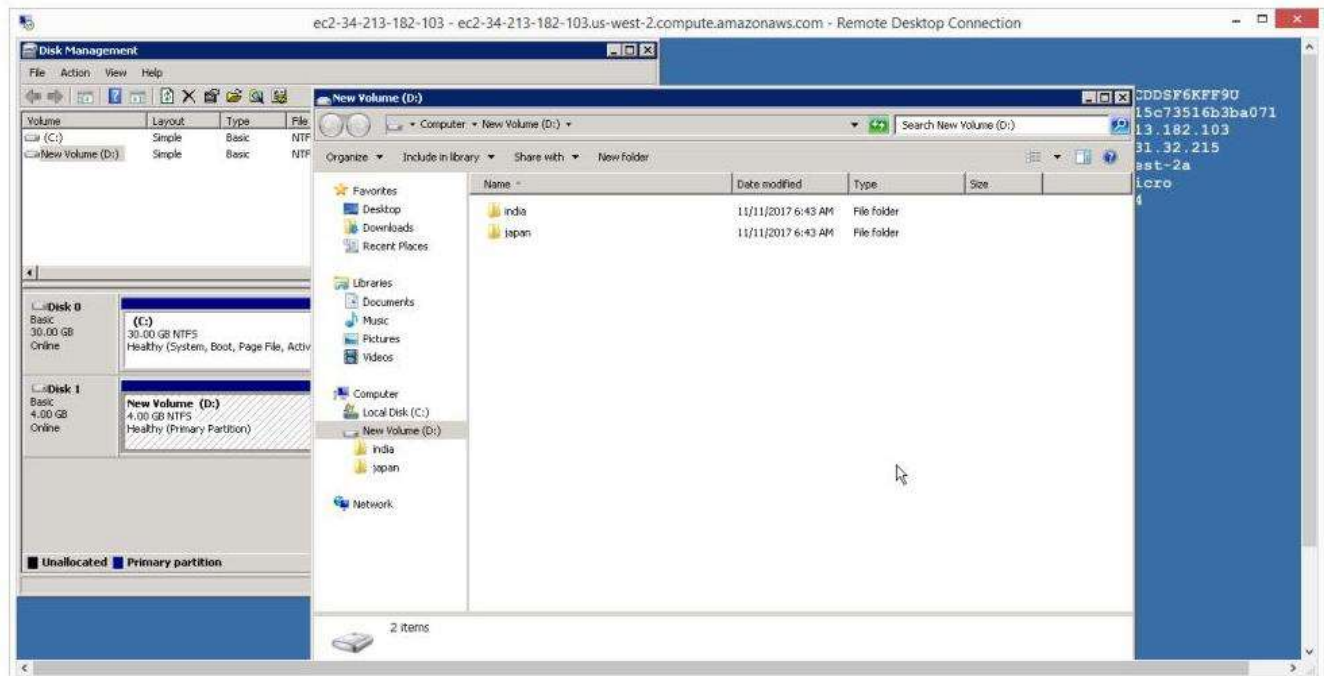
Right Click on D drive extent your volume to your desired size



Verify that 4 GB volume available

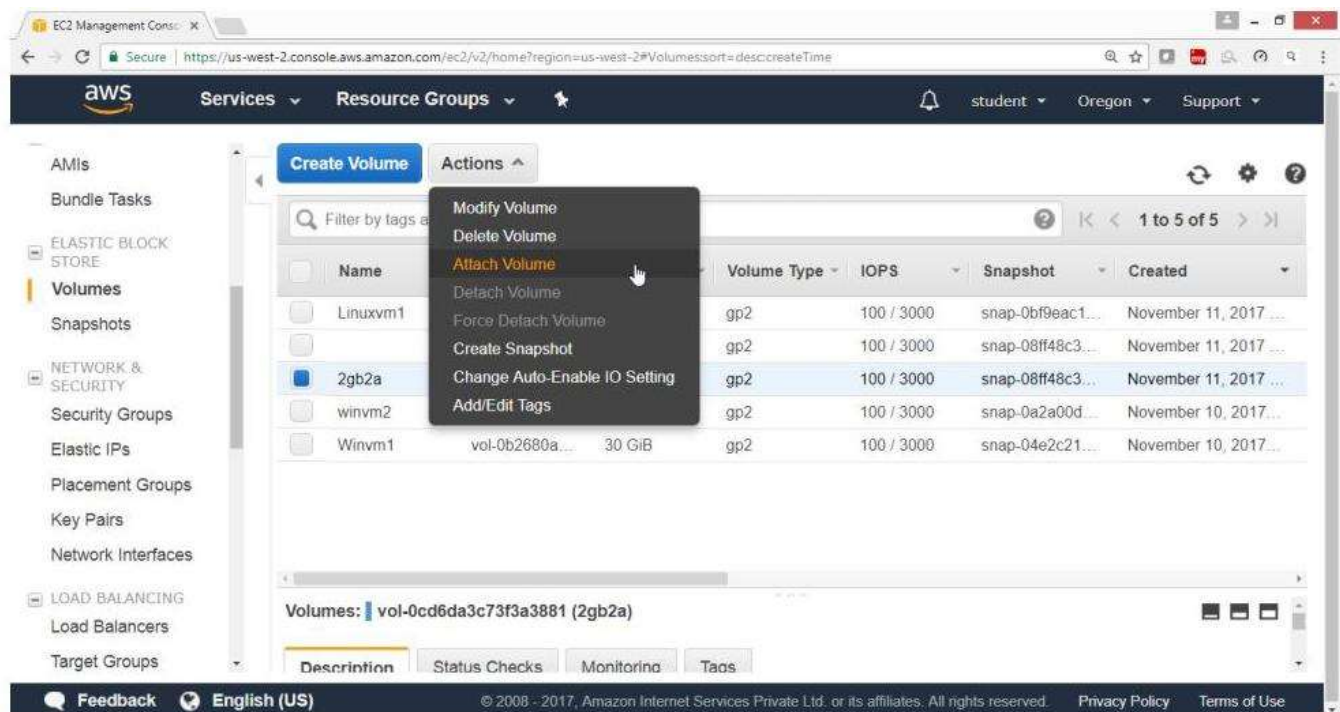


Verify that D drive contains two folders that was there in 2 GB drive earlier

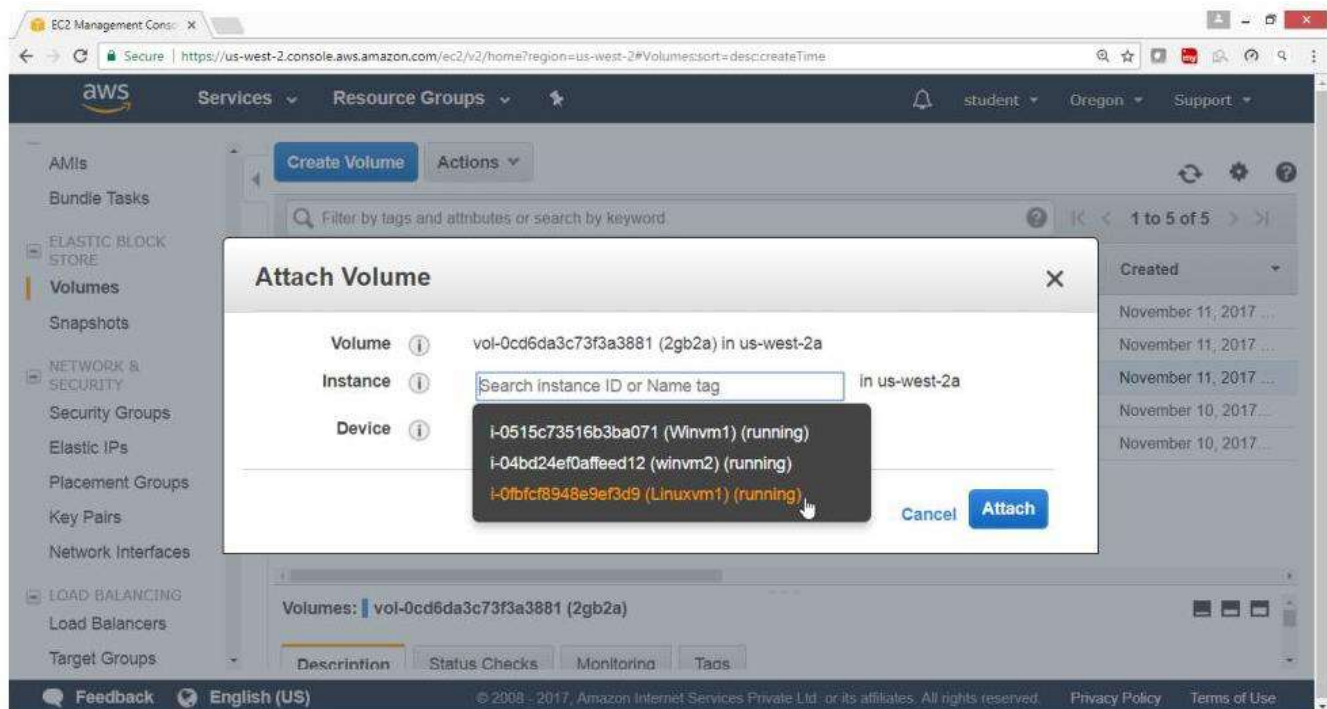


Similarly check volume in Linux instance

From Action Select "Attach Volume"



Select Linux Instance



Now connect to Linux Instance

```
[2017-11-11 12:58.46] /drives/e/awskeys
[shaikh.pc_mas] > ssh -i "studentaws.pem" ec2-user@ec2-54-244-106-102.us-west-2.com
ute.amazonaws.com
X11 forwarding request failed on channel 0
Last login: Sat Nov 11 07:28:43 2017 from 49.206.203.114

 _ _ | ( _ _ )
 _ _ | \ _ _ |   Amazon Linux AMI

https://aws.amazon.com/amazon-linux-ami/2017.09-release-notes/
[ec2-user@ip-172-31-40-234 ~]$ sudo su
[root@ip-172-31-40-234 ec2-user]#
```

To Verify

Switch to root user and run fdisk -l

\$ sudo su

To check the list of drives and partitions

#fdisk -l

```
[ec2-user@ip-172-31-40-234 ~]$ sudo su
[root@ip-172-31-40-234 ec2-user]#
[root@ip-172-31-40-234 ec2-user]# fdisk -l
WARNING: fdisk GPT support is currently new, and therefore in an experimental phase.
Use at your own discretion.

Disk /dev/xvda: 8589 MB, 8589934592 bytes, 16777216 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: gpt

#           Start          End          Size Type         Name
 1          4096        16777182         8G Linux fileyste Linux
128         2048          4095          1M BIOS boot parti BIOS Boot Partition

Disk /dev/xvdf: 2147 MB, 2147483648 bytes, 4194304 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xb9c39eba
```

Explain types of storage for the Root Device and difference between them?

There are 2 types of storage for the Root Device, as either backed by Amazon EBS or backed by Instance store. The former means that the root device for an instance launched from the AMI is an Amazon EBS volume created from an Amazon EBS snapshot.

This section summarizes the important differences between the two types of AMIs. The following table provides a quick summary of these differences.

Characteristic	Amazon EBS-Backed	Amazon Instance Store-Backed
Boot time	Usually less than 1 minute	Usually less than 5 minutes
Size limit	16 TiB	10 GiB
Root device volume	Amazon EBS volume	Instance store volume
Data persistence:		

By default, the root volume is deleted when the instance terminates. * Data on any other Amazon EBS volumes persists after instance termination by default. Data on any instance store volumes persists only during the life of the instance. Data on any instance store volumes persists only during the life of the instance. Data on any Amazon EBS volumes persists after instance termination by default.

Upgrading:

The instance type, kernel, RAM disk, and user data can be changed while the instance is stopped. Instance attributes are fixed for the life of an instance.

Charges:

You're charged for instance usage, Amazon EBS volume usage, and storing your AMI as an Amazon EBS snapshot. You're charged for instance usage and storing your AMI in Amazon S3.

AMI creation/bundling Uses a single command/call

Requires installation and use of AMI tools **Stopped State:** Can be placed in stopped state where instance is not running, but the root volume is persisted in Amazon EBS

Cannot be in stopped state: instances are running or terminated

How do you pass custom environment variable on Amazon Elastic Beanstalk (AWS EBS)?

As a heads up to anyone who uses the .ebextensions/*.config way: nowadays you can add, edit and remove environment variables in the Elastic Beanstalk web interface.

The variables are under Configuration

Software Configuration:

RAILS_SKIP_MIGRATIONS	false	X
This key-value pair will be made available to your application as an environment variable.		
SECRET_TOKEN	very-secret	X
This key-value pair will be made available to your application as an environment variable.		
CUSTOM_ENV	something-something	+

Cancel Save

What are the benefits of EBS vs. instance-store?

- EBS backed instances can be set so that they cannot be (accidentally) terminated through the API.
- EBS backed instances can be stopped when you're not using them and resumed when you need them again (like pausing a Virtual PC), at least with my usage patterns saving much more money than I spend on a few dozen GB of EBS storage.
- EBS backed instances don't lose their instance storage when they crash (not a requirement for all users, but makes recovery much faster)
- You can dynamically resize EBS instance storage.
- You can transfer the EBS instance storage to a brand-new instance (useful if the hardware at Amazon you were running on gets flaky or dies, which does happen from time to time)
- It is faster to launch an EBS backed instance because the image does not have to be fetched from S3.

Some of the Amazon EC instances types provide the option of using a directly attached block-device storage. This kind of storage is known as Instance Store. In other Amazon EC2 instances, we have to attach an Elastic Block Store (EBS).

Persistence: The main difference between Instance Store and EBS is that in Instance Store data is not persisted for long-term use. If the Instance terminates or fails, we can lose Instance Store data. Any data stored in EBS is persisted for longer duration. Even if an instance fails, we can use the data stored in EBS to connect it to another EC2 instance.

Encryption: EBS provides a full-volume encryption of data stored in it. Whereas Instance Store is not considered good for encrypting data.

Differences

- Instance Store Volumes are sometimes called Ephemeral Storage
- Instance store volumes cannot be stopped. If the underlying host fails, you will lose your data
- EBS backed instances can be stopped. You will not lose the data on this instance if it is stopped.
- You can reboot both, you will not lose your data
- By default, both ROOT volumes will be deleted on termination, however with EBS volumes, you can tell AWS to keep the root device volume

What is Snapshots?

A Snapshot is created by copying the data of a volume to another location at a specific time. We can even replicate same Snapshot to multiple availability zones.

How to create Snapshots of Root Device Volumes?

To create a snapshot for Amazon EBS volumes that serve as root devices, you should stop the instance before taking the snapshot.

How can I take a Snapshot of a RAID Array?

Problem - Take a snapshot excludes data held in the cache by applications and the OS. This tends not to matter on a single volume, however using multiple volumes in a RAID Array, this can be a problem due to inter dependencies of the array.

Solution:

- Take an application with consistent snapshot
- Stop the application from writing to disk
- Flush all caches to the disk

We can do this by

- Freeze the file system
- Unmount the RAID Array
- Shutting down the associated EC2 instance

What is the difference between Volume and Snapshot in Amazon Web Services?

In Amazon Web Services, a Volume is a durable, block level storage device that can be attached to a single EC2 instance. In plain words it is like a hard disk on which we can write or read from.

A Snapshot is created by copying the data of a volume to another location at a specific time. We can even replicate same Snapshot to multiple availability zones. So, Snapshot is a single point in time view of a volume. We can create a Snapshot only when we have a Volume. Also, from a Snapshot we can create a Volume. In AWS, we have to pay for storage that is used by a Volume as well as the one used by Snapshots.

Differences

- Volumes exist on EBS - Virtual Hard Disk
- Snapshot exist on S3
- You can take a snapshot of a volume, this will store that volume on S3
- Snapshots are point in time copies of volumes
- Snapshots are incremental, this means that only blocks that have changed since your last snapshots are moved to S3
- If this is your first snapshot, it may take some time to create

- Snapshots of encrypted volumes are encrypted automatically
- Volumes restored from encrypted snapshots are encrypted automatically
- You can share snapshots, but only if they are unencrypted
- These snapshots can be shared with other AWS accounts or made public

How you will change the root EBS device of my amazon EC2 instance?

- Stop the instance.
- Detach the root EBS volume.
- Attach the alternate EBS volume (as the root e.g. /dev/sda1)
- Start the instance.
- This presupposes that your alternate EBS volume is bootable, of course – it has to contain the bootable OS image.

Amazon Machine Image

Amazon Machine Image Highlights

An Amazon Machine Image (AMI) is a template that contains a software configuration (for example, an operating system, an application server, and applications). From an AMI, we launch an instance, which is a copy of the AMI running as a virtual server in the cloud. We can launch multiple instances of an AMI.

AMI's are regional. You can only launch an AMI from the region in which it is stored. However, you can copy AMI's to other regions using the console, command line or the Amazon EC2 API

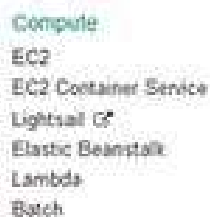
In-short

- AMI is a Server template | VM Image
- AMI Comes with pre-installed OS and optional S/W
- AMI can be launched to create instances
- A variety of pre-built AMIs in the catalog
- Existing AMIs can be customized and saved (Bundling)
- Independent of the configuration
- Every AMI is uniquely identified

Share how to create Amazon Machine Image with Linux Configuration Step by Step?

STEP#1: Login to Amazon Web Service Console

The AWS Management Console is a web control panel for managing all your AWS resources, from EC2 instances. The Console enables cloud management for all aspects of the AWS account, including managing security credentials, or even setting up new IAM Users.



STEP#2: Select the right AWS Region

Amazon Web Services is available in different Regions all over the world and the Console lets you provision resources across multiple regions. You usually choose a region those best suits your business needs to optimize your customer's experience

SUVEN IT ▾

N. California ▲

US East (N. Virginia)

US West (N. California)

US West (Oregon)

EU (Ireland)

EU (Frankfurt)

Asia Pacific (Tokyo)

Asia Pacific (Seoul)

Asia Pacific (Singapore)

Asia Pacific (Sydney)

South America (São Paulo)

Create an AMI starting from an EBS-backed instance

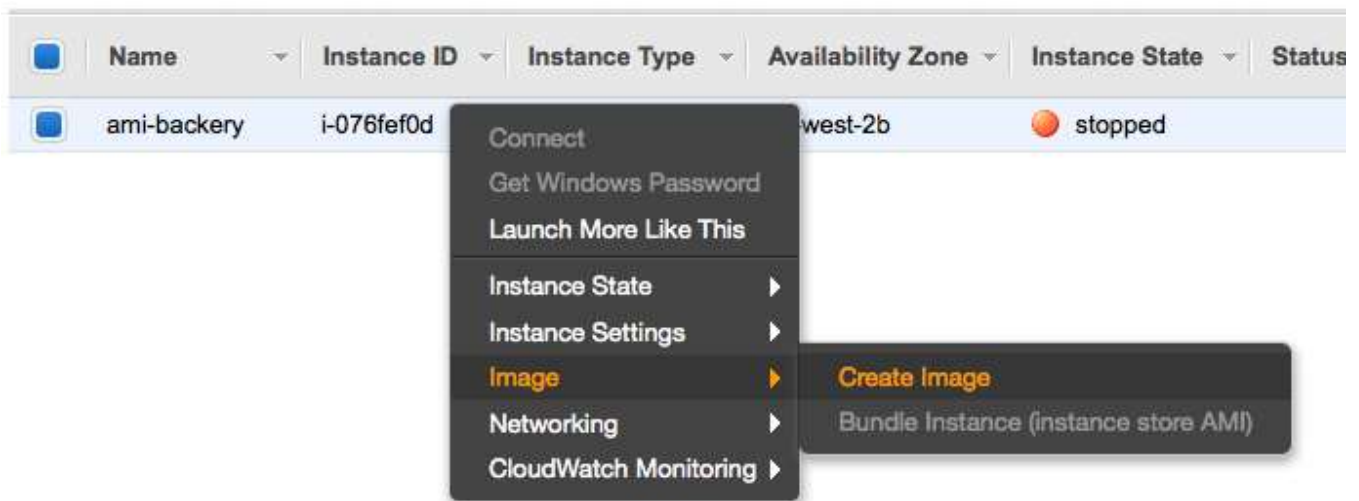
An AMI contains all information necessary to boot an Amazon EC2 instance with your software. An AMI is like a virtual machine template and it might contain custom software, standard system packages or any other file added by the AMI author. Creating your own AMI is a crucial operation if you have to build a clustered infrastructure that uses the EC2 Autoscaling Group feature.

AWS Auto Scaling needs self-configurable instances in order to automatically scale up or down your cluster according to the specified policies. Your AMI becomes the basic unit of deployment; it enables you to rapidly boot new custom instances as you need them.

All AMIs are categorized as either backed by Amazon EBS or backed by instance store. The former means that the root device for an instance launched from the AMI is an Amazon EBS volume created from an Amazon EBS snapshot. The latter means that the root device for an instance launched from the AMI is an instance store volume created from a template stored in Amazon S3. You can implement Amazon EBS backed AMIs by creating a set of snapshots and registering an AMI that uses those snapshots. The AMI publisher controls the default size of the root device through the size of the snapshot.

Creating an AMI from an EBS-backed instance is an easy and automated task.

- Go to the Instances section of the EC2 Console
- Locate the previously created instance, select it and then right click on it.
- Select Image submenu and click on Create Image.



Enter the Image name, the Image description and check the Instances Volumes configuration. You can choose to add more volumes of different types and sizes.

When you have been finished, click on Create Image blue button.

The screenshot shows the 'Create Image' dialog box. It has a title bar 'Create Image' with a close button. Below the title bar, there are several fields: 'Instance ID' (i-076fef0d), 'Image name' (cloudacademy-labs-webserver-basic), 'Image description' (Ubuntu image with nginx, php, git, awscli), and 'No reboot' (checkbox). Below these fields is the 'Instance Volumes' section. It contains a table with columns: Type, Device, Snapshot, Size (GiB), Volume Type, IOPS, Delete on Termination, and Encrypted. The table has one row for the 'Root' volume. Below the table is an 'Add New Volume' button. At the bottom of the dialog, there are 'Cancel' and 'Create Image' buttons.

Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Delete on Termination	Encrypted
Root	/dev/sda1	snap-ddd48814	8	General Purpose (SSD)	24 / 3000	<input checked="" type="checkbox"/>	Not Encrypted

The AMI creation takes some minutes to be processed, because AWS has to create an EBS snapshot and then register the newly created AMI. You can check the status by going to the Snapshot section and then to the AMI section.

Create Snapshot

Actions

Owned By Me

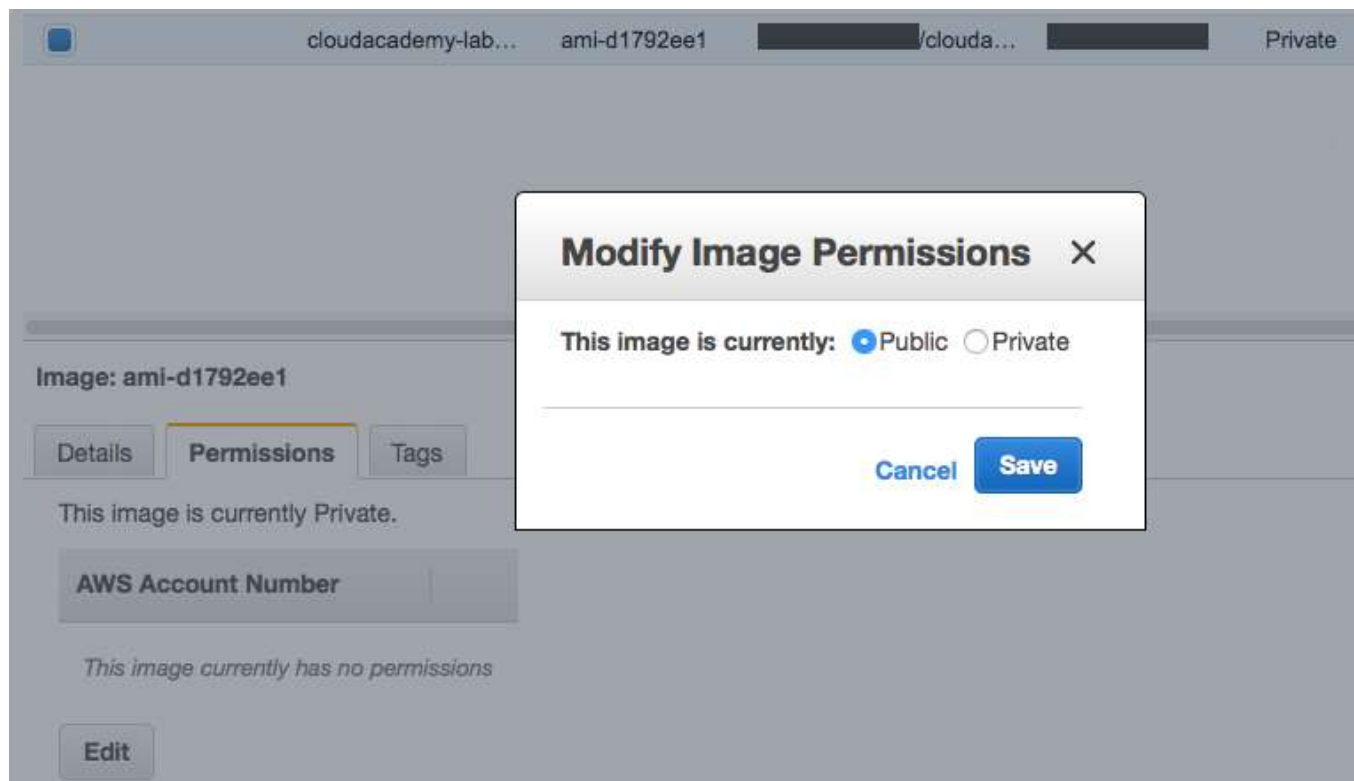
Filter by tags and attributes or search by keyword

	Name	Snapshot ID	Size	Description	Status
		snap-a1fc262d	8 GiB	Created by CreateImage(i-076fef0d) for ami-d1792ee1 from vol...	 pending

When the AMI status switches from pending to available, you are able to create new EC2 instances by using it.

Make public an AMI

After the creation of an AMI, you are the only user able to use it during the EC2 launching process. If you want to allow the deployment of new EC2 instances starting from your AMI, you have to edit the Image permissions.



Select your AMI, click on the Permissions Tab and then on the Edit button.

You can choose to make it publicly available or to allow its usage only to a restricted set of AWS accounts.

What does an AMI include?

An AMI includes the following things

- A **template** for the root volume for the instance
- **Launch permissions** decide which AWS accounts can avail the AMI to launch instances
- A **block device mapping** that determines the volumes to attach to the instance when it is launched

Where do you think an AMI fits, when you are designing an architecture for a solution?

AMIs (Amazon Machine Images) are like templates of virtual machines and an instance is derived from an AMI. AWS offers pre-baked AMIs which you can choose while you are launching an instance, some AMIs are not free, therefore can be bought from the AWS Marketplace. You can also choose to create your own custom AMI which would help you save space on AWS. For example, if you don't need a set of software on your installation, you can customize your AMI to do that. This makes it cost efficient, since you are removing the unwanted things.

What is the relation between Instance and AMI?

AMI can be elaborated as Amazon Machine Image, basically, a template consisting software configuration part. For example, an OS, applications, application server. If you start an instance, a duplicate of the AMI in a row as an unspoken attendant in the cloud.

We 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.

After we launch an instance, it looks like a traditional host, and we can interact with it as we would any computer. We have complete control of our instances; we can use sudo to run commands that require root privileges.

Amazon Web Services provides several ways to access Amazon EC2, like web-based interface, AWS Command Line Interface (CLI) and Amazon Tools for Windows Powershell. First you need to signed up for an AWS account and you can access Amazon EC2.

Amazon EC2 provides a Query API. These requests are HTTP or HTTPS requests that use the HTTP verbs GET or POST and a Query parameter named Action.

How to determine the Root Device type of your AMI?

We can determine the Root Device type of AMI using following 2 methods.

Method 1: Following are the steps to determine the Root Device type of an AMI using the console

- 1.1 Open the Amazon EC2 console
- 1.2 In the navigation pane, click AMIs, and select the AMI
- 1.3 Check the value of Root Device Type in the Details tab as follows

1.3.1 If the value is ebs, this is an Amazon EBS-backed AMI

1.3.2 If the value is instance store, this is an instance store-backed AMI

Method 2: Following are the steps to determine the root device type of an AMI using the command line

We can use one of the following commands.

2.1 describe-images (AWS CLI)

2.2 Get-EC2Image (AWS Tools for Windows PowerShell)

What is the size limit for Amazon EC2 instance store-backed AMIs and Amazon EBS-backed AMIs?

- All AMIs are categorized as either backed by Amazon EBS or backed by instance store.
- Backed by Amazon EBS – means that the root device for an instance launched from the AMI is an Amazon EBS volume created from an Amazon EBS snapshot.
- Backed by instance store – means that the root device for an instance launched from the AMI is an instance store volume created from a template stored in Amazon S3.
- Root device size limit for –
 - Amazon EBS – Backed is 16 TiB
 - Amazon Instance Store-Backed is 10 GiB

What is shared AMI?

A shared AMI is an AMI that a developer created and made available for other developers to use.

One of the easiest ways to get started with Amazon EC2 is to use a shared AMI that has the components you need and then add custom content. You can also create your own AMIs and share them with others.

Note: Use a shared AMI at your own risk. Amazon can't vouch for the integrity or security of AMIs shared by other Amazon EC2 users. AWS recommends that you get an AMI from a trusted source.

How to update AMI tools at Boot Time?

AWS recommends that your AMIs download and upgrade the Amazon EC2 AMI creation tools during startup. This ensures that new AMIs based on your shared AMIs have the latest AMI tools.

For Amazon Linux, add the following to `/etc/rc.local`:

```
# Update the Amazon EC2 AMI tools
echo " + Updating EC2 AMI tools"
yum update -y aws-amitools-ec2
echo " + Updated EC2 AMI tools"
```

How to create your own Amazon Machine Image (AMI)?

You can customize an instance that is launched from a public AMI and then save that configuration as a custom AMI for your own use. Instances that you launch from your AMI use all the customizations that you've made.

Amazon EC2 Auto-scaling

Auto-scaling Highlights

Auto-scaling is the ability of a system to scale itself automatically based on the triggers like- crashing of a server or low performance. AWS extensively supports Auto-scaling. It provides tools to create, configure and automatically start new instances without any manual intervention. We can set the thresholds at which new instances will come up. Or we can monitor the metrics like API response time, number of requests per seconds and based on these metrics, let the AWS provision and start new servers.

Auto- scaling is one of the remarkable features of AWS where it permits you to arrange and robotically stipulation and spin up fresh examples without the requirement for your involvement. This can be achieved by setting brinks and metrics to watch. If those entrances are overcome, a fresh example of your selection will be configured, spun up and copied into the weight planner collection.

An Auto Scaling group is a representation of multiple Amazon EC2 instances that share similar characteristics, and that are treated as a logical grouping for the purposes of instance scaling and management.

For example, if a single application operates across multiple instances, you might want to increase or decrease the number of instances in that group to improve the performance of the application. You can use the Auto Scaling group to automatically scale the number of instances or maintain a fixed number of instances. You create Auto Scaling groups by defining the minimum, maximum, and desired number of running EC2 instances the group must have at any given point of time.

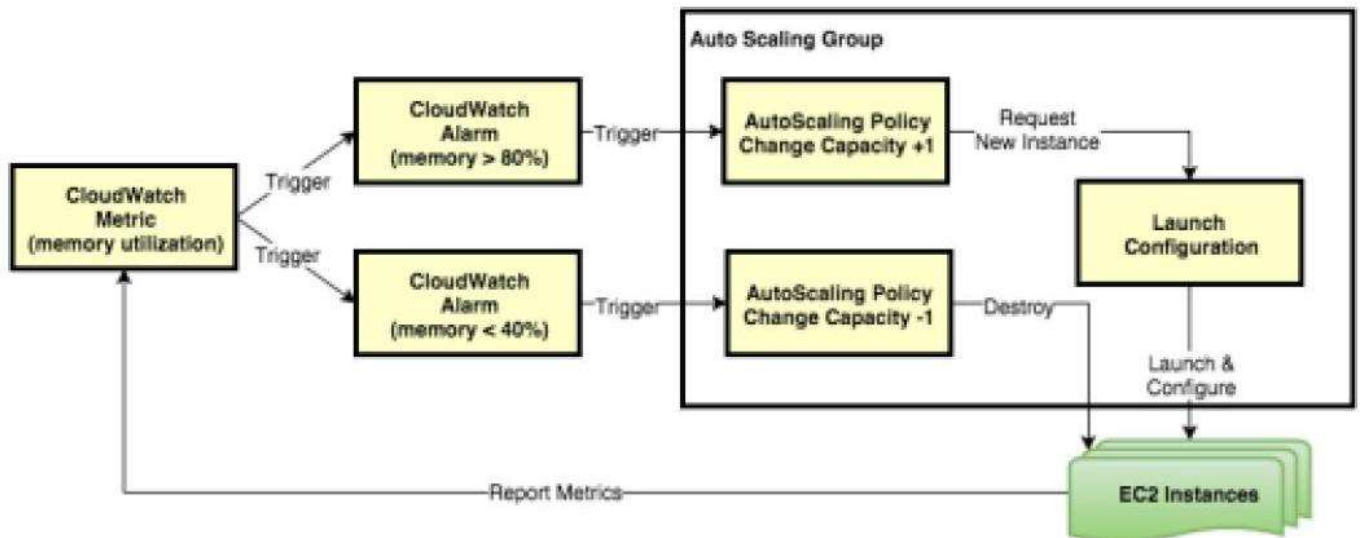
An Auto Scaling group starts by launching the minimum number (or the desired number, if specified) of EC2 instances and then increases or decreases the number of running EC2 instances automatically according to the conditions that you define.

Auto Scaling also maintains the current instance levels by conducting periodic health check on all the instances within the Auto Scaling group. If an EC2 instance within the Auto Scaling group becomes unhealthy, Auto Scaling terminates the unhealthy instance and launches a new one to replace the unhealthy instance. This automatic scaling and maintenance of the instance levels in an Auto Scaling group is the core value of the Auto Scaling service.

Share the Auto Scaling Group Configuration Step by Step?

To configure Autoscaling group in AWS

Topology



Pre-requisites

User should have AWS account, or IAM user with EC2FullAccess

Task

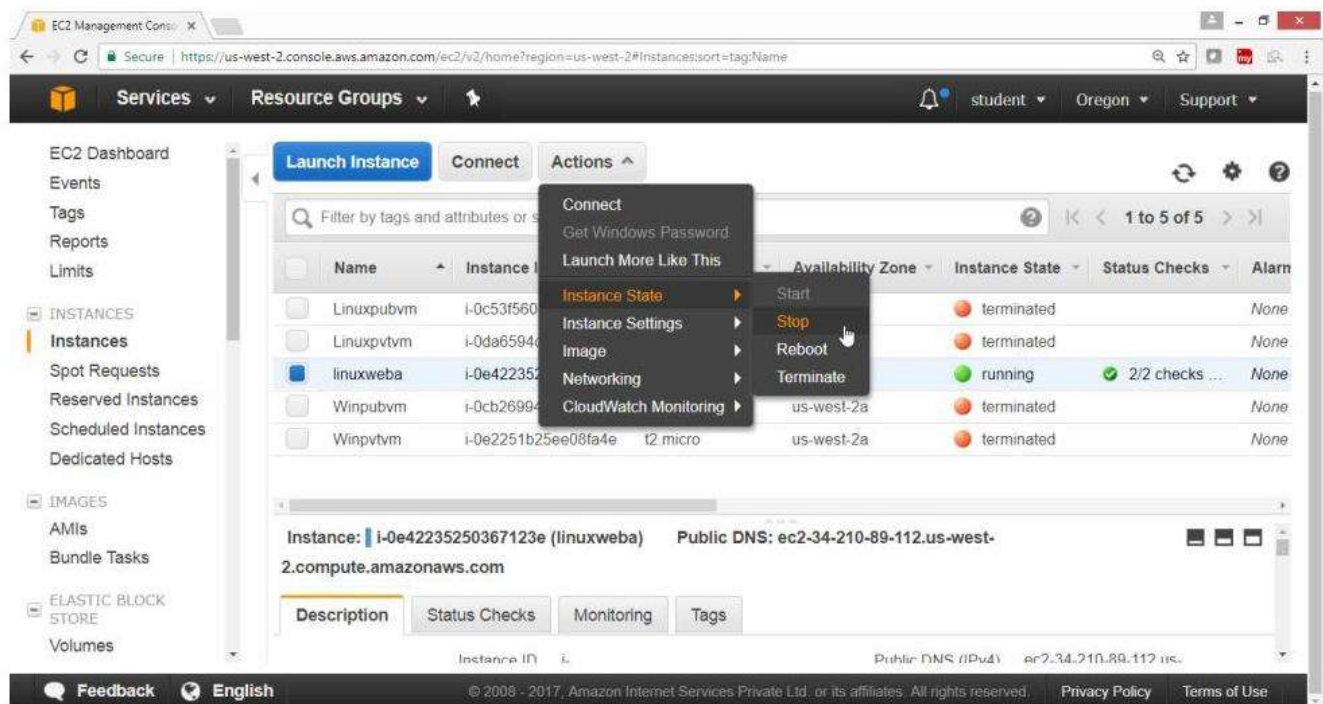
- Launch Amazon Linux Instance
- Configure Web Server
- Stop the Instance
- Create AMI image of above instance
- Configure Autoscaling launch configuration and autoscaling group
- Configure Load balancer with Autoscaling

1) First launch Amazon Linux Instance and configure Webserver

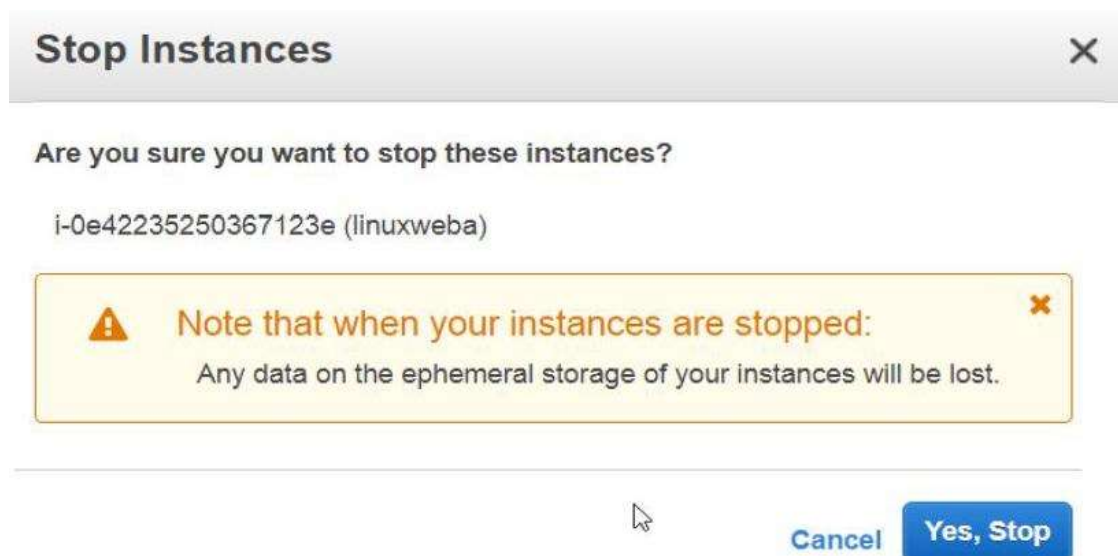
2) Create an Amazon Machine Image

To create AMI from this instance

- On "EC2 Dashboard" panel
- Click on "Action" Button
- Select Instance State
- Click Stop

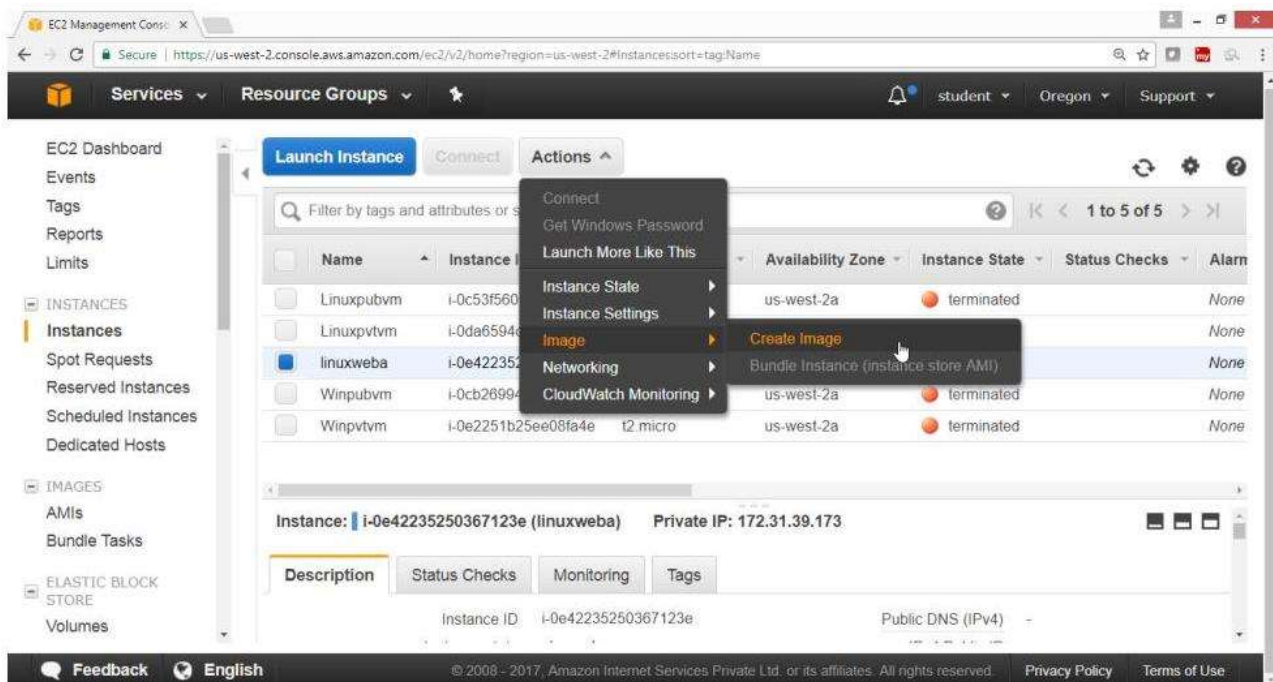


Click on "Yes Stop" Button

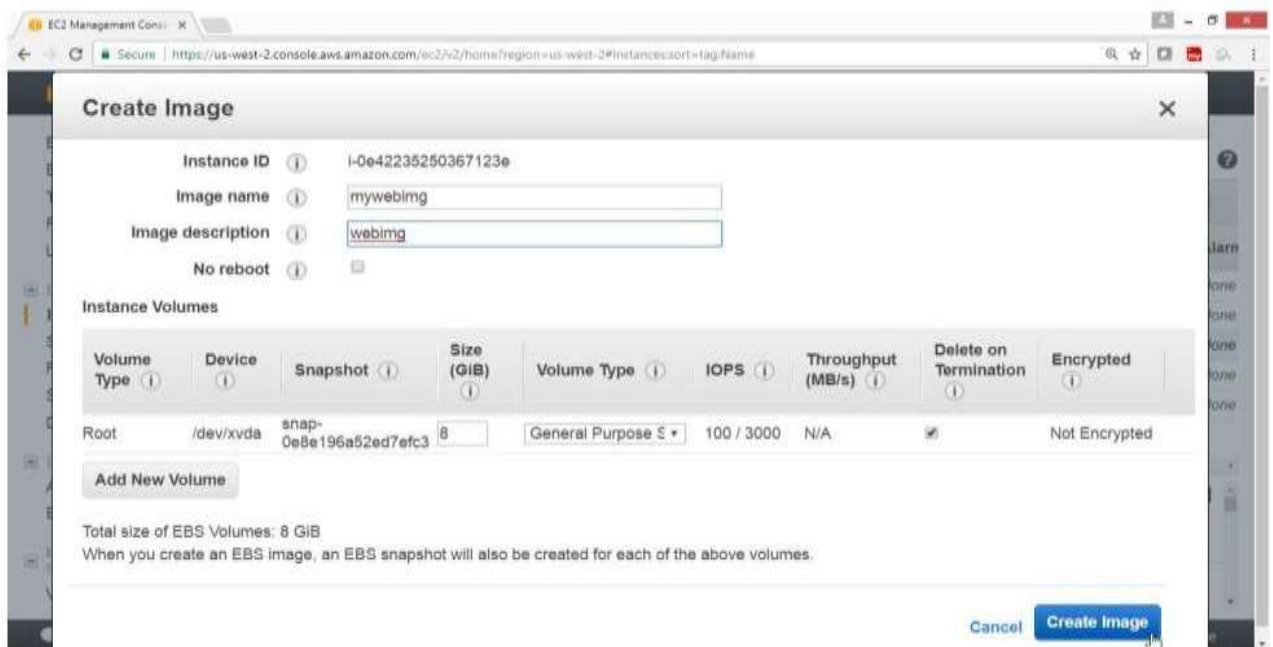


Select the stopped instance

- Click on "Action" Button
- Select Image
- Click on "Create Image" button



From Image name-> mywebimg
 For Image description->webimg
 Leave remaining default
 Click on Create image button



Click on Close button

Create Image



Create Image request received.

[View pending image ami-3ffe1947](#)

Any snapshots backing your new EBS image can be managed on the [snapshots screen](#) after successful image creation.

Close

Verify AMI is created

On the "EC2 Dashboard" panel

Select "Images"

Click on "AMIs"

Check the status is "available"

EC2 Management Console

Services ▾ Resource Groups ▾

student ▾ Oregon ▾ Support ▾

EC2 Dashboard
Events
Tags
Reports
Limits

INSTANCES
Instances
Spot Requests
Reserved Instances
Scheduled Instances
Dedicated Hosts

IMAGES
AMIs
Bundle Tasks

ELASTIC BLOCK STORE
Volumes

Launch Actions

Owned by me Filter by tags and attributes or search by keyword 1 to 1 of 1

Name	AMI Name	AMI ID	Source	Owner	Visibility	Status
mywebimg	ami-3ffe1947	523251683217/...	523251683217	Private	available	

Image: ami-3ffe1947

Details Permissions Tags

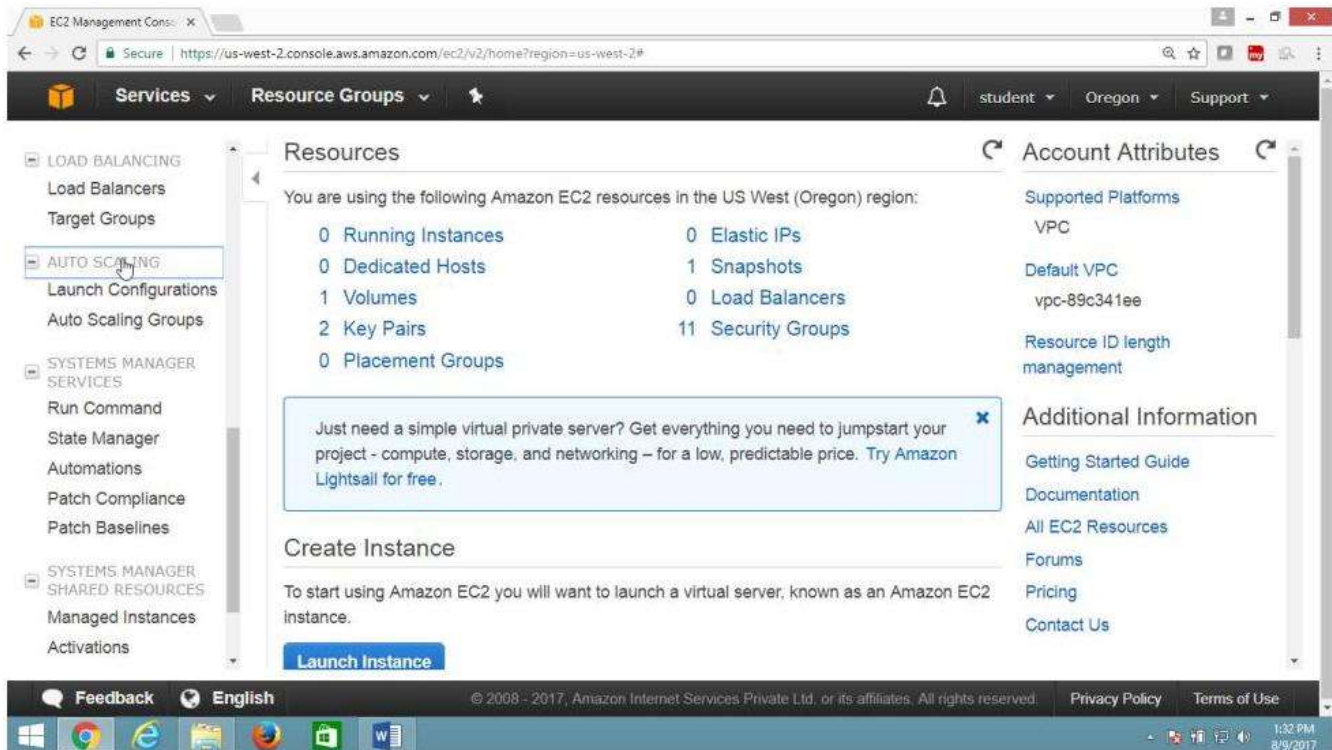
Edit

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

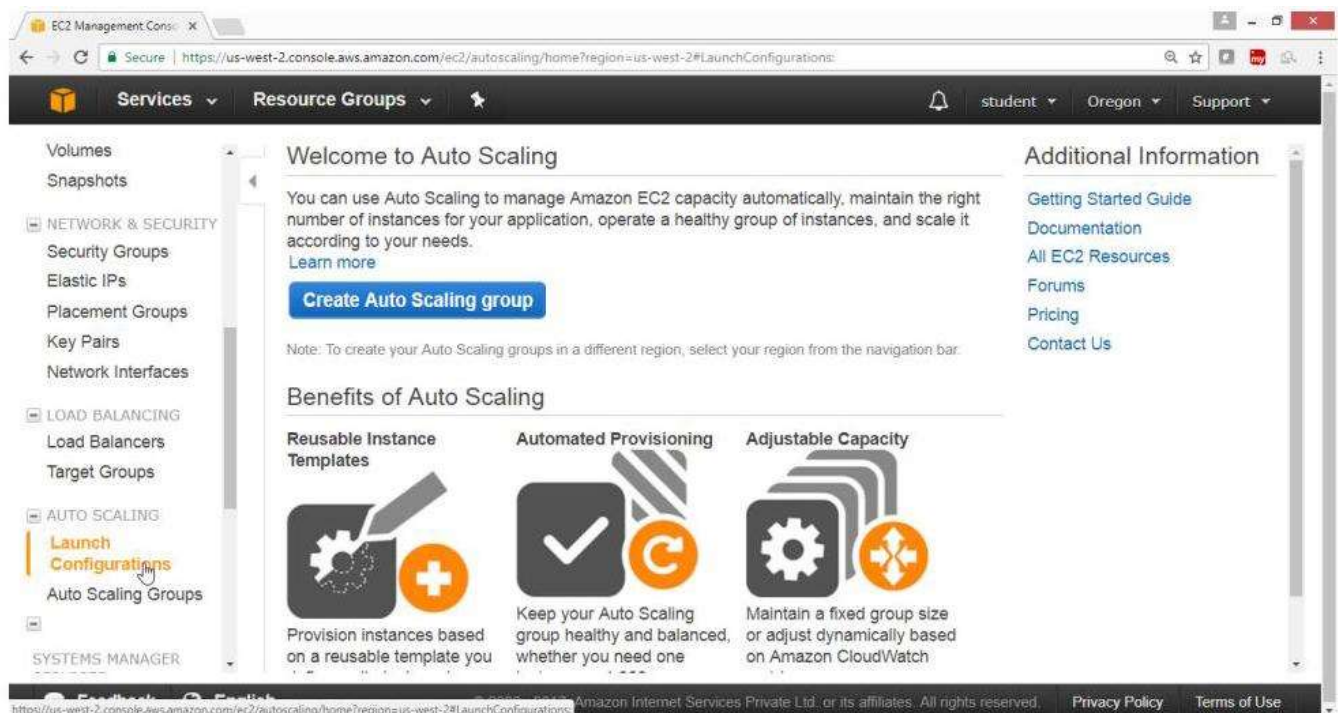
3) To Configure Auto Scaling

On the EC2 Dashboard Panel

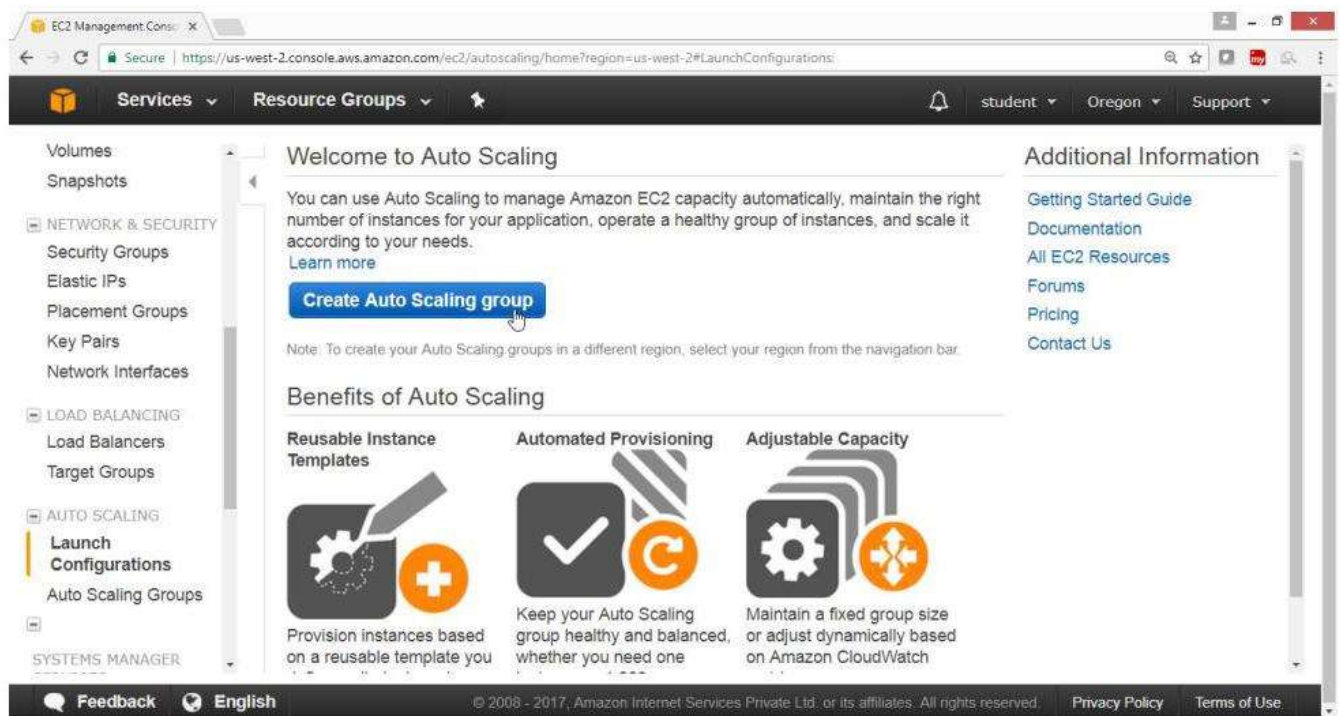
Select "AUTO SCALING"



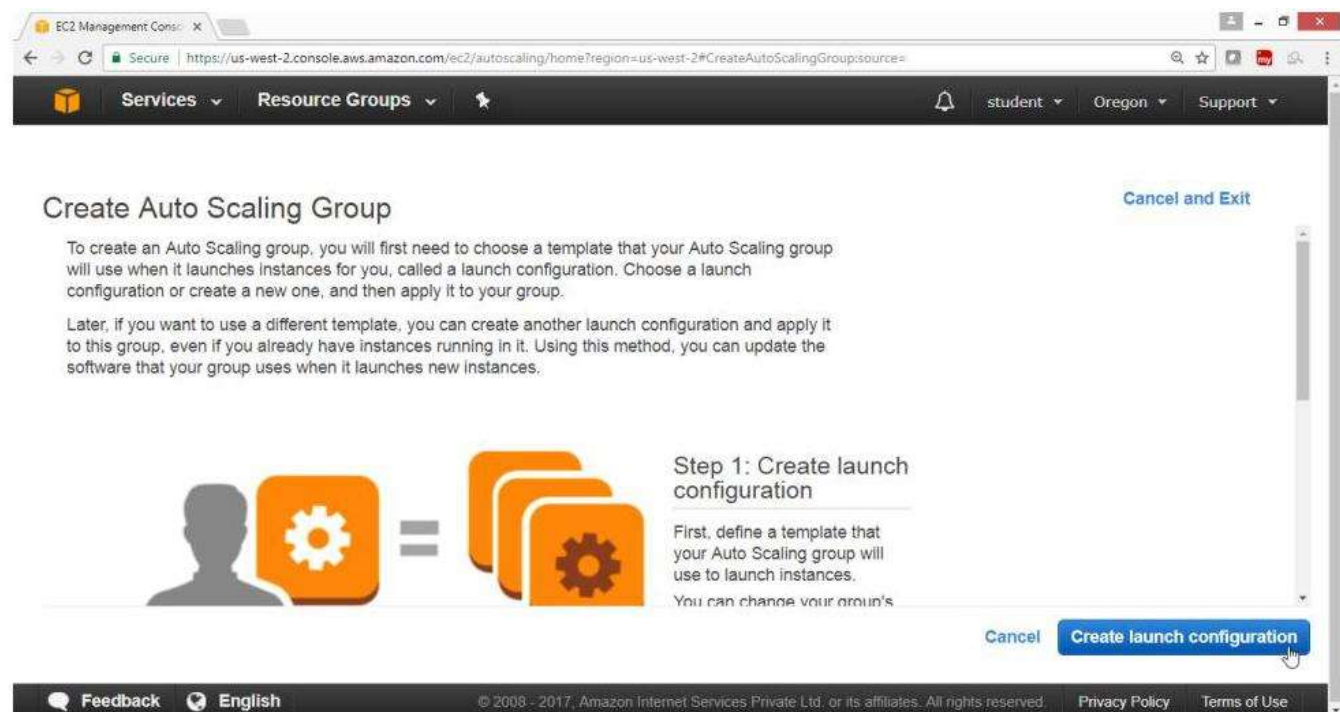
Click on "Launch Configuration"



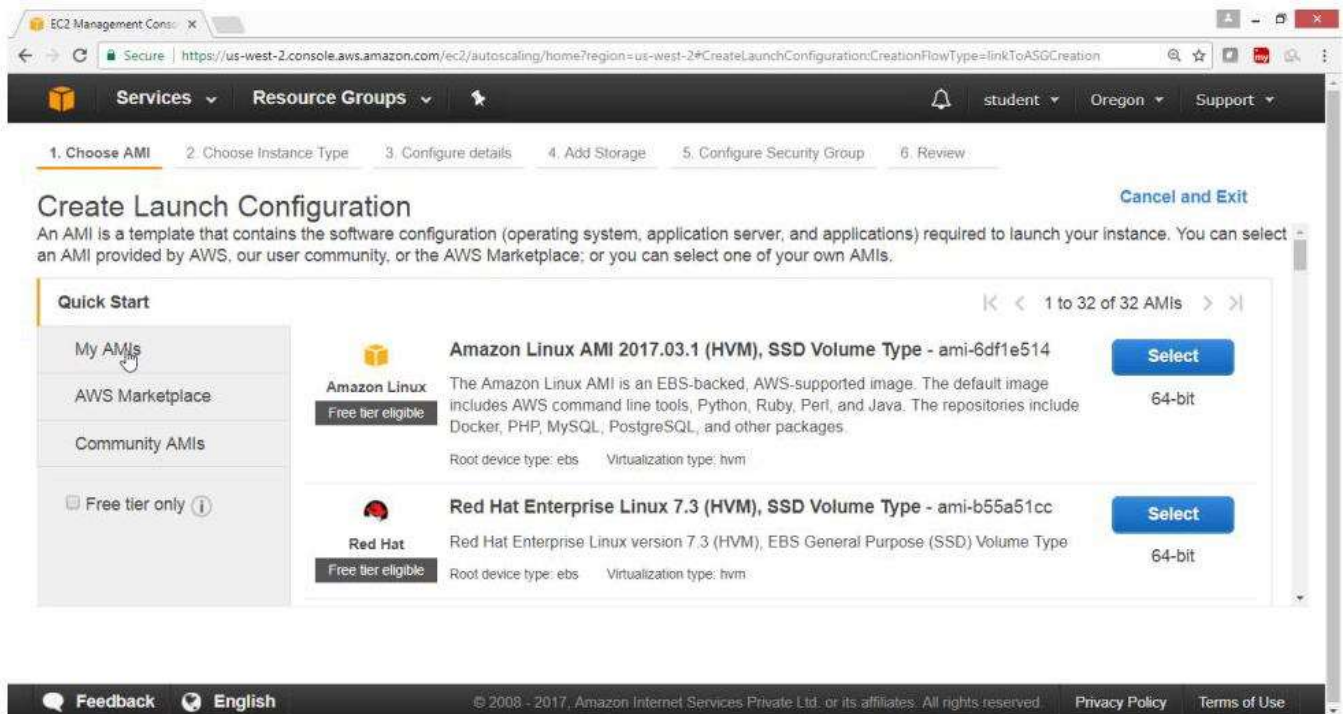
Click on "Create Auto Scaling Group" Button



Click on "Create Launch Confirmation" Button

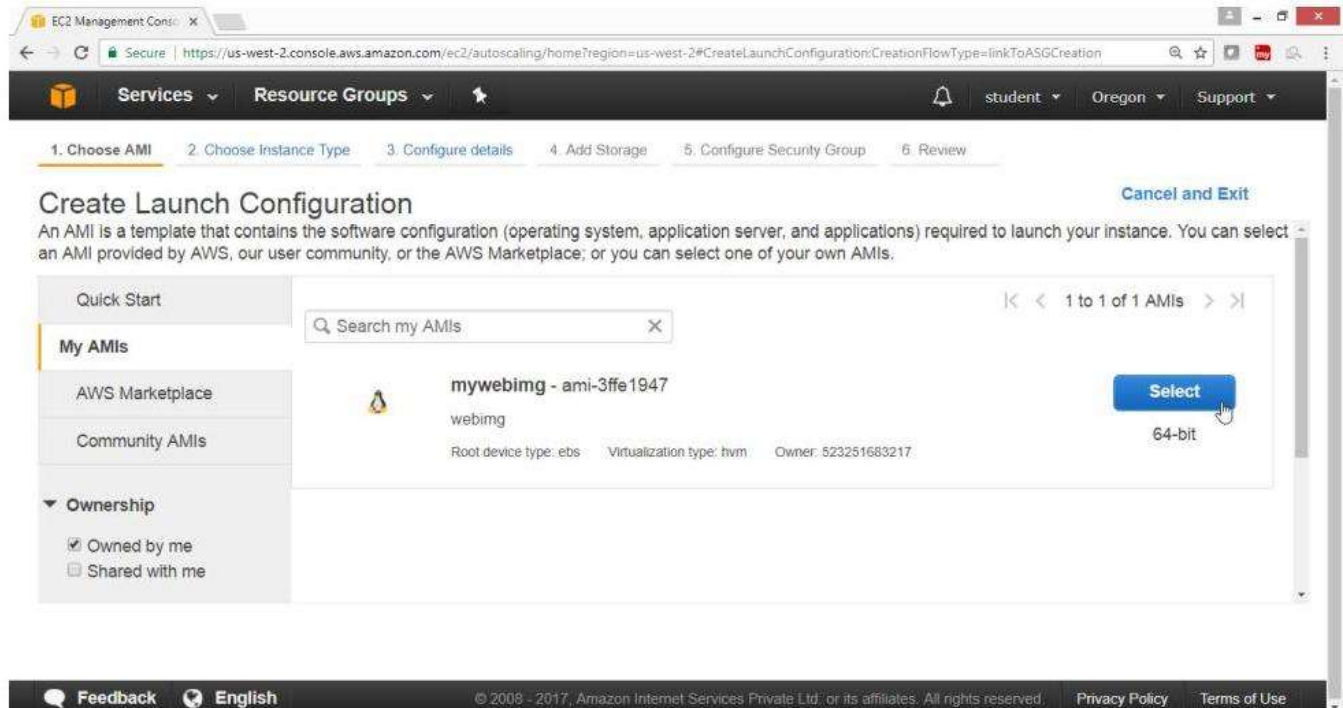


Click on "MyAMI"



Select the AMI which was created with Webserver

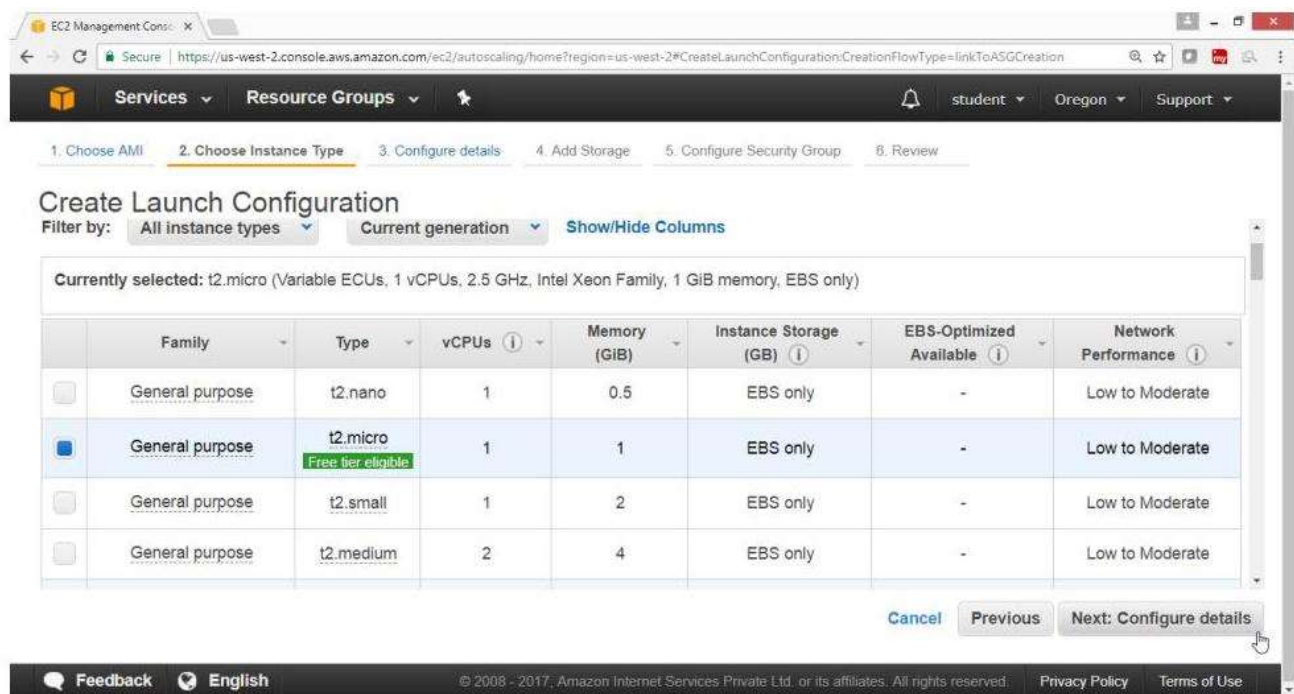
Click on "Select" Button



Choose Instance Type

General purpose, t2.micro free tier

Click on Next: Configuration Details

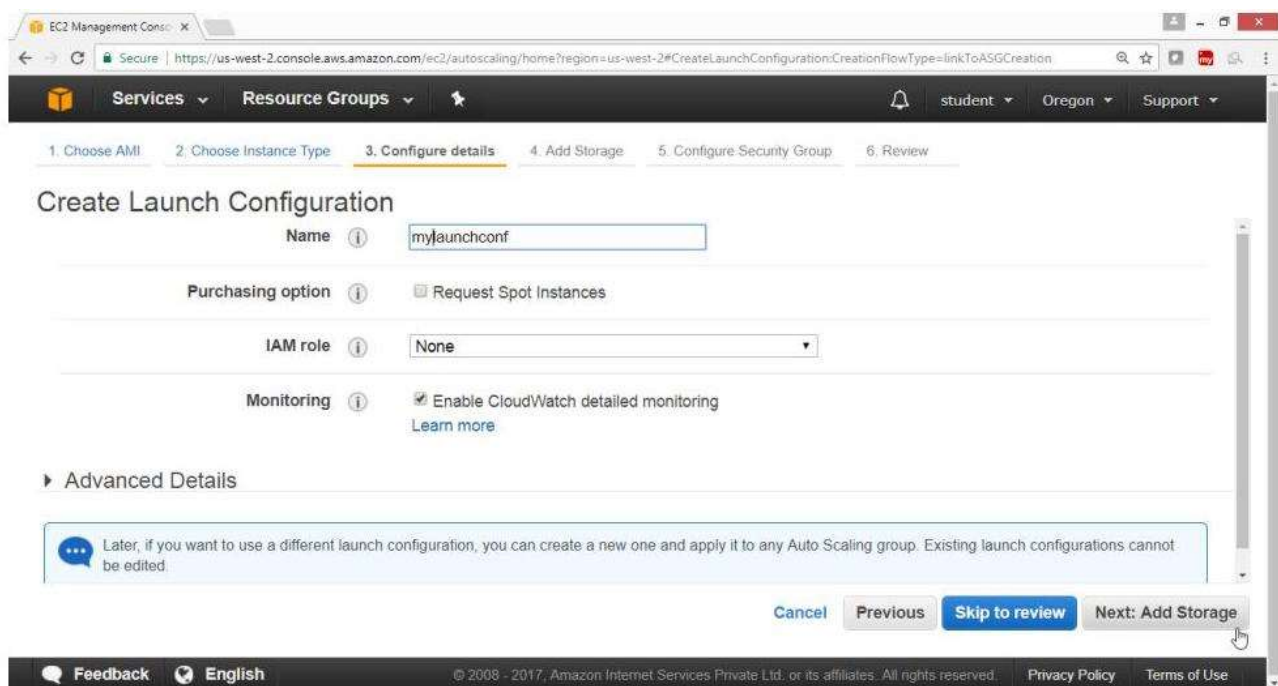


On "Create Launch Configuration" Page

Name -> mylaunchconf

Monitoring->Enable check box

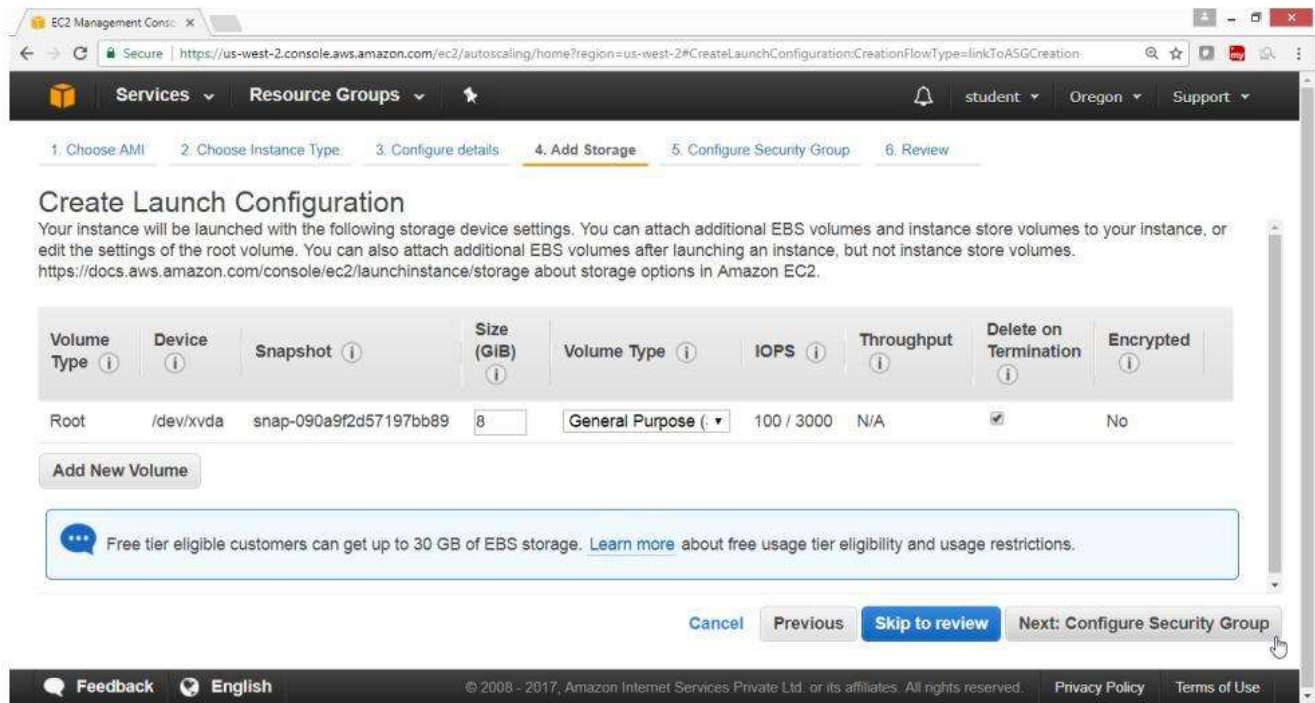
Click on Next: Add Storage button



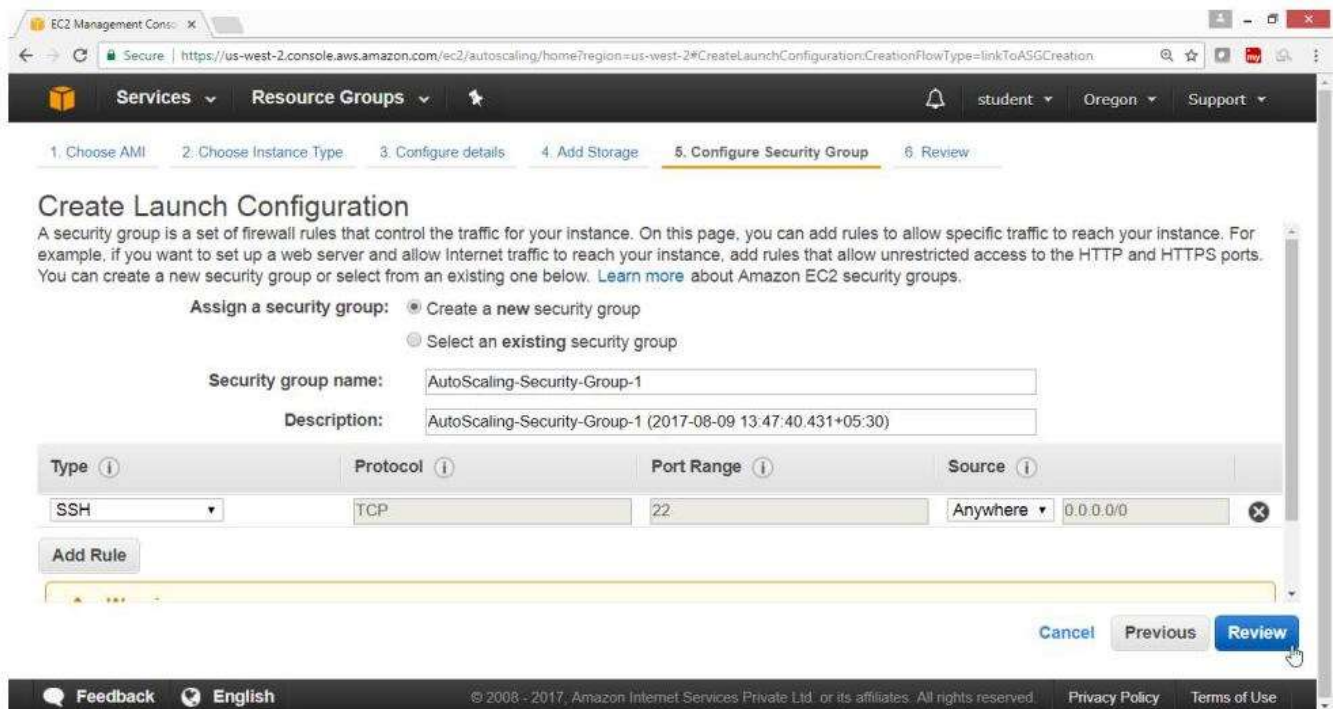
By default, Linux takes 8 GB EBS Volume

Leave all values as default

Click on "Next: Configure Security Group" Button



On Create Launch Configuration Page
Select "Create a new security Group"
Click on Review



Check the summary

Click on "Create launch configuration" Button

- On "Select an existing pair or create a new key pair" page
- Select "Choose an existing key pair"
- Select a key pair -> student
- Select Acknowledge check box
- Click on " Create launch configuration" Button

EC2 Management Console


Securehttps://us-west-2.console.aws.amazon.com/ec2/autoscaling/home?region=us-west-2#CreateLaunchConfiguration:CreationFlowType=linkToASGCreation

ServicesResource GroupsstudentOregonSupport

1. Choose AMI2. Choose Instance Type3. Configure details4. Add Storage5. Configure Security Group6. Review

Create Launch Configuration


Review the details of your launch configuration. You can go back to edit the details of each section before you finish.



Improve security of instances launched using your launch configuration, mylaunchconf. Your security group, AutoScaling-Security-Group-1, 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



mywebimg - ami-3ffe1947

webimg
Root device type: ebs Virtualization Type: hvm

[Edit AMI](#)


Instance Type

[Edit instance type](#)

[Cancel](#) [Previous](#) [Create launch configuration](#)

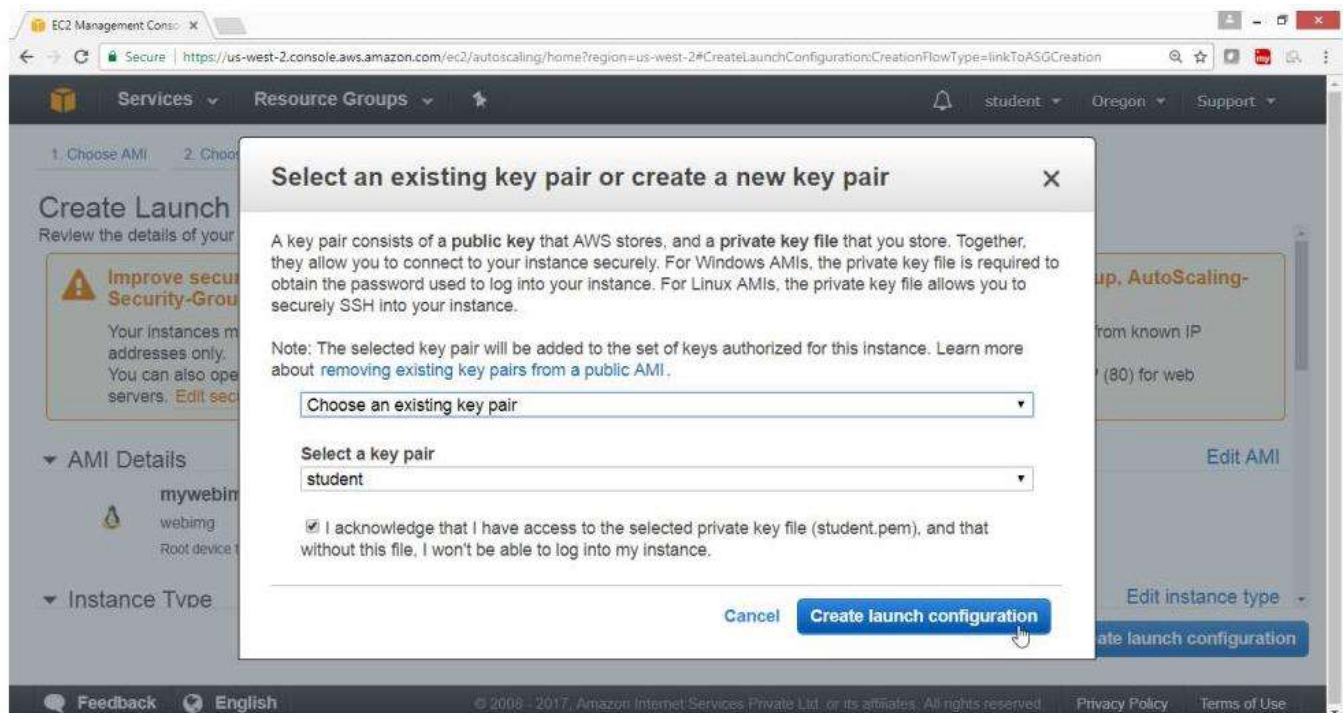
FeedbackEnglish

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)



1:50 PM
8/9/2017

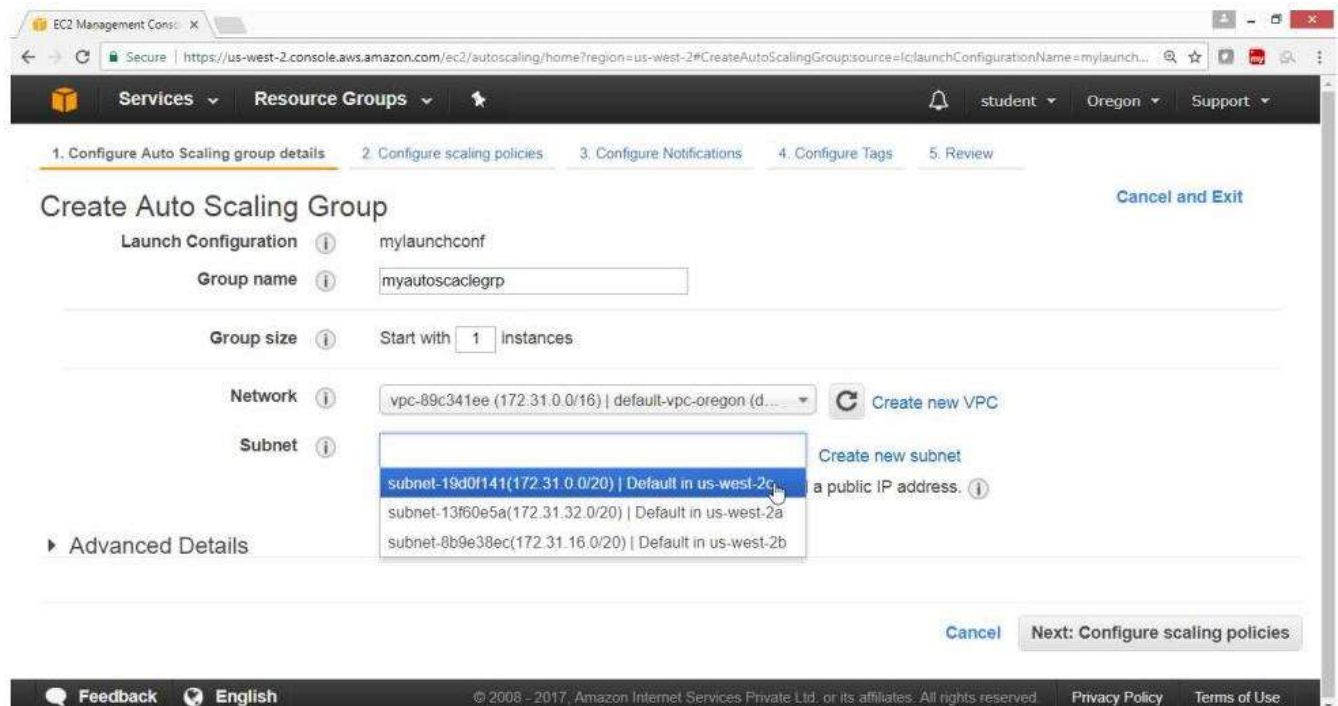
On "Create Auto Scaling Group" page, give values as



Launch Configuration-> mylaunchconf

Group Name->myautoscalegrp

For Network->select default



Select All subnet one by one

Click on "Next configure scaling policies" button

The screenshot shows the 'Create Auto Scaling Group' page in the AWS Management Console, specifically the 'Configure scaling policies' step. The page has a top navigation bar with 'Services', 'Resource Groups', and user information. Below the navigation bar is a progress indicator with five steps: 1. Configure Auto Scaling group details, 2. Configure scaling policies (active), 3. Configure Notifications, 4. Configure Tags, and 5. Review. The main content area is titled 'Create Auto Scaling Group' and includes a 'Cancel and Exit' link. The form contains several sections: 'Launch Configuration' with a dropdown set to 'mylaunchconf'; 'Group name' with a text input 'myautoscalegrp'; 'Group size' with a dropdown set to 'Start with 1 instances'; 'Network' with a dropdown set to 'vpc-89c341ee (172.31.0.0/16) | default-vpc-oregon (d...)' and a 'Create new VPC' button; and 'Subnet' with a list of three subnets: 'subnet-19d0f141 (172.31.0.0/20) | Default in us-west-2c', 'subnet-13f60e5a (172.31.32.0/20) | Default in us-west-2a', and 'subnet-8b9e38ec (172.31.16.0/20) | Default in us-west-2b'. There are 'Create new VPC' and 'Create new subnet' buttons. At the bottom right, there are 'Cancel' and 'Next: Configure scaling policies' buttons. The footer includes 'Feedback', 'English', copyright information, and 'Privacy Policy' and 'Terms of Use' links.

On "Create Auto Scaling Group" page

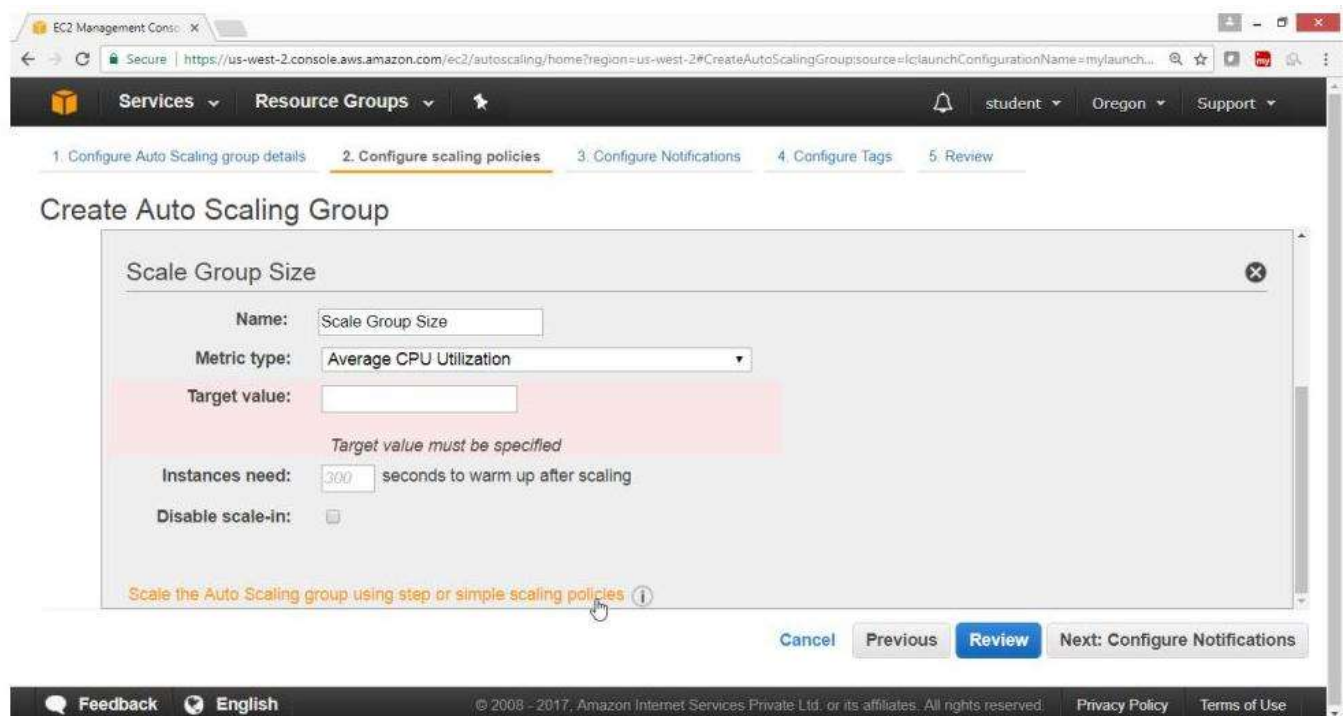
Select "Use scaling policies to adjust the capacity of this group"

Scale between ----- and ---- instances

The screenshot shows the 'Create Auto Scaling Group' page in the AWS Management Console, specifically the 'Configure scaling policies' step. The page has a top navigation bar with 'Services', 'Resource Groups', and user information. Below the navigation bar is a progress indicator with five steps: 1. Configure Auto Scaling group details, 2. Configure scaling policies (active), 3. Configure Notifications, 4. Configure Tags, and 5. Review. The main content area is titled 'Create Auto Scaling Group' and includes a 'Cancel and Exit' link. The form contains several sections: 'Launch Configuration' with a dropdown set to 'mylaunchconf'; 'Group name' with a text input 'myautoscalegrp'; 'Group size' with a dropdown set to 'Start with 1 instances'; 'Network' with a dropdown set to 'vpc-89c341ee (172.31.0.0/16) | default-vpc-oregon (d...)' and a 'Create new VPC' button; and 'Subnet' with a list of three subnets: 'subnet-19d0f141 (172.31.0.0/20) | Default in us-west-2c', 'subnet-13f60e5a (172.31.32.0/20) | Default in us-west-2a', and 'subnet-8b9e38ec (172.31.16.0/20) | Default in us-west-2b'. There are 'Create new VPC' and 'Create new subnet' buttons. At the bottom right, there are 'Cancel' and 'Next: Configure scaling policies' buttons. The footer includes 'Feedback', 'English', copyright information, and 'Privacy Policy' and 'Terms of Use' links.

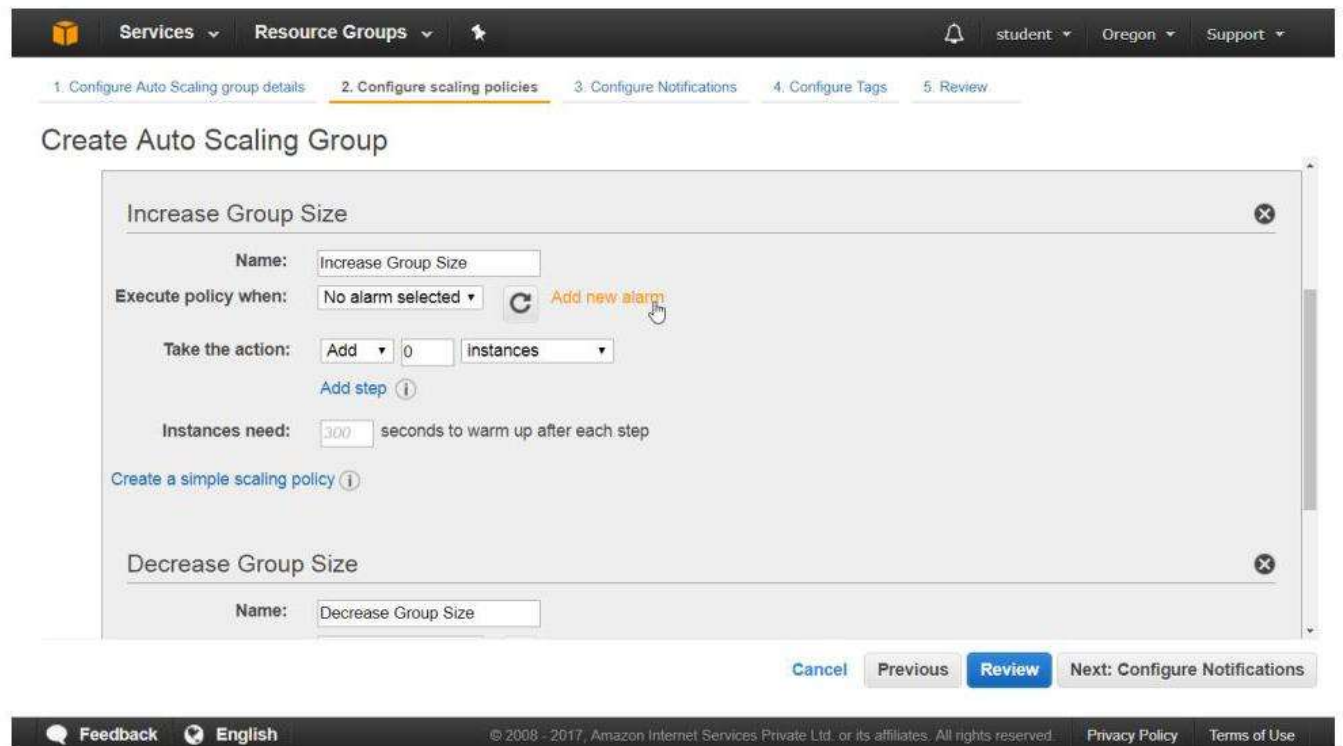
Drag Down

Click on "Scale the Auto Scaling group using step or simple scaling policies"



Select Increase Group Size

Click on "Add new alarm"



Click on "create topic"

Create Alarm



You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define.

To edit an alarm, first choose whom to notify and then define when the notification should be sent.

☒ Send a notification to: No SNS topics found... [create topic](#)

Whenever: Average of CPU Utilization

Is: \geq Percent

For at least: 1 consecutive period(s) of 5 Minutes

Name of alarm: awsec2-myautoscalegrp-High-CPU-Utilization

CPU Utilization Percent



Cancel

Create Alarm

On "Create Alarm" box, give values as
Send a notification to -> Cpuutilizationabc
With this recipients -> <<email id>
Whenever average of CPU Utilization
is \geq 30
Remaining value leave default
Click on "Create Alarm" button

Create Alarm



You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define.

To edit an alarm, first choose whom to notify and then define when the notification should be sent.

☒ Send a notification to: Cpuutilizationabc [cancel](#)

With these recipients: skmarhaan999@gmail.com

Whenever: Average of CPU Utilization

Is: \geq 30 Percent

For at least: 1 consecutive period(s) of 5 Minutes

Name of alarm: awsec2-myautoscalegrp-High-CPU-Utilization

CPU Utilization Percent



Cancel

Create Alarm

For Take the action -> Add1

Drag down and give Decrease policy parameters

EC2 Management Console

Services Resource Groups

1. Configure Auto Scaling group details 2. Configure scaling policies 3. Configure Notifications 4. Configure Tags 5. Review

Create Auto Scaling Group

Scale between 1 and 3 instances. These will be the minimum and maximum size of your group.

Increase Group Size

Name: Increase Group Size

Execute policy when: awsec2-myautoscalegrp-High-CPU-Utilization [Edit](#) [Remove](#)
breaches the alarm threshold: CPUUtilization >= 30 for 300 seconds
for the metric dimensions AutoScalingGroupName = myautoscalegrp

Take the action: Add 1 instances when 30 <= CPUUtilization < +infinity

[Add step](#)

Instances need: 300 seconds to warm up after each step

[Cancel](#) [Previous](#) [Review](#) Next: Configure Notifications

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

In Decrease Group Wizard

Click on "Add new alarm"

EC2 Management Console

Services Resource Groups

1. Configure Auto Scaling group details 2. Configure scaling policies 3. Configure Notifications 4. Configure Tags 5. Review

Create Auto Scaling Group

Decrease Group Size

Name: Decrease Group Size

Execute policy when: No alarm selected [Add new alarm](#)

Take the action: Remove 0 instances

[Add step](#)

[Create a simple scaling policy](#)

[Scale the Auto Scaling group using a target tracking scaling policy](#)

[Cancel](#) [Previous](#) [Review](#) Next: Configure Notifications

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Select the topic "Cpuutilizationabc"

Whenever Average of CPU utilization is select "<="

Create Alarm



You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define.

To edit an alarm, first choose whom to notify and then define when the notification should be sent.

☒ Send a notification to: Cpuutilizationabc (skmarhaan999@gmail) [create topic](#)

Whenever: Average of CPU Utilization

Is: >= Percent

For at least: >= consecutive period(s) of 5 Minutes

Name of alarm: awsec2-myautoscalegrp-High-CPU-Utilization

CPU Utilization Percent



Cancel

Create Alarm

Give the value->20

Click on "Create Alarm" Button

Create Alarm



You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define.

To edit an alarm, first choose whom to notify and then define when the notification should be sent.

☒ Send a notification to: Cpuutilizationabc (skmarhaan999@gmail) [create topic](#)

Whenever: Average of CPU Utilization

Is: <= 20 Percent

For at least: 1 consecutive period(s) of 5 Minutes

Name of alarm: awsec2-myautoscalegrp-High-CPU-Utilization

CPU Utilization Percent



Cancel

Create Alarm

Check the summary

Click on "Next: Configure Notification"

EC2 Management Console

Services Resource Groups

1. Configure Auto Scaling group details 2. **Configure scaling policies** 3. Configure Notifications 4. Configure Tags 5. Review

Create Auto Scaling Group

Decrease Group Size

Name: Decrease Group Size

Execute policy when: awsec2-myautoscalegrp-High-CPU-Utilization [Edit](#) [Remove](#)
breaches the alarm threshold: CPUUtilization <= 20 for 300 seconds
for the metric dimensions AutoScalingGroupName = myautoscalegrp

Take the action: Remove 1 instances when 20 >= CPUUtilization > -Infinity

[Add step](#)

[Create a simple scaling policy](#)

[Cancel](#) [Previous](#) [Review](#) [Next: Configure Notifications](#)

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Click on "Add notification" button

EC2 Management Console

Services Resource Groups

1. Configure Auto Scaling group details 2. Configure scaling policies 3. **Configure Notifications** 4. Configure Tags 5. Review

Create Auto Scaling Group

Configure your Auto Scaling group to send notifications to a specified endpoint, such as an email address, whenever a specified event takes place, including: successful launch of an instance, failed instance launch, instance termination, and failed instance termination.

If you created a new topic, check your email for a confirmation message and click the included link to confirm your subscription. Notifications can only be sent to confirmed addresses.

[Add notification](#)

[Cancel](#) [Previous](#) [Review](#) [Next: Configure Tags](#)

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Check the following output

Click on "Next: Configure tags"

The screenshot shows the 'Create Auto Scaling Group' wizard in the AWS Management Console, specifically the 'Configure Notifications' step. The breadcrumb trail at the top indicates the sequence: 1. Configure Auto Scaling group details, 2. Configure scaling policies, 3. Configure Notifications (current step), 4. Configure Tags, and 5. Review. The main heading is 'Create Auto Scaling Group', followed by a description of notifications. Below this, there's a section 'Send a notification to:' with a dropdown menu showing 'Cpuutilizationabc (skmarhaan999@gmail.com)' and a 'create topic' link. Under 'Whenever instances:', four checkboxes are checked: 'launch', 'terminate', 'fail to launch', and 'fail to terminate'. An 'Add notification' button is present. At the bottom right, navigation buttons include 'Cancel', 'Previous', 'Review', and 'Next: Configure Tags', with the latter being highlighted by a mouse cursor. The footer contains 'Feedback', 'English', copyright information, and links to 'Privacy Policy' and 'Terms of Use'.

From tag key->Name

From tag value-> WebAutoscale

Click on "Review" Button

The screenshot shows the 'Create Auto Scaling Group' wizard in the AWS Management Console, specifically the 'Configure Tags' step. The breadcrumb trail at the top indicates the sequence: 1. Configure Auto Scaling group details, 2. Configure scaling policies, 3. Configure Notifications, 4. Configure Tags (current step), and 5. Review. The main heading is 'Create Auto Scaling Group', followed by a description of tags. Below this, there's a table with columns 'Key', 'Value', and 'Tag New Instances'. The first row has 'Name' in the 'Key' column and 'WebAutoscale' in the 'Value' column, with the 'Tag New Instances' checkbox checked. An 'Add tag' button is present, along with a note '49 remaining'. At the bottom right, navigation buttons include 'Cancel', 'Previous', and 'Review', with the latter being highlighted by a mouse cursor. The footer contains 'Feedback', 'English', copyright information, and links to 'Privacy Policy' and 'Terms of Use'.

Check the summary

Drag Down

EC2 Management Console

Services Resource Groups

1. Configure Auto Scaling group details 2. Configure scaling policies 3. Configure Notifications 4. Configure Tags 5. Review

Create Auto Scaling Group

Please review your Auto Scaling group details. You can go back to edit changes for each section. Click **Create Auto Scaling group** to complete the creation of an Auto Scaling group.

▼ Auto Scaling Group Details [Edit details](#)

Group name	myautoscalegrp
Group size	1
Minimum Group Size	1
Maximum Group Size	3
Subnet(s)	subnet-19d0f141, subnet-13f60e5a, subnet-8b9e38ec
Health Check Grace Period	300
Detailed Monitoring	No
Instance Protection	None

▼ Scaling Policies [Edit scaling policies](#)

[Cancel](#) [Previous](#) [Create Auto Scaling group](#)

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Drag Down

Click on "Create Auto Scaling Group" Button

EC2 Management Console

Services Resource Groups

1. Configure Auto Scaling group details 2. Configure scaling policies 3. Configure Notifications 4. Configure Tags 5. Review

Create Auto Scaling Group

Detailed Monitoring No

Instance Protection None

▼ Scaling Policies [Edit scaling policies](#)

Increase Group Size	With alarm = awsec2-myautoscalegrp-High-CPU-Utilization; Add 1 instances and 300 seconds for instances to warm up
Decrease Group Size	With alarm = awsec2-myautoscalegrp-High-CPU-Utilization; Remove 1 instances

▼ Notifications [Edit notifications](#)

Cpuutilizationabc (skmarhaan999@gmail.com)	launch, terminate, fail to launch, fail to terminate
--	--

▼ Tags [Edit tags](#)

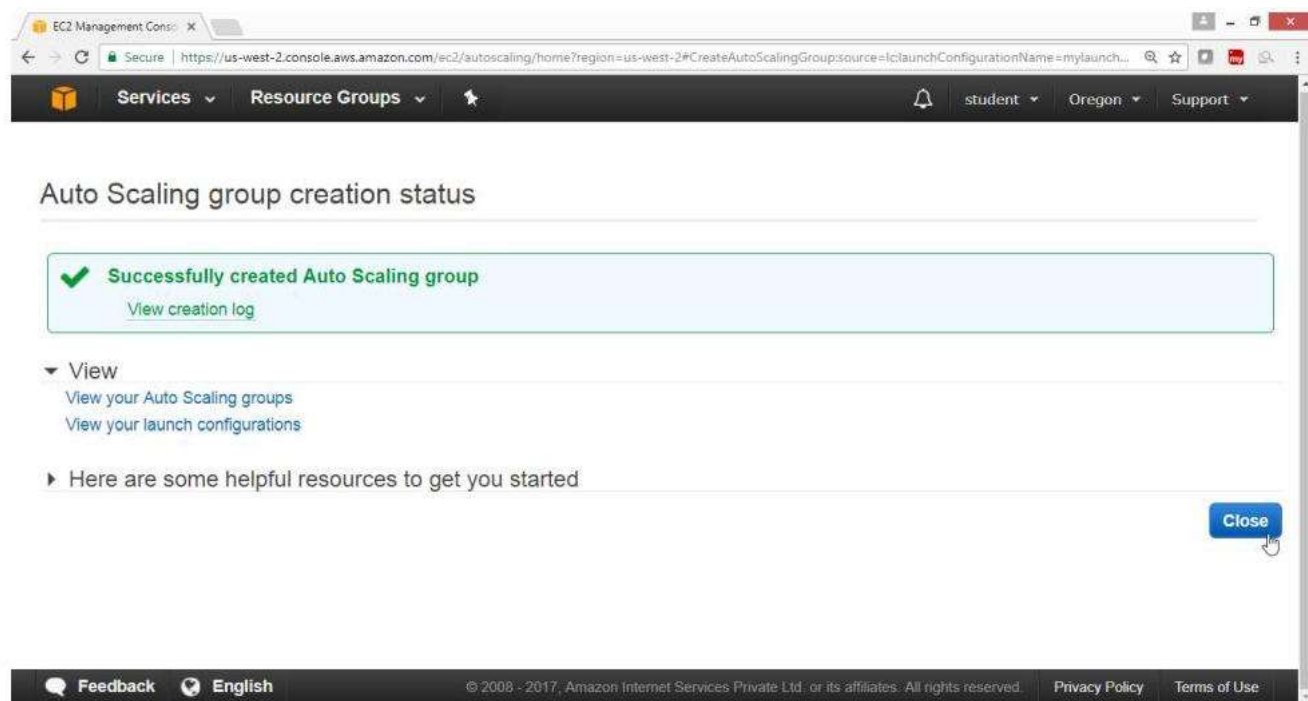
Name	WebAutoscale	tag new instances
------	--------------	-------------------

[Cancel](#) [Previous](#) [Create Auto Scaling group](#)

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

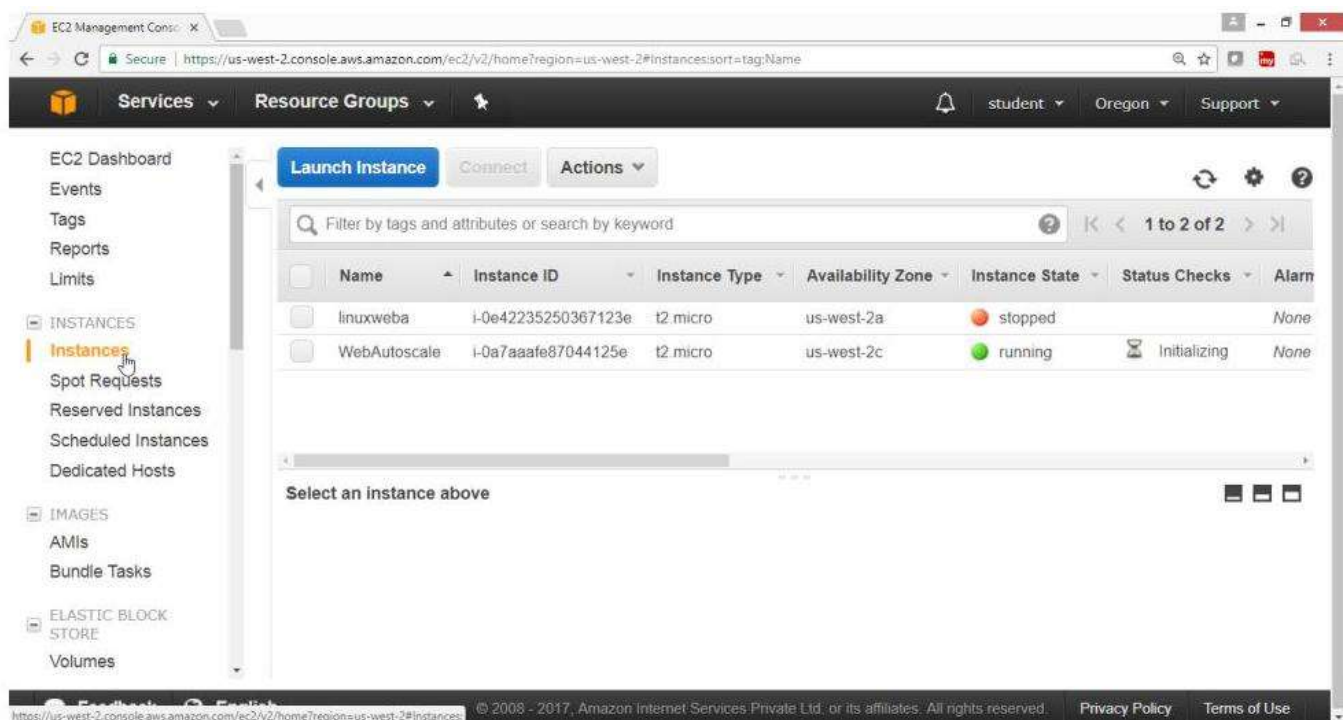
Successfully created

Click on "Close" button

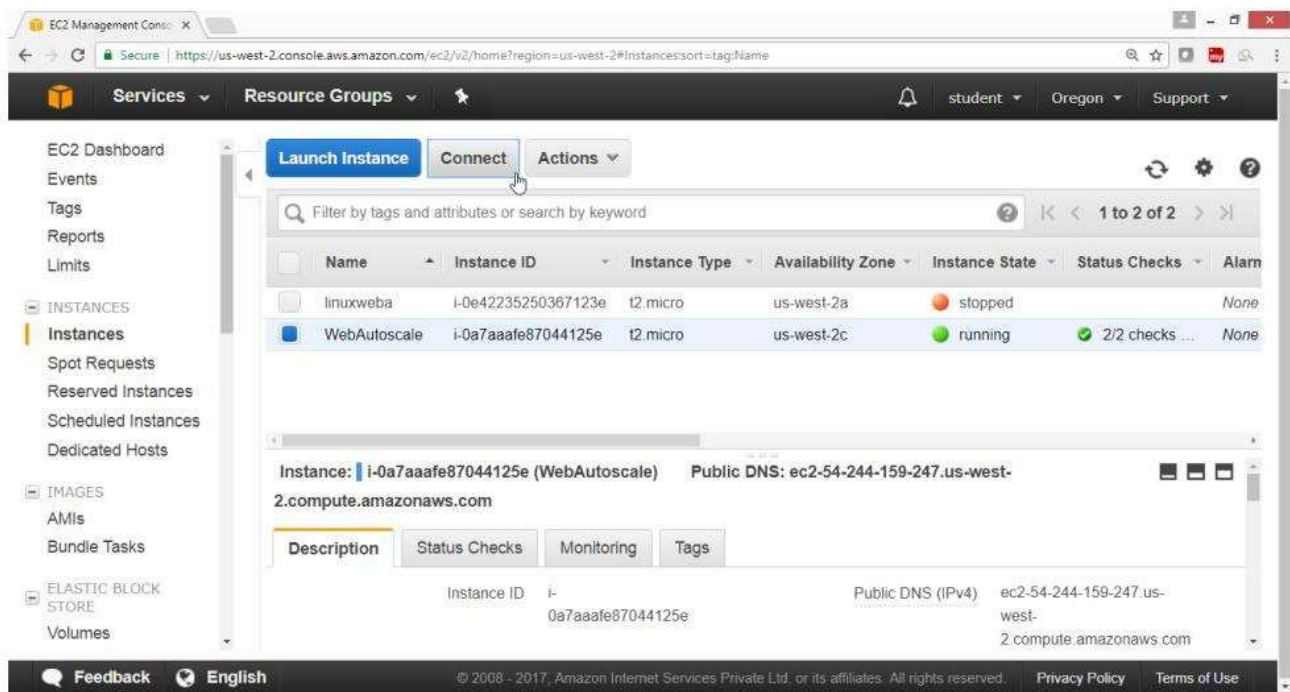


Verification

- Now go to EC2 Dash board
- Click on "Instances"
- Observer that "WebAutoscale" instance got launched



Now login to Web Autoscale instance



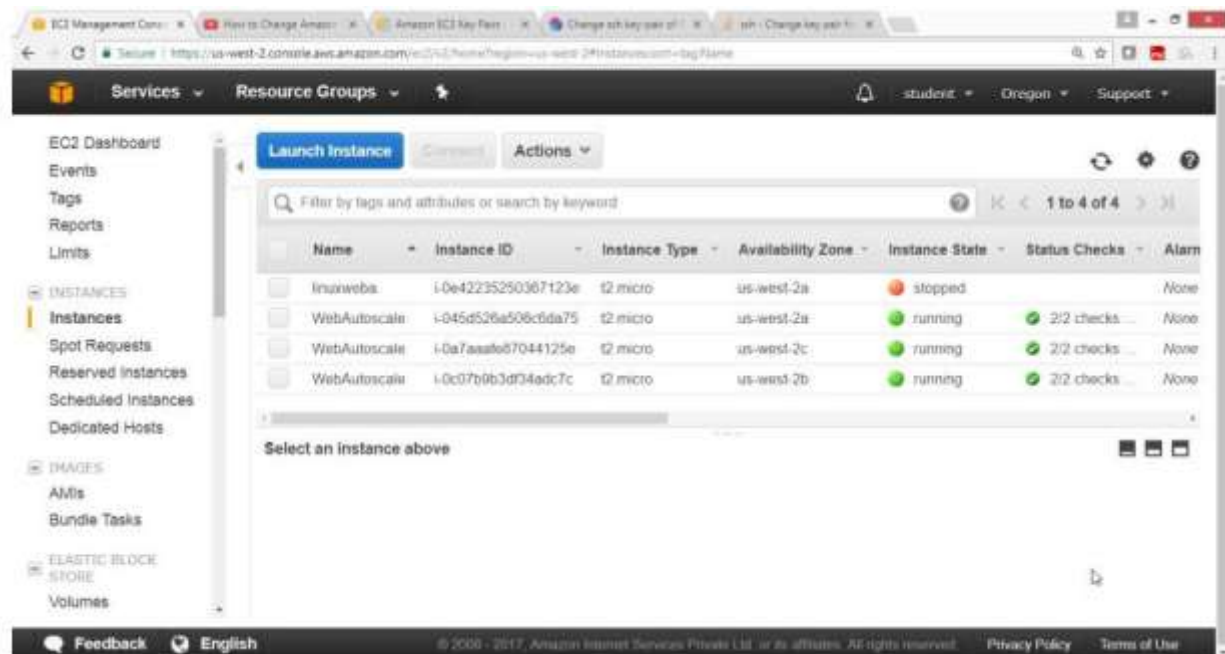
Run the following command to increase the load

```
#yum install stress
```

```
#stress --cpu --timeout 1000
```

Verification

After 15 minutes 3 instances got loaded automatically



What are lifecycle hooks used for in Autoscaling?

- A. They are used to do health checks on instances
- B. They are used to put an additional wait time to a scale in or scale out event.
- C. They are used to shorten the wait time to a scale in or scale out event
- D. None of these

Answer B

Explanation: Lifecycle hooks are used for putting wait time before any lifecycle action i.e launching or terminating an instance happens. The purpose of this wait time, can be anything from extracting log files before terminating an instance or installing the necessary software's in an instance before launching it.

A user has setup an Auto Scaling group. Due to some issue the group has failed to launch a single instance for more than 24 hours. What will happen to Auto Scaling in this condition?

- A. Auto Scaling will keep trying to launch the instance for 72 hours
- B. Auto Scaling will suspend the scaling process**
- C. Auto Scaling will start an instance in a separate region
- D. The Auto Scaling group will be terminated automatically

Answer B

Explanation: Auto Scaling allows you to suspend and then resume one or more of the Auto Scaling processes in your Auto Scaling group. This can be very useful when you want to investigate a configuration problem or other issue with your web application, and then make changes to your application, without triggering the Auto Scaling process.

AWS Elastic Bean Stack

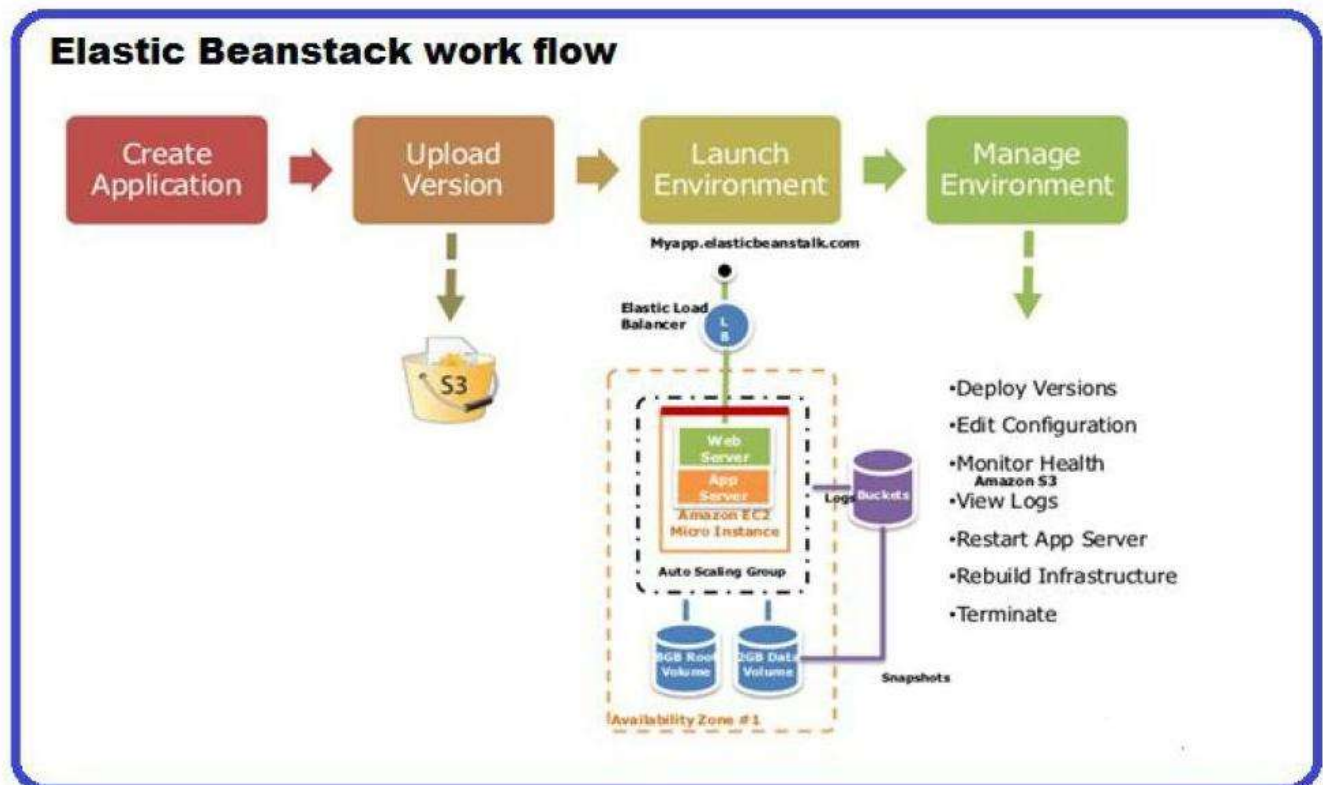
Elastic Bean Stack Highlights

Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring based on the code you upload it.

Share the Elastic Bean Stack Configuration Step by Step?

To configure Elastic Bean Stack in AWS

Topology



Pre-requisites

User should have AWS account, or IAM user with `AWSElasticBeanStalkFullAccess`

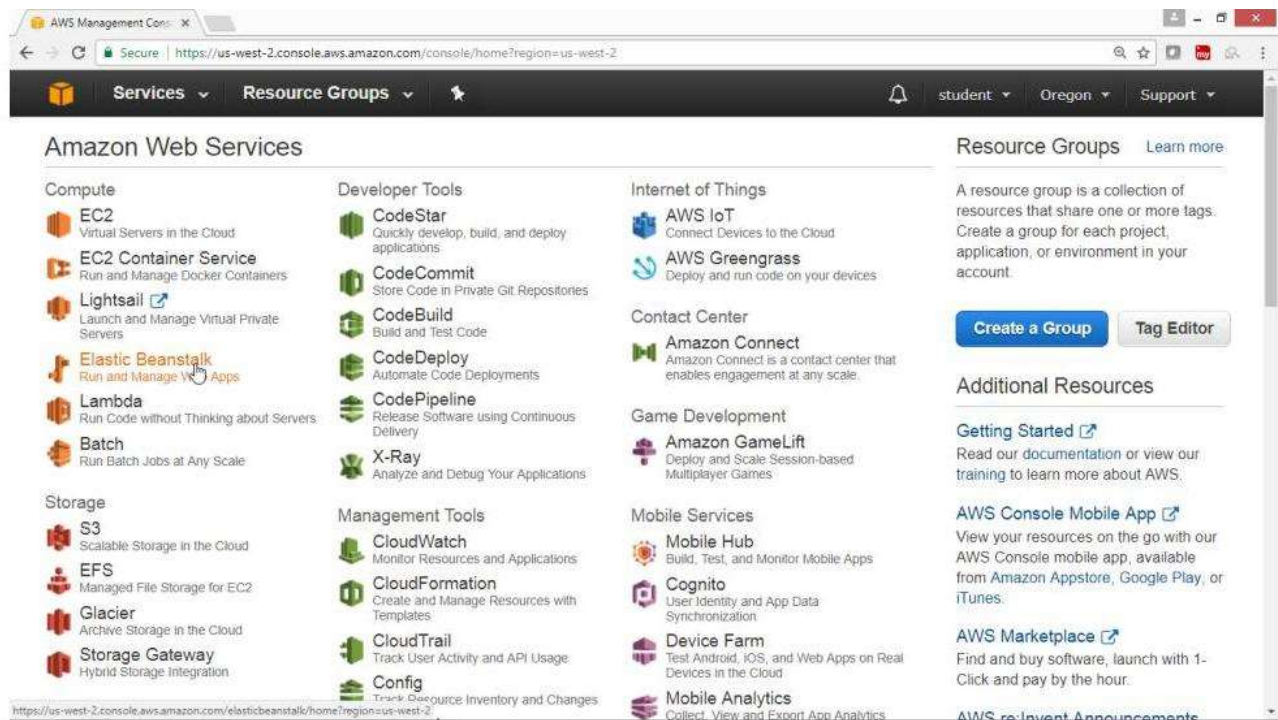
Task

- Create Elastic Beanstalk Tomcat Application
- Deploy java war files
- Open browser and check your web application

To create Elastic Beanstalk Application

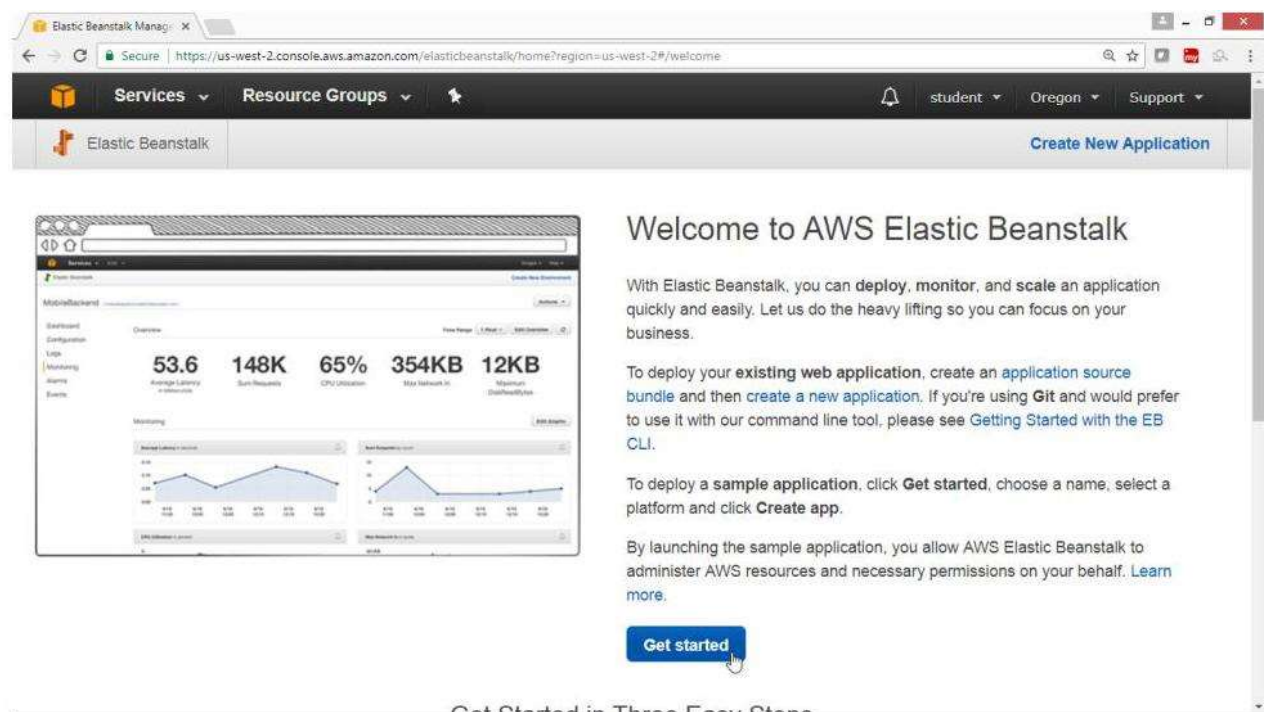
Open AWS Console, Select Compute Service

Click on "Elastic Beanstalk"



"Welcome to Amazon Elastic Beanstalk" page opens

Click on "Get Started" button



On "Create a Web App", page, provide values

Application Name -> Tomcatapp

Environment Name -> Tomcatenv

Drag Down

Elastic Beanstalk Manager

Services Resource Groups

Elastic Beanstalk Create New Application

Create a web app

Create a new application and environment with a sample application or your own code. By creating an environment, you allow AWS Elastic Beanstalk to manage AWS resources and permissions on your behalf. [Learn more](#)

Application information

Application name

Up to 100 Unicode characters, not including forward slash (/).

Environment information

Choose the name, subdomain, and description for your environment. These cannot be changed later.

Environment name

Domain

In Platform box select "Tomcat"

Drag Down

Elastic Beanstalk Manager

Base configuration

Platform -- Choose a platform --

Application code

- Choose a platform --
- Preconfigured
 - Node.js
 - PHP
 - Python
 - Ruby
 - Tomcat**
 - .NET (Windows/IIS)
 - Java
 - Go
 - Packer
- Preconfigured - Docker
 - GlassFish
 - Go
 - Python
- Generic
 - Docker
 - Multi-container Docker

options Create application

We're moving to a new design for AWS Elastic Beanstalk. [Let us know what you think!](#) You can switch back to the previous version while we finalize the design.

Feedback English

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Select Upload your code

Elastic Beanstalk Manager

Secure | https://us-west-2.console.aws.amazon.com/elasticbeanstalk/home?region=us-west-2#/gettingStarted

Base configuration

Platform

Tomcat

Choose **Configure more options** for more platform configuration options.

Application code

☐ Sample application

Get started right away with sample code.

☒ Upload your code

Upload a source bundle from your computer or copy one from Amazon S3.

Upload

ZIP or WAR

Cancel

Configure more options

Create application

We're moving to a new design for AWS Elastic Beanstalk. [Let us know what you think!](#) You can switch back to the [previous version](#) while we finalize the design.

Feedback

English

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

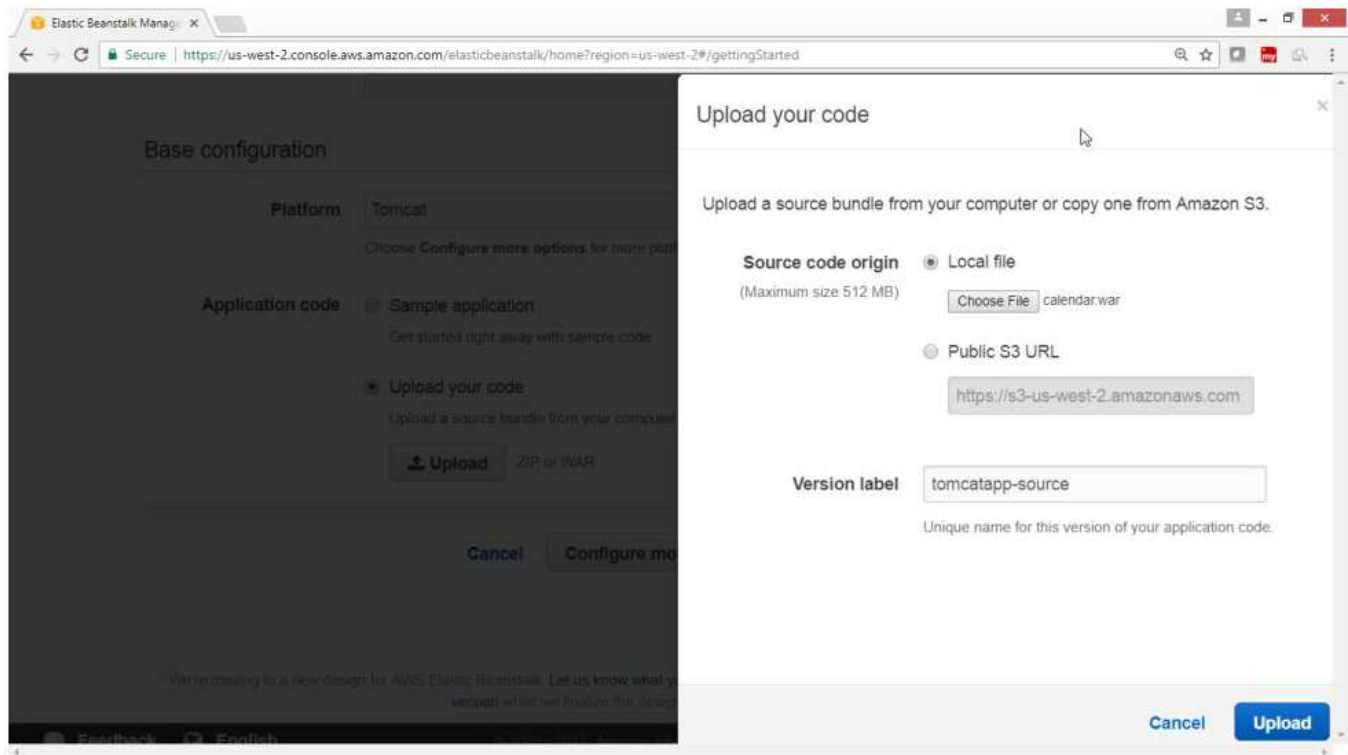
Privacy Policy

Terms of Use

Upload "calendar. War" file

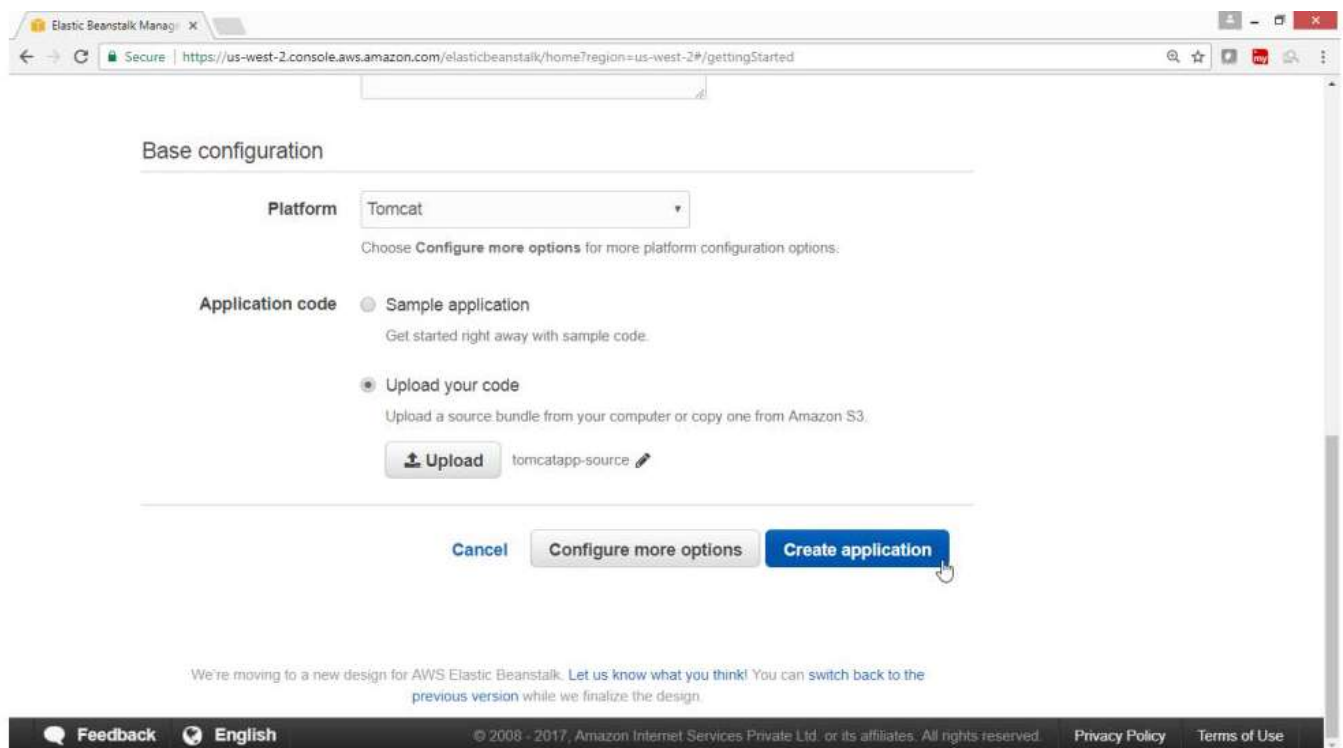
Click on "Upload" Button

Leave remaining fields as defaults



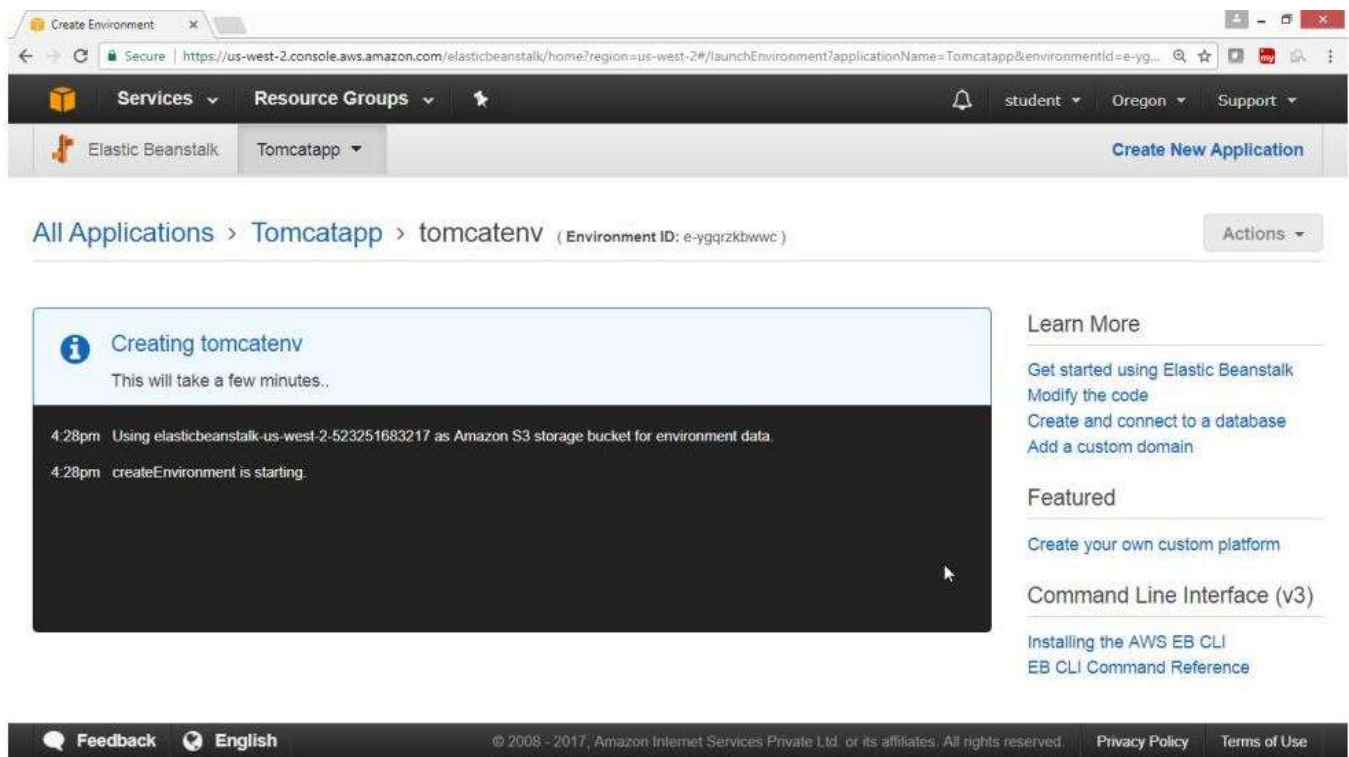
Verify that file is uploaded, beside "Upload" button

Click "Create Application" Button

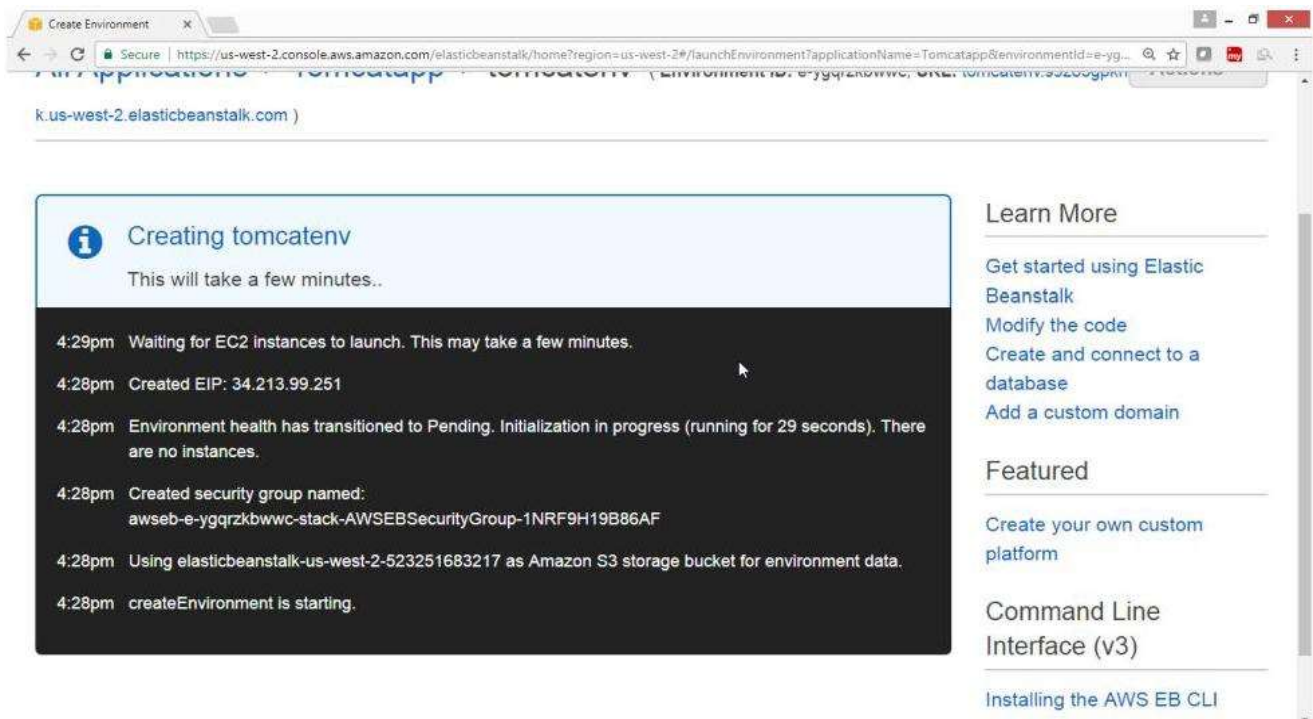


Verification:

Tomcat application at background is getting created,
Progress on screen are displayed



Verify



Note: This will few minutes to start
Wait until Tomcat Dashboard is displayed on the screen
Click on the URL link

tomcatenv - Dashboard

Services Resource Groups

Elastic Beanstalk Tomcatapp Create New Application

All Applications > Tomcatapp > tomcatenv (Environment ID: e-ygqrzkbwwc, URL: tomcatenv.s9z85gpkk.us-west-2.elasticbeanstalk.com)

Dashboard Overview Refresh

Configuration

Logs

Health Health: Ok Causes

Monitoring

Alarms

Running Version: tomcatapp-source Upload and Deploy

Configuration: 64bit Amazon Linux 2017.03 v2.6.2 running Tomcat 8 Java 8

Verification

Open any Browser, Click on URL link, Now Website is open

tomcatenv - Dashboard Welcome: Calendar Demo

tomcatenv.s9z85gpkk.us-west-2.elasticbeanstalk.com

Simple Calendar:

BirthDate: 10/7/1975

Supports Multiple Calendars:

StartDate: 10/7/1975 EndDate: 10/07/2000

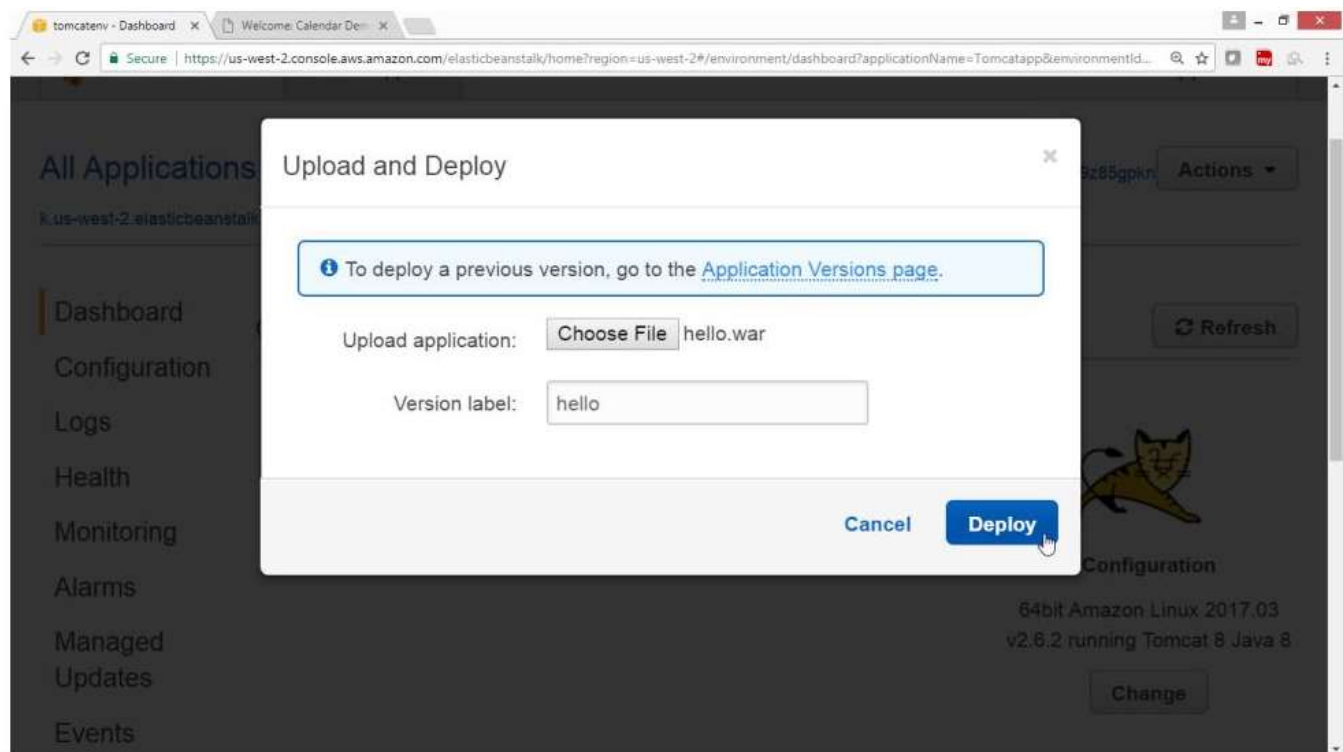
Supports indexed properties:

StartDate: 10/7/1975	EndDate: 10/07/2000
StartDate: 10/7/1975	EndDate: 10/07/2000
StartDate: 10/7/1975	EndDate: 10/07/2000
StartDate: 10/7/1975	EndDate: 10/07/2000
StartDate: 10/7/1975	EndDate: 10/07/2000

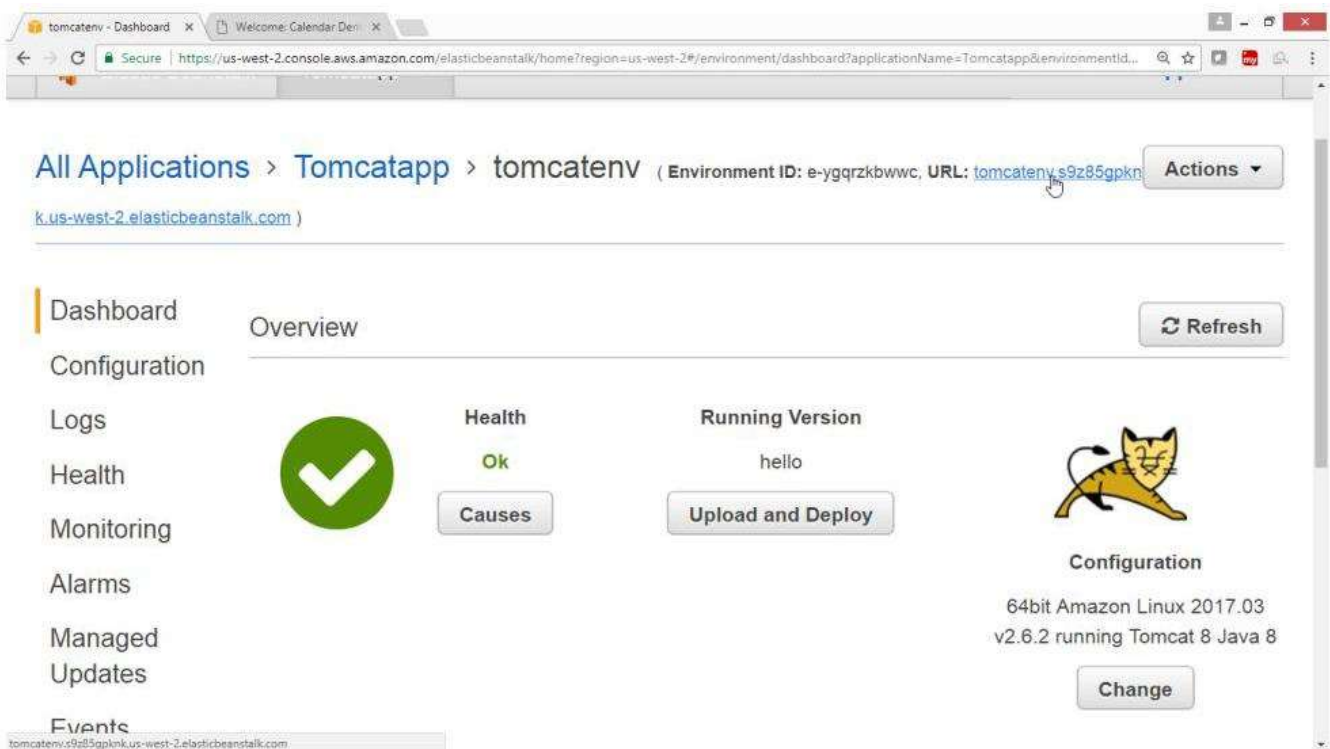
To Deploy another war file for eb hello.war

Go to Upload application, choose file provide hello.war file name

Click "Deploy" button



Click on URL



View the Website



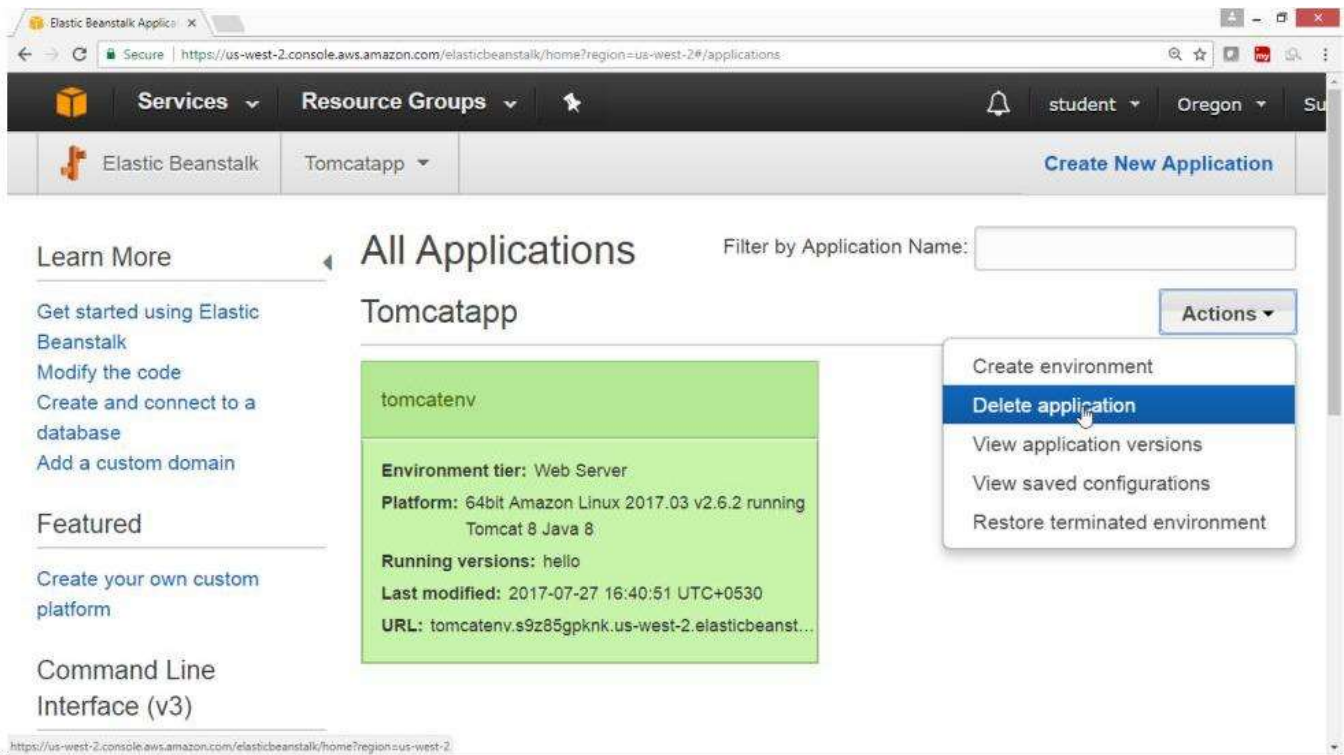
Hello Index

Try the [servlet](#).

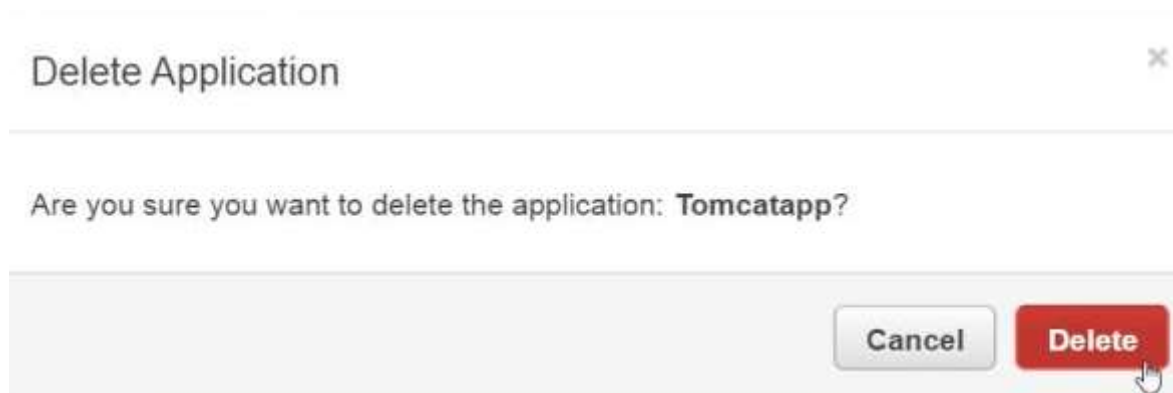
To remove Elastic Bean Stalk

Select "Action" Button

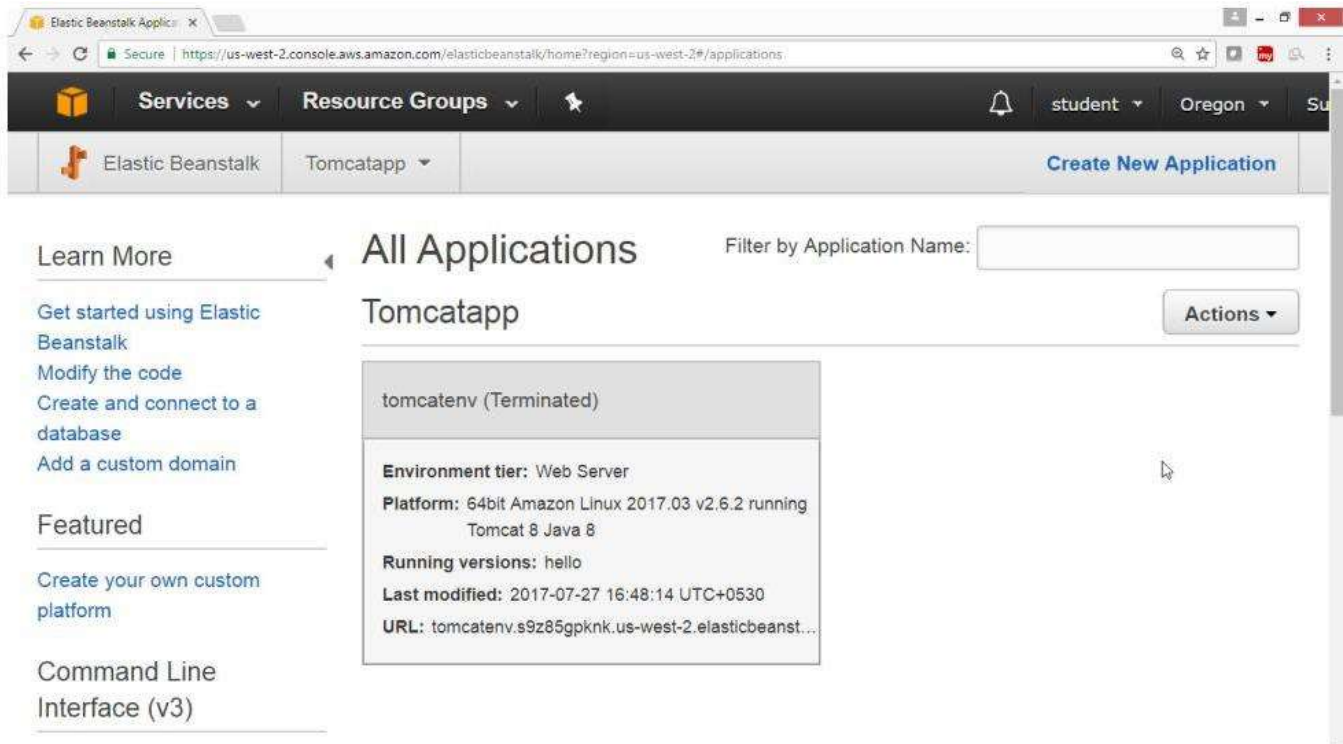
Click "Delete application" Button



Confirm "Delete"

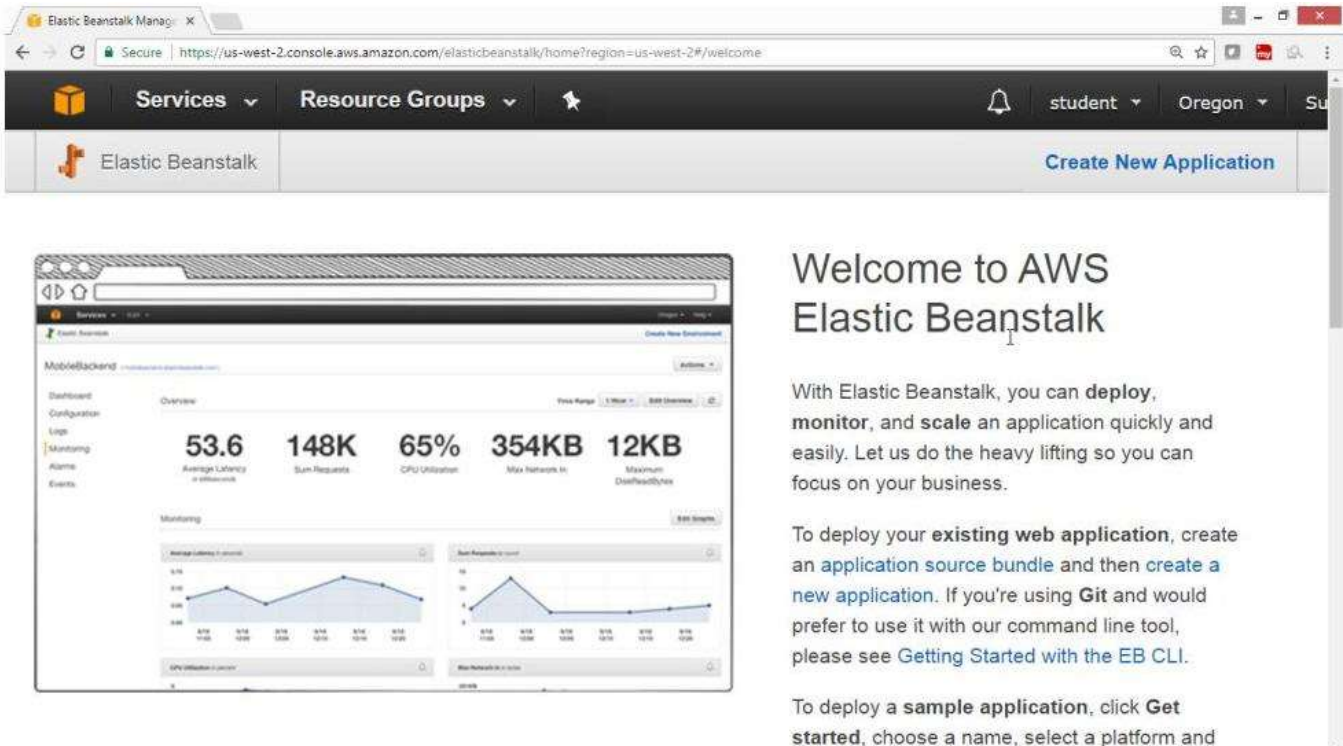


Application will now get terminated



Verification

After termination following screen will come



To delete Elastic Beanstalk bucket policy is created in S3 bucket

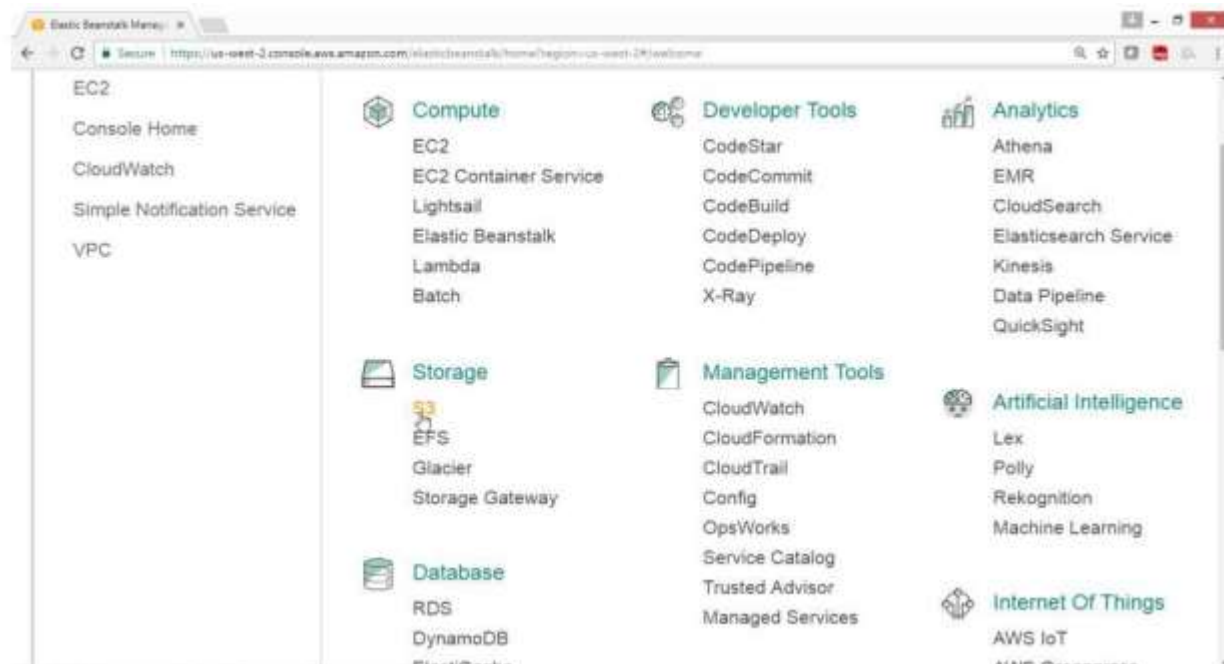
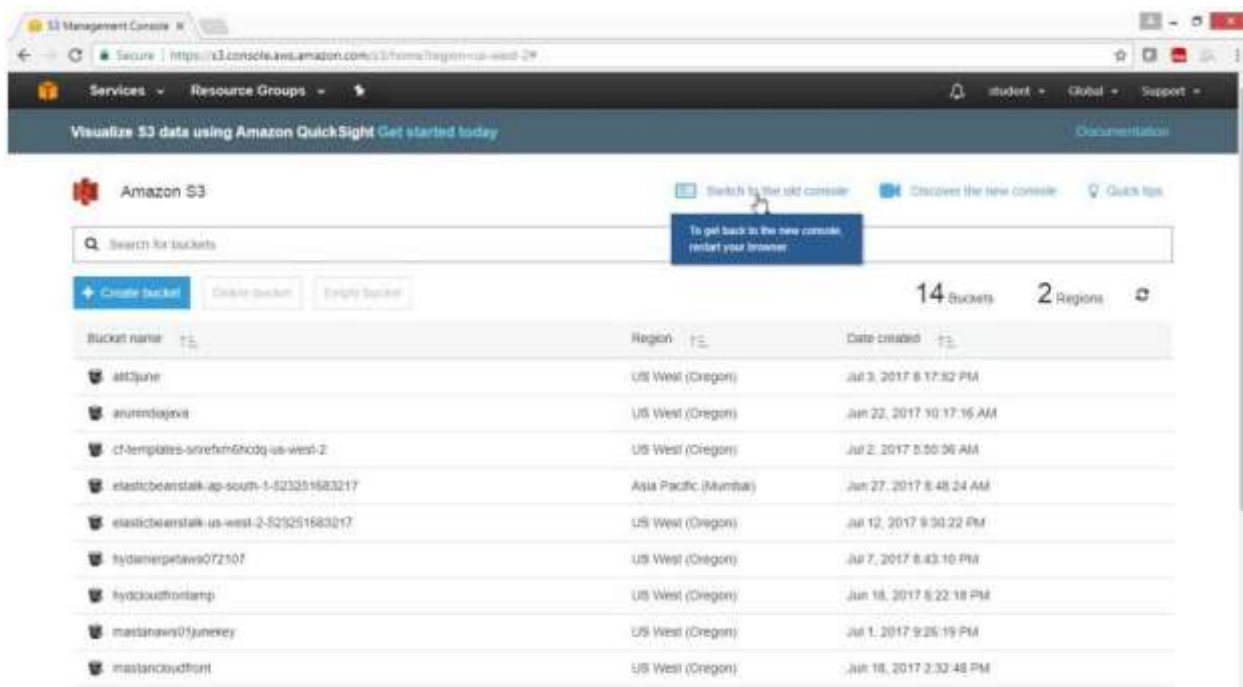
Note: S3 bucket created by Elastic Beanstalk is not deleted automatically

It could be charged after free usage limits are over, so manually delete the beanstalk bucket

From console select "Storage"

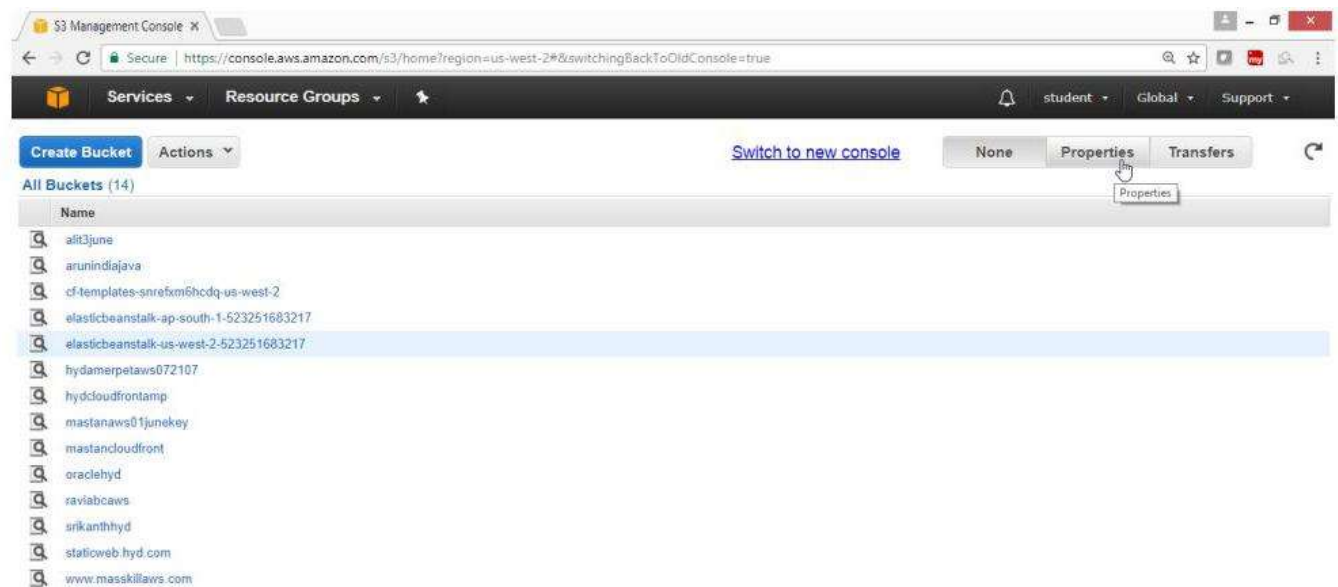
Select S3

Click on "Switch to old console"

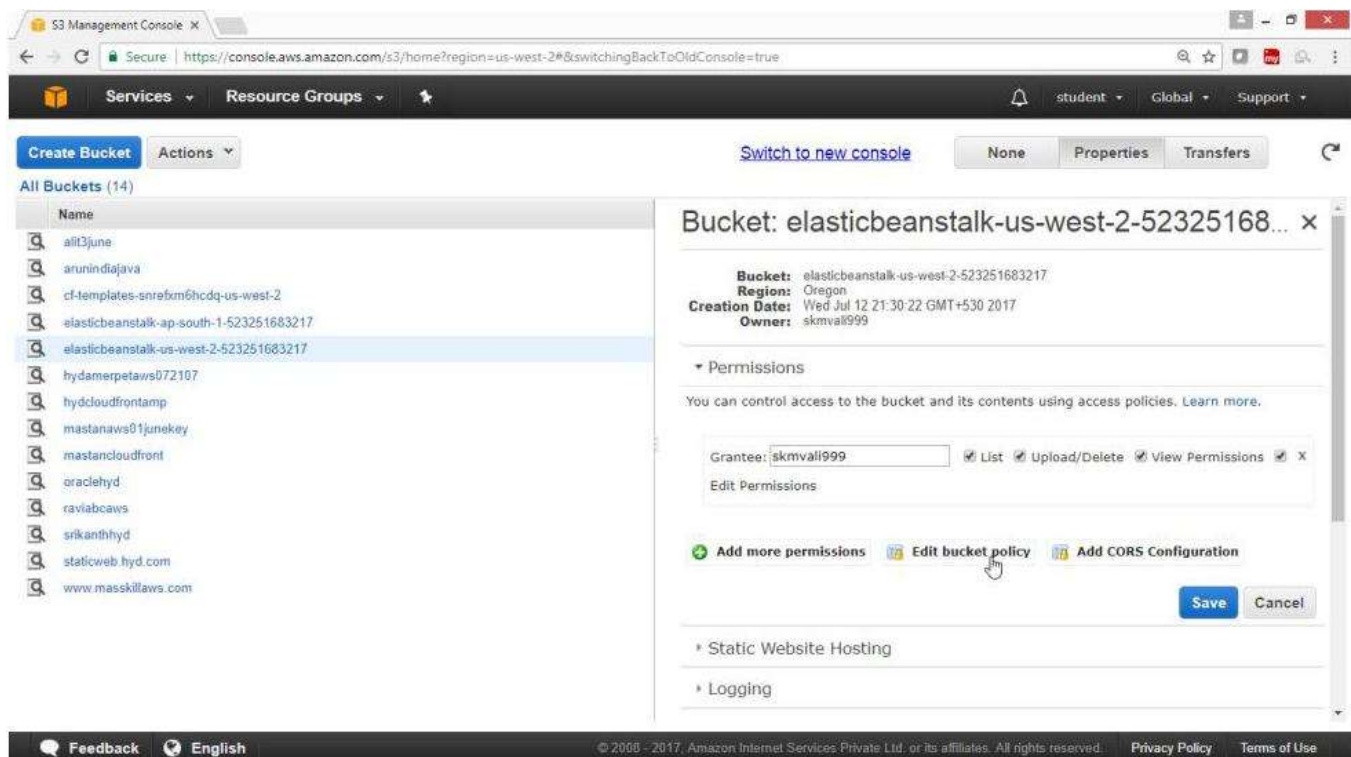


Select Elastic Beanstalk Bucket, click properties

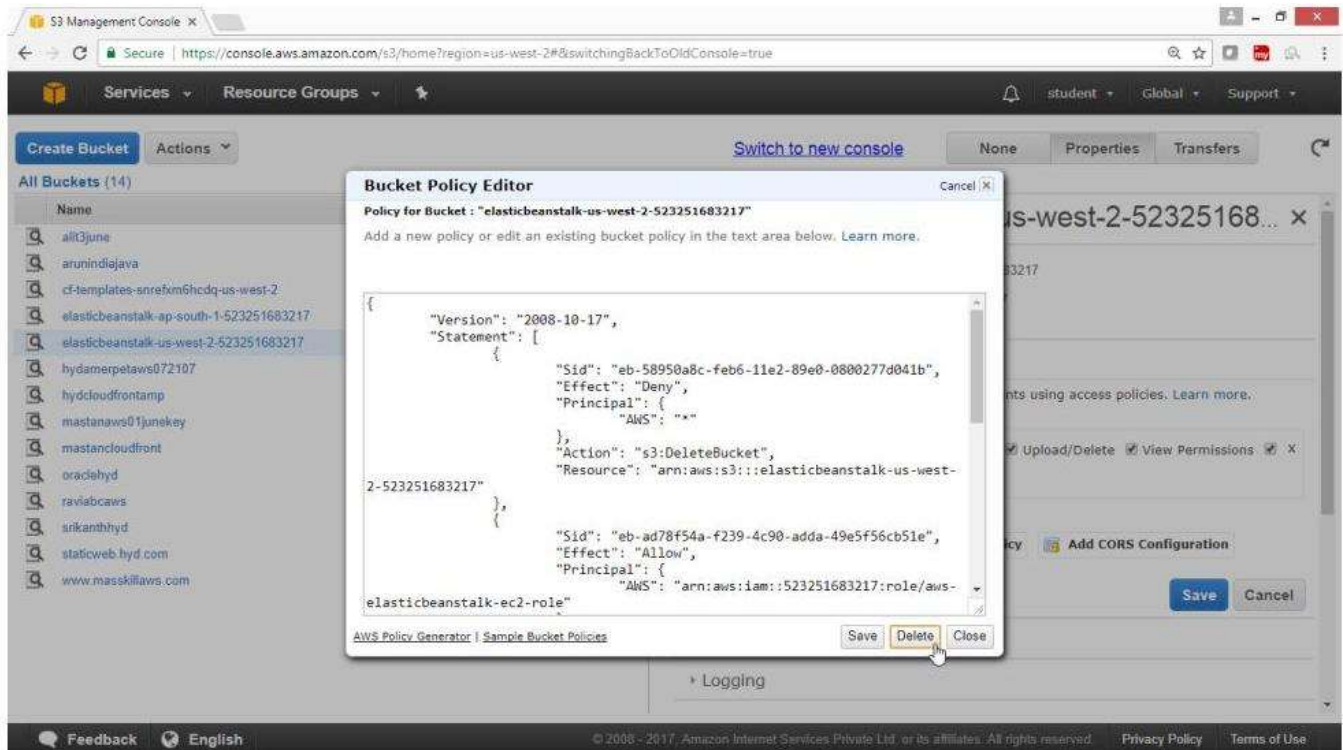
Select Permissions



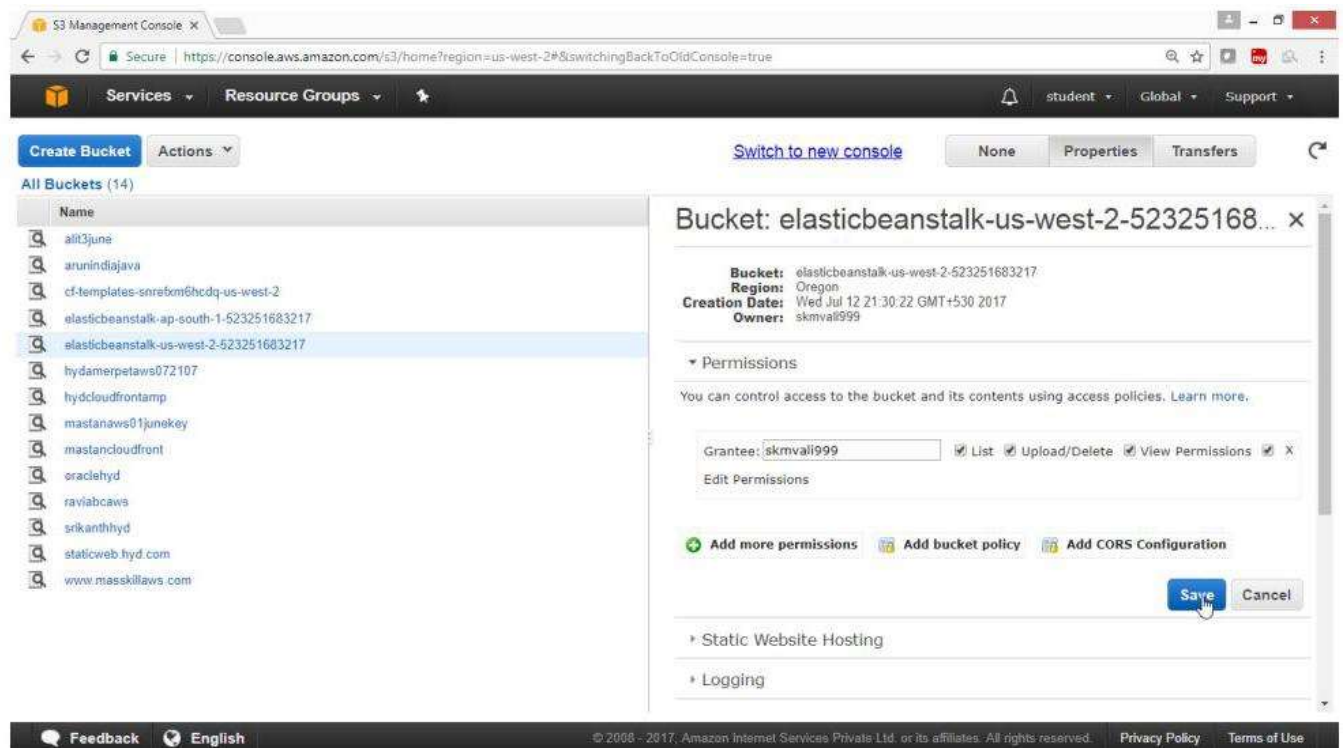
Click "Edit bucket policy"



In Bucket Policy Editor Wizard, Click Delete to remove policy, Click Ok



Click on Save Button



What is the difference between Elastic Beanstalk and CloudFormation?

Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring based on the code you upload it.

Cloud Formation is an automated provisioning engine to deploy entire cloud environments via JSON.

How is AWS Elastic Beanstalk different than AWS OpsWorks?

AWS Elastic Beanstalk is an application management platform while OpsWorks is a configuration management platform.

Beanstalk is an easy to use service which is used for deploying and scaling web applications developed with Java, .Net, PHP, Node.js, Python, Ruby, Go and Docker. Customers upload their code and Elastic Beanstalk automatically handles the deployment. The application will be ready to use without any infrastructure or resource configuration.

In contrast, AWS Opsworks is an integrated configuration management platform for IT administrators or DevOps engineers who want a high degree of customization and control over operations.

What happens if my application stops responding to requests in beanstalk?

AWS Beanstalk applications have a system in place for avoiding failures in the underlying infrastructure. If an Amazon EC2 instance fails for any reason, Beanstalk will use Auto Scaling to automatically launch a new instance. Beanstalk can also detect if your application is not responding on the custom link, even though the infrastructure appears healthy, it will be logged as an environmental event (e.g a bad version was deployed) so you can take an appropriate action.

How does Elastic Beanstalk apply updates?

- A. By having a duplicate ready with updates before swapping.
- B. By updating on the instance while it is running
- C. By taking the instance down in the maintenance window
- D. Updates should be installed manually

Answer A

Explanation: Elastic Beanstalk prepares a duplicate copy of the instance, before updating the original instance, and routes your traffic to the duplicate instance, so that, incase your updated application fails, it will switch back to the original instance, and there will be no downtime experienced by the users who are using your application.

AWS Lambda

What is AWS Lambda?

AWS Lambda is a service from Amazon to run a specific piece of code in Amazon cloud, without provisioning any server. So, there is no effort involved in administration of servers. In AWS Lambda, we are not charged until our code starts running. Therefore, it is very cost-effective solution to run code.

AWS Lambda can automatically scale our application when the number of requests to run the code increases. So, we do not have to worry about scalability of application to use AWS Lambda.

AWS Lambda is a compute service where you can upload code and create Lambda function. AWS Lambda takes care of provisioning and managing the server that you use to run the code. You don't have to worry about Operating System, Patching, Scaling, etc.,

You can use Lambda in the following ways: -

- As an event-driven compute service where AWS Lambda runs your code in response to events. These events could be changes to data in an Amazon S3 bucket or an Amazon DynamoDB table.
- As a compute service to run your code in response to HTTP requests using Amazon API Gateway or API calls made using AWS SDKs.

What is a Serverless application in AWS?

In AWS, we can create applications based on AWS Lambda. These applications are composed of functions that are triggered by an event.

These functions are executed by AWS in cloud. But we do not have to specify/buy any instances or server for running these functions. An application created on AWS Lambda is called Serverless application in AWS.

How will you manage and run a serverless application in AWS?

We can use AWS Serverless Application Model (AWS SAM) to deploy and run a serverless application. AWS SAM is not a server or software.

It is just a specification that has to be followed for creating a serverless application. Once we create our serverless application, we can use CodePipeline to release and deploy it in AWS. CodePipeline is built on Continuous Integration Continuous Deployment (CI/CD) concept.

What are the main use cases for AWS Lambda?

Some of the main use cases in which AWS Lambda can be used are as follows: -

Web Application: We can integrate AWS Lambda with other AWS Services to create a web application that can scale up or down with zero administrative effort for server management, backup or scalability. Internet of Things (IoT): In the Internet of Things applications, we can use AWS Lambda to execute a piece of code on the basis of an event that is triggered by a device.

Mobile Backend: We can create Backend applications for Mobile apps by using AWS Lambda. Real-time Stream Processing: We can use AWS Lambda with Amazon Kinesis for processing real-time streaming data.

ETL: We can use AWS Lambda for Extract, Transform, and Load (ETL) operations in data warehousing applications. AWS Lambda can execute the code that can validate data, filter information, sort data or transform data from one form to another form.

Real-time File processing: AWS Lambda can also be used for handling any updates to a file in Amazon S3. When we upload a file to S3, AWS Lambda can create thumbnails, index files, new formats etc in real-time.

How does AWS Lambda handle failure during event processing?

In AWS Lambda we can run a function in synchronous or asynchronous mode. In synchronous mode, if AWS Lambda function fails, then it will just give an exception to the calling application. In asynchronous mode, if AWS Lambda function fails then it will retry the same function at least 3 times.

If AWS Lambda is running in response to an event in Amazon DynamoDB or Amazon Kinesis, then the event will be retried till the Lambda function succeeds or the data expires. In DynamoDB or Kinesis, AWS maintains data for at least 24 hours.

What is Lambda@Edge in AWS?

In AWS, we can use Lambda@Edge utility to solve the problem of low network latency for end users. In Lambda@Edge there is no need to provision or manage servers. We can just upload our Node.js code to AWS Lambda and create functions that will be triggered on CloudFront requests. When a request for content is received by CloudFront edge location, the Lambda code is ready to execute. This is a very good option for scaling up the operations in CloudFront without managing servers.

Which of the following services you would not use to deploy an app?

- A. Elastic Beanstalk
- B. Lambda**
- C. Opsworks
- D. CloudFormation

Answer B

Explanation: Lambda is used for running server-less applications. It can be used to deploy functions triggered by events. When we say serverless, we mean without you worrying about the computing resources running in the background. It is not designed for creating applications which are publicly accessed.



Storage

Amazon S3 Scalable Storage in the Cloud	Amazon EBS Block Storage for EC2	AWS Elastic File System Managed File Storage for EC2
Amazon Glacier Low-cost Achieve Storage in the cloud	AWS Storage Gateway Hybrid Storage Integration	Amazon Snowball Petabyte-Scale Data Transport
AWS Snowball Edge Petabyte-scale Data Transport with On-Demand Compute	AWS Snowmobile Exabyte-scale Data Transport	



Storage

Amazon S3

S3 Highlights

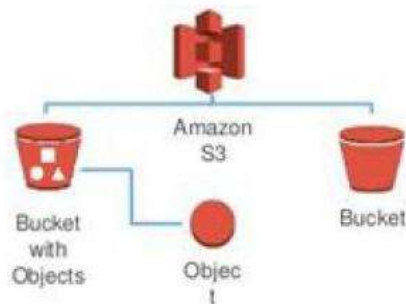
- **S3** is object- based storage, it allows you to upload files
- Files can be from 0 bytes to 5 TB
- There is an unlimited storage
- **Object** consists of raw object data and metadata
- Objects are stored and retrieved using a developer-assigned key
- Data are kept secured from unauthorized access through authentication mechanism
- Object can be made available to public by the http or bit torrent protocol
- All Files| objects are stored in Buckets
- A **bucket** is simply a container for objects. It is used to partition the namespace of objects at the highest level
- Buckets are similar to Internet domain names
- Buckets are accessed via `bucketname.s3.amazonaws.com`
- Each developer account has a limit of 100 buckets
- A key is the unique identifier for an object within a bucket
- A bucket and a key together uniquely identify each object in S3. Every object can be addressed through bucket and key combination
- For example, if your bucket name is mybucket and key is myhomepage.html, the URL for the object will be <https://mybucket.s3.amazonaws.com/myhomepage.html>
- Write to **S3** - **HTTP 200 code for a successful write**
- You can load files to S3 much faster by enabling multipart upload
- **S3** is a universal namespace, that is name must be unique globally
<https://s3-eu-west-1.amazonaws.com/google>

Share the S3 Configuration Step by Step?

To Configure and use AWS S3 Service

[Topology](#)

AWS Storage | Amazon S3 Storage Concepts



Amazon S3 Concepts

- Amazon S3 stores data as objects within **buckets**
- An **object** is comprised of a file and optionally any metadata that describes that file
- You can have **up to 100 buckets** in each account
- You can **control access** to the bucket and its objects

Pre-requisites

User should have AWS account, IAM user with AmazonS3FullAccess Policy

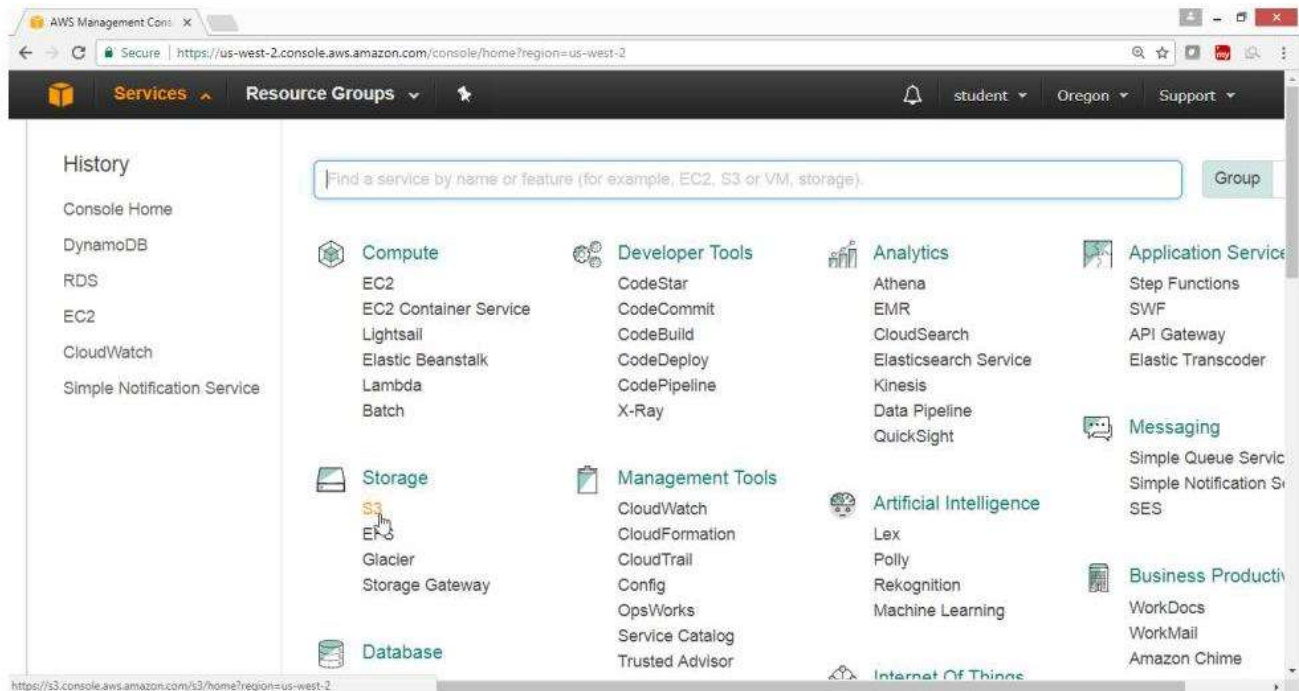
To Configure S3 with following task

- Sign up for Amazon S3
- Create a Bucket
- Add an object to a Bucket
- Add a folder to Bucket
- View an Object
- Move an Object
- Delete an Object and Bucket
- To empty a Bucket
- To delete a bucket
- Hosting a Static Website on Amazon S3
- AWS user to control S3

To create S3 bucket for storing objects that is files and folders

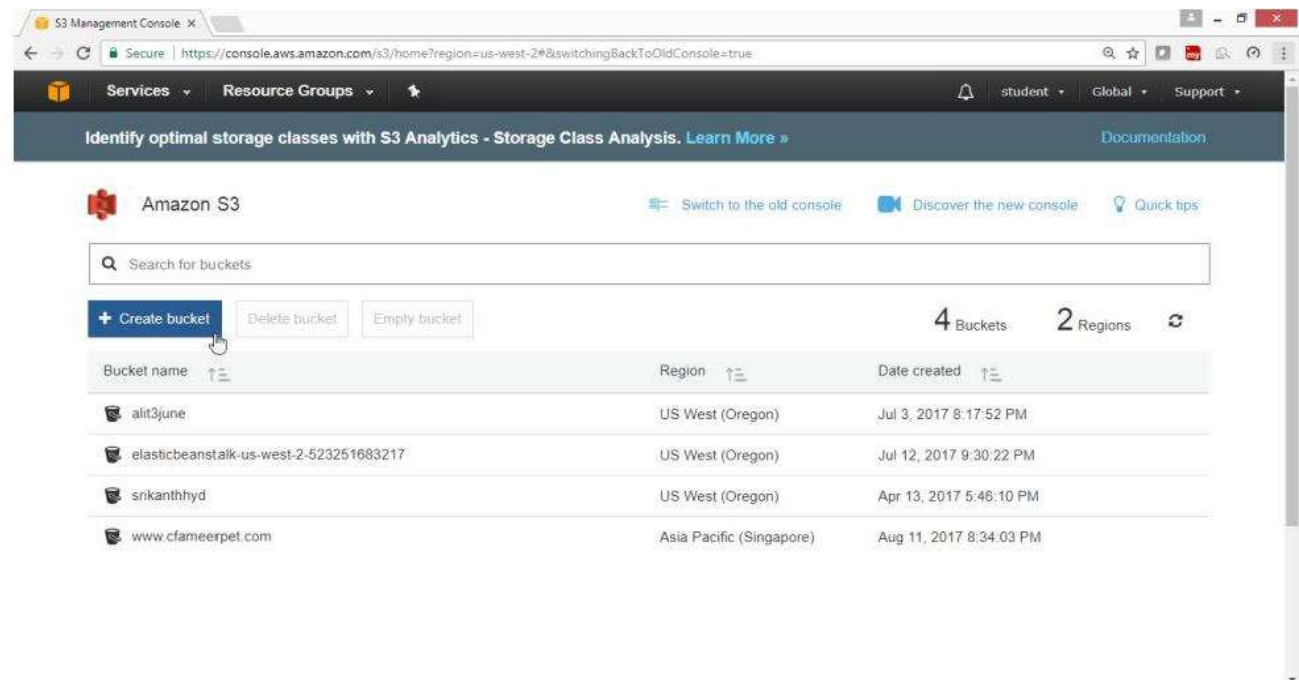
Open AWS Console

- Select **"Storage"** service
- Click on **"S3"**



On Amazon S3 page

Click on **Create Bucket**

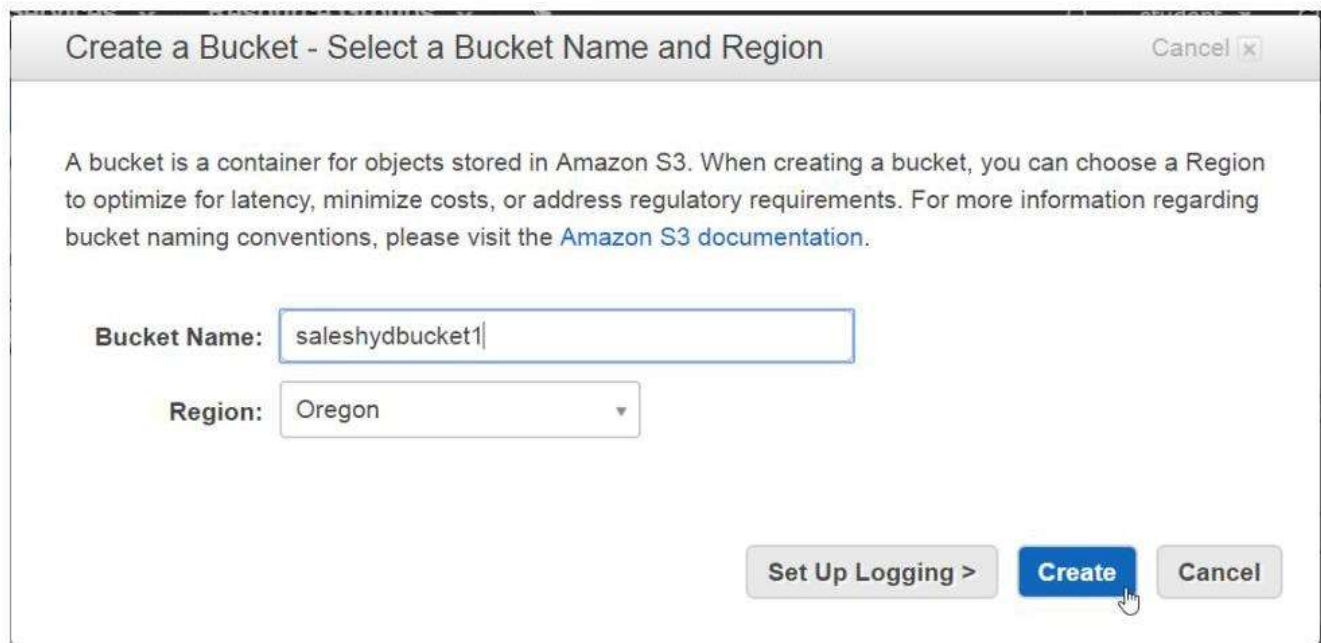


On "Create Bucket - Select a Bucket Name and Region" box

Provide following values

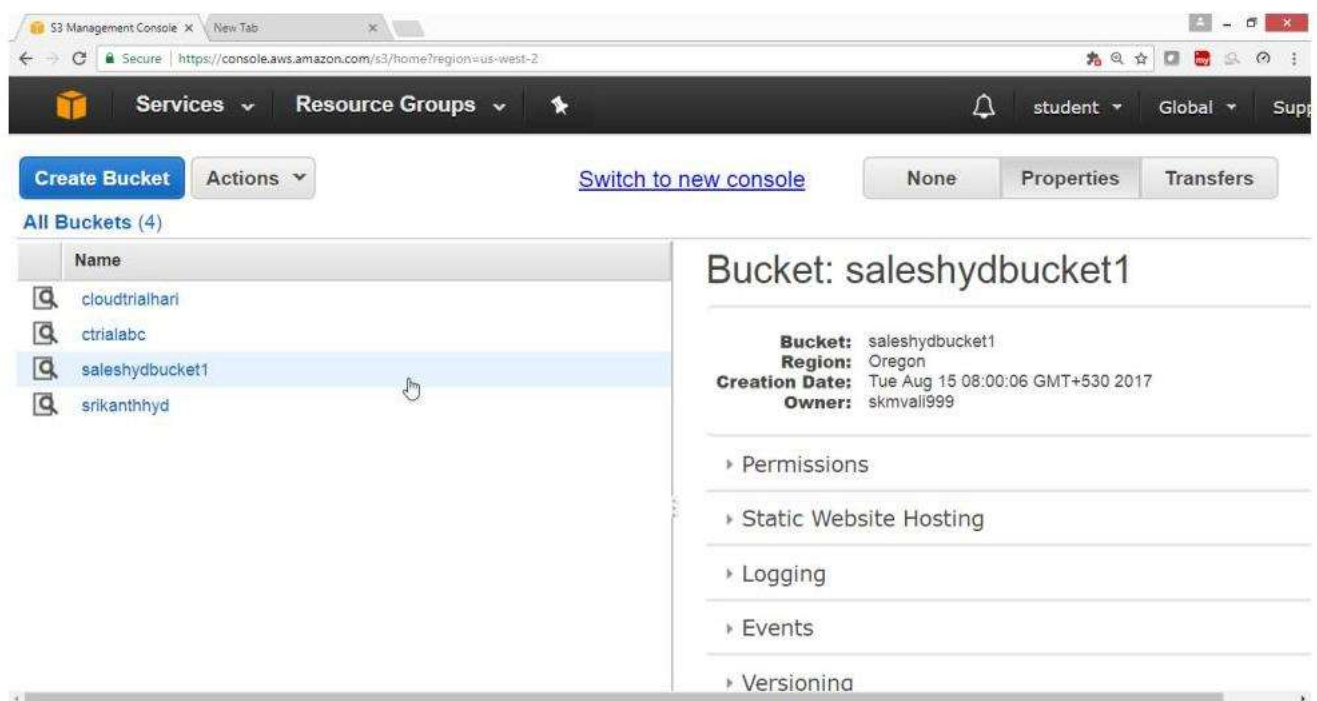
- Bucket Name -> saleshydbucket
- Region->Oregon

Note: A bucket name in region must contain only lower-case characters and should be unique in entire Amazon bucket names from all the region



The screenshot shows the 'Create a Bucket' dialog box. The title bar reads 'Create a Bucket - Select a Bucket Name and Region'. Below the title, there is explanatory text: 'A bucket is a container for objects stored in Amazon S3. When creating a bucket, you can choose a Region to optimize for latency, minimize costs, or address regulatory requirements. For more information regarding bucket naming conventions, please visit the [Amazon S3 documentation](#).' The 'Bucket Name' field contains 'saleshydbucket1'. The 'Region' dropdown menu is set to 'Oregon'. At the bottom right, there are three buttons: 'Set Up Logging >', 'Create' (highlighted with a mouse cursor), and 'Cancel'.

Verify that bucket is created



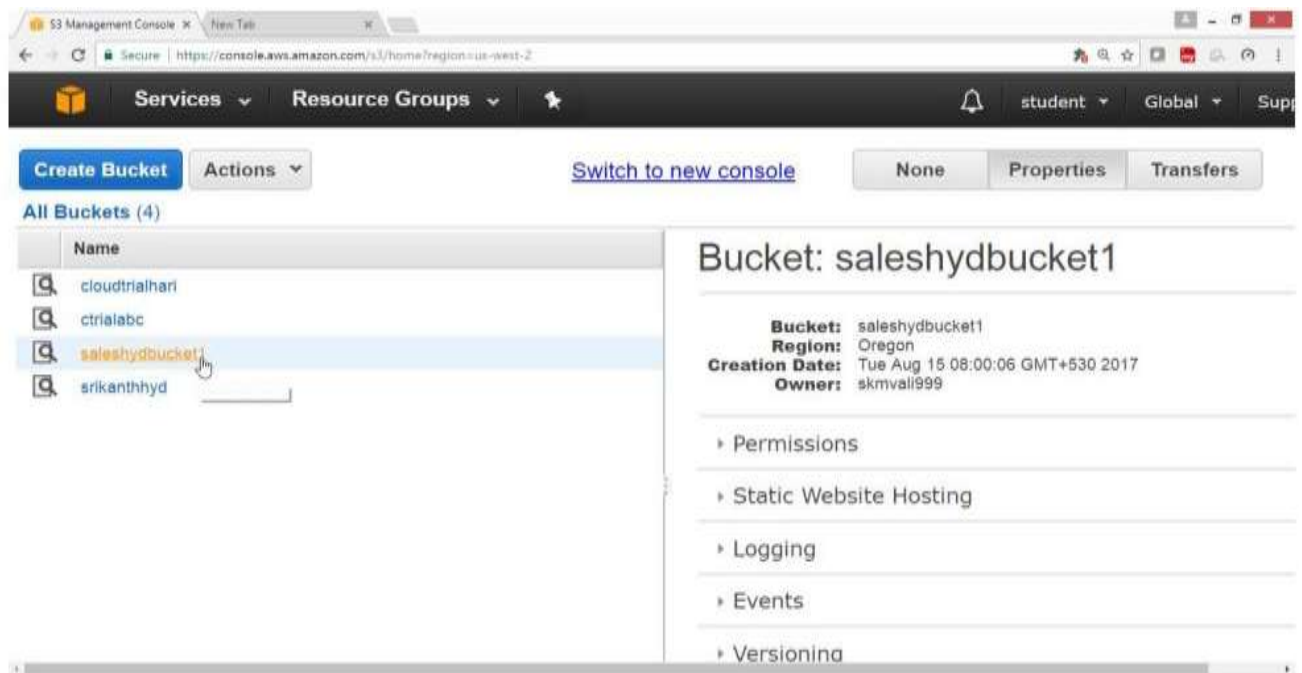
The screenshot shows the AWS S3 Management Console. The top navigation bar includes 'Services', 'Resource Groups', and user information. The main content area has a 'Create Bucket' button and a list of 'All Buckets (4)'. The buckets listed are 'cloudtrialhari', 'ctrialabc', 'saleshydbucket1' (highlighted), and 'srikanthhyd'. To the right of the list, a detailed view for 'Bucket: saleshydbucket1' is shown. It includes the following information: 'Bucket: saleshydbucket1', 'Region: Oregon', 'Creation Date: Tue Aug 15 08:00:06 GMT+530 2017', and 'Owner: skmvali999'. Below this information are expandable sections for 'Permissions', 'Static Website Hosting', 'Logging', 'Events', and 'Versioning'.

To Upload files of any types

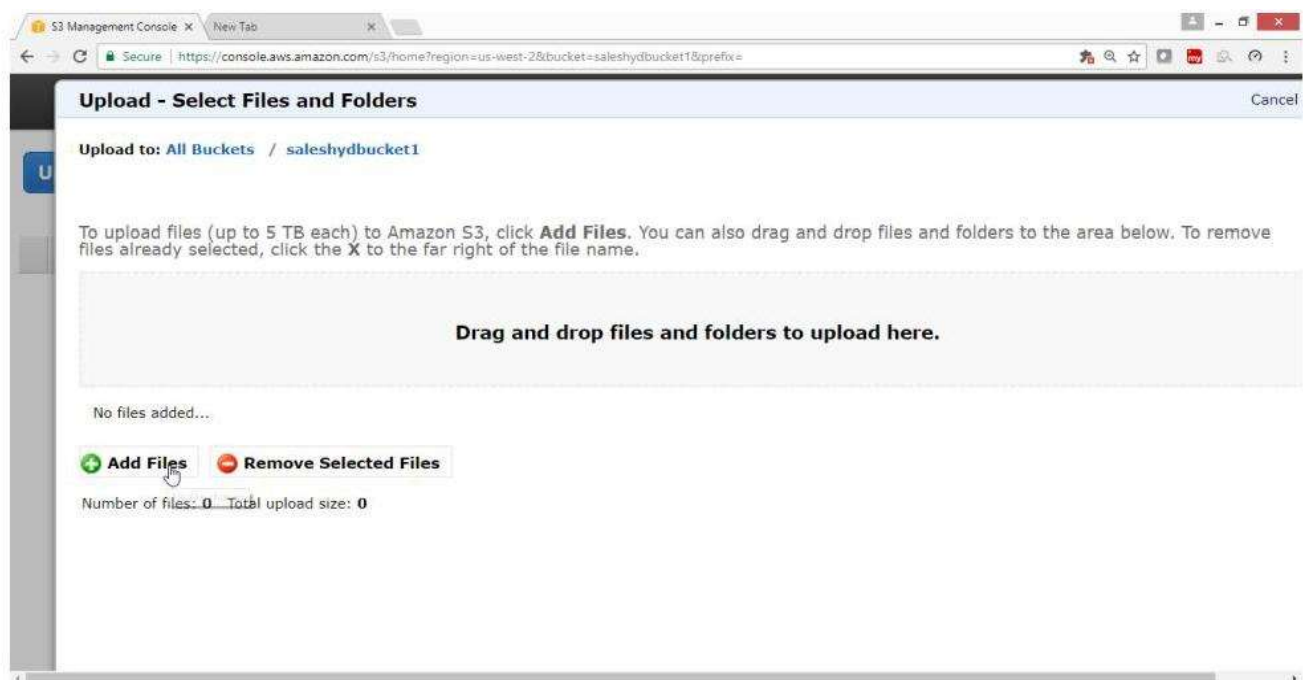
Right click in empty space, select "Upload"

Note: 5 GB can be uploaded, It, will be charged if crossed free tier usage

Click on [Created Bucket](#)



Click on "Add Files"

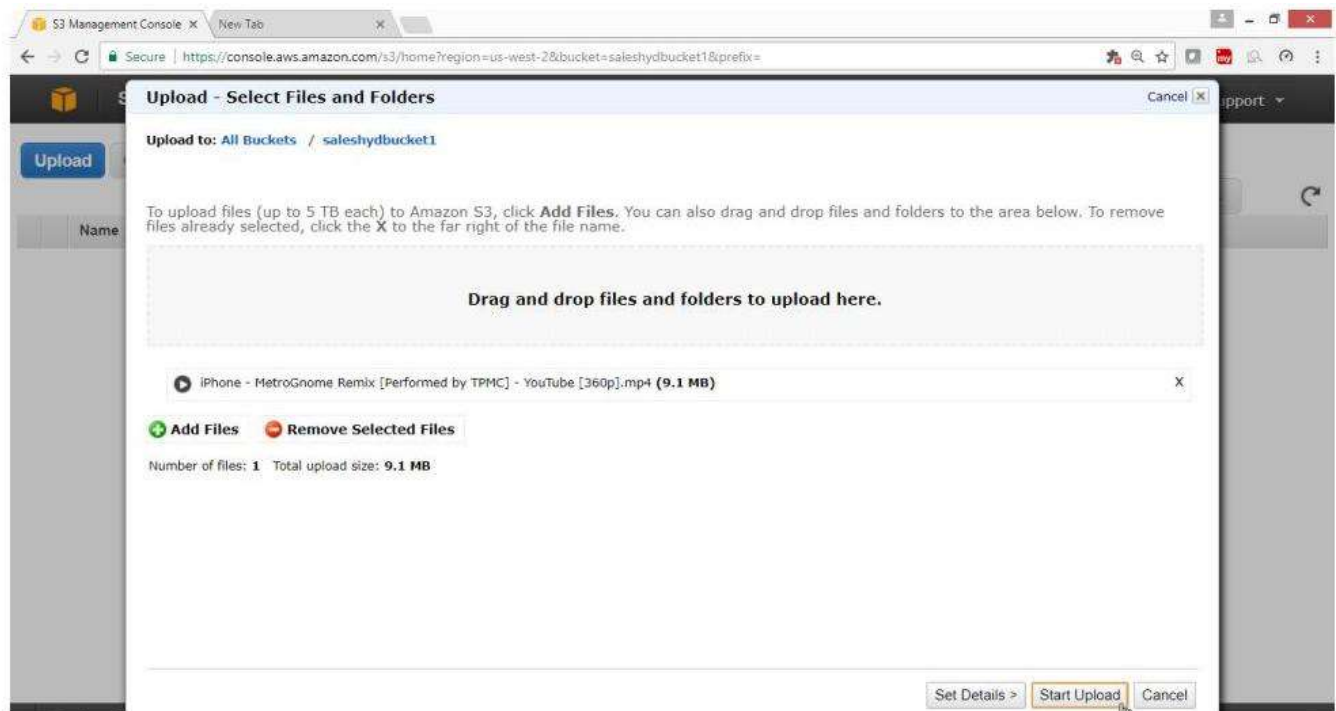


In the Upload Wizard

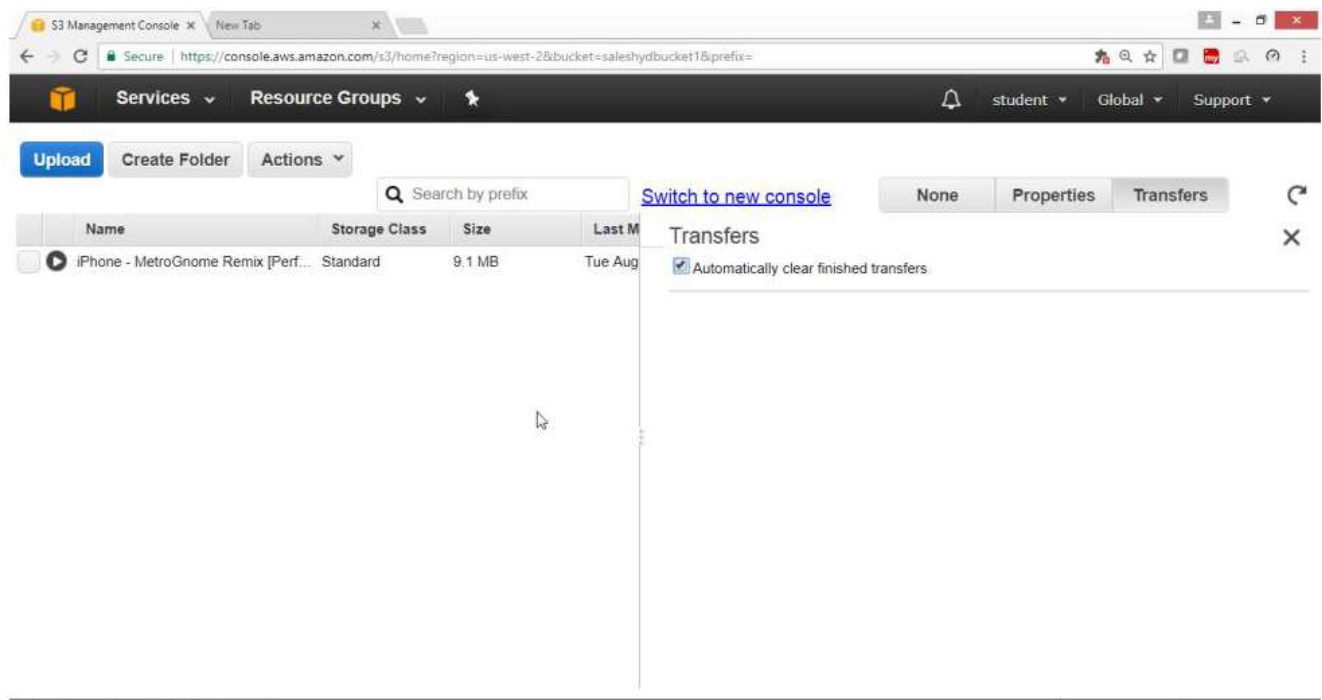
Click on **Add files**

Select some txt, pdf, video files

Click "**Start Upload**"



Verify that the file got uploaded



Select the **file**, Click on **Properties** on Right Panel,

Click on the link

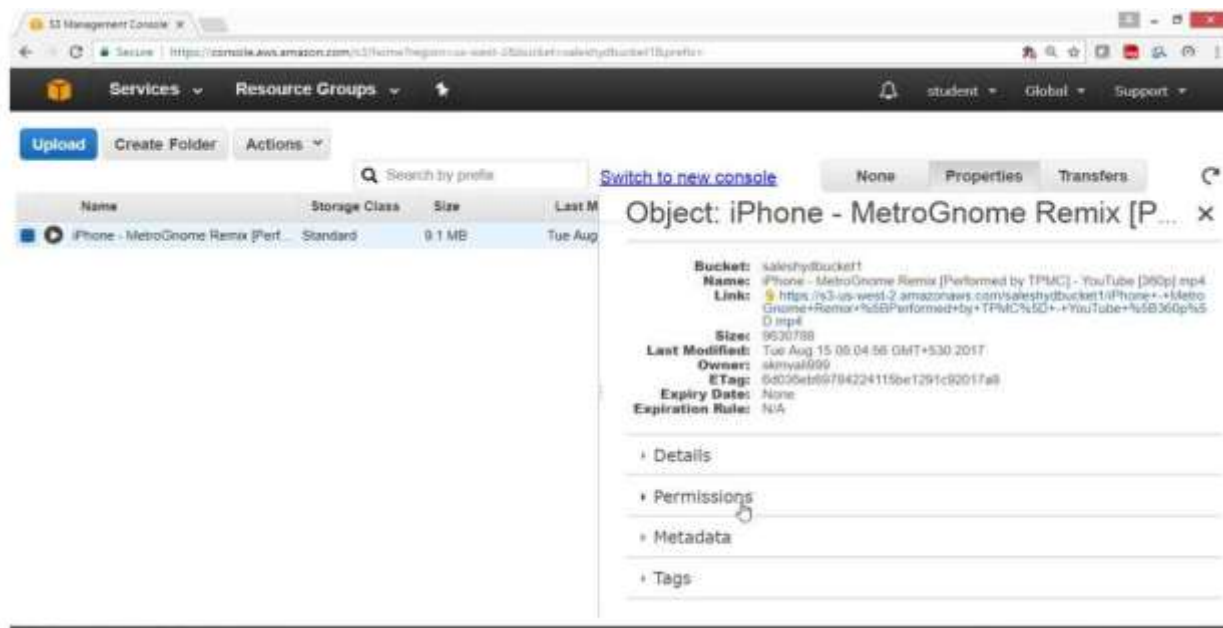
The screenshot shows the AWS S3 Management Console interface. At the top, there's a navigation bar with 'Services', 'Resource Groups', and a search bar. Below this, there's a table of objects. The first object is 'iPhone - MetroGnome Remix [Perf...]' with a size of 9.1 MB and a last modified date of Tue Aug. To the right of the table, there's a detailed view of the selected object. The details include the bucket name 'saleshydbucket1', the object name 'iPhone - MetroGnome Remix [Performed by TPMC] - YouTube [360p].mp4', a link to the object, its size (9630788), last modified date (Tue Aug 15 08:04:56 GMT+530 2017), owner (skmval999), ETag (6d036eb69784224115be1291c92017a9), expiry date (None), and expiration rule (N/A). Below the details, there are tabs for 'Details', 'Permissions', 'Metadata', and 'Tags'.

Verification: Cannot access due to lack of permission

The screenshot shows the AWS S3 Management Console interface. At the top, there's a navigation bar with 'Services', 'Resource Groups', and a search bar. Below this, there's a table of objects. The first object is 'iPhone - MetroGnome Remix [Perf...]' with a size of 9.1 MB and a last modified date of Tue Aug. To the right of the table, there's a detailed view of the selected object. The details include the bucket name 'saleshydbucket1', the object name 'iPhone - MetroGnome Remix [Performed by TPMC] - YouTube [360p].mp4', a link to the object, its size (9630788), last modified date (Tue Aug 15 08:04:56 GMT+530 2017), owner (skmval999), ETag (6d036eb69784224115be1291c92017a9), expiry date (None), and expiration rule (N/A). Below the details, there are tabs for 'Details', 'Permissions', 'Metadata', and 'Tags'.

To allow users to Download, or view give permission

Select, "Permission" tag

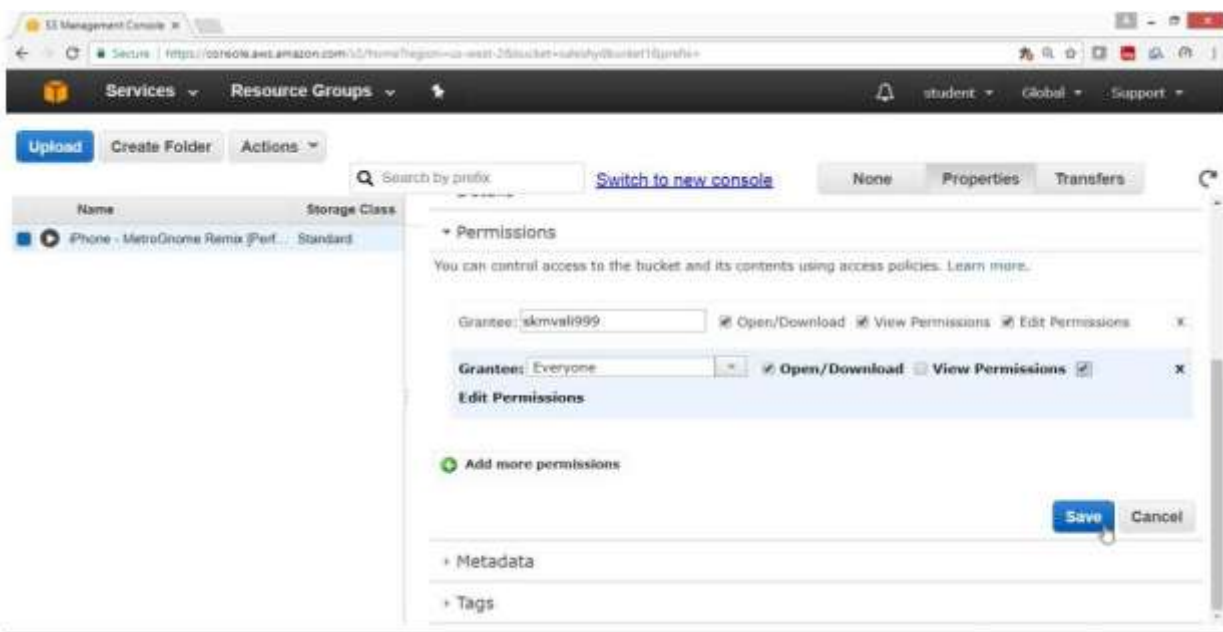


Click on "Plus Radio" button for "Add more permissions"

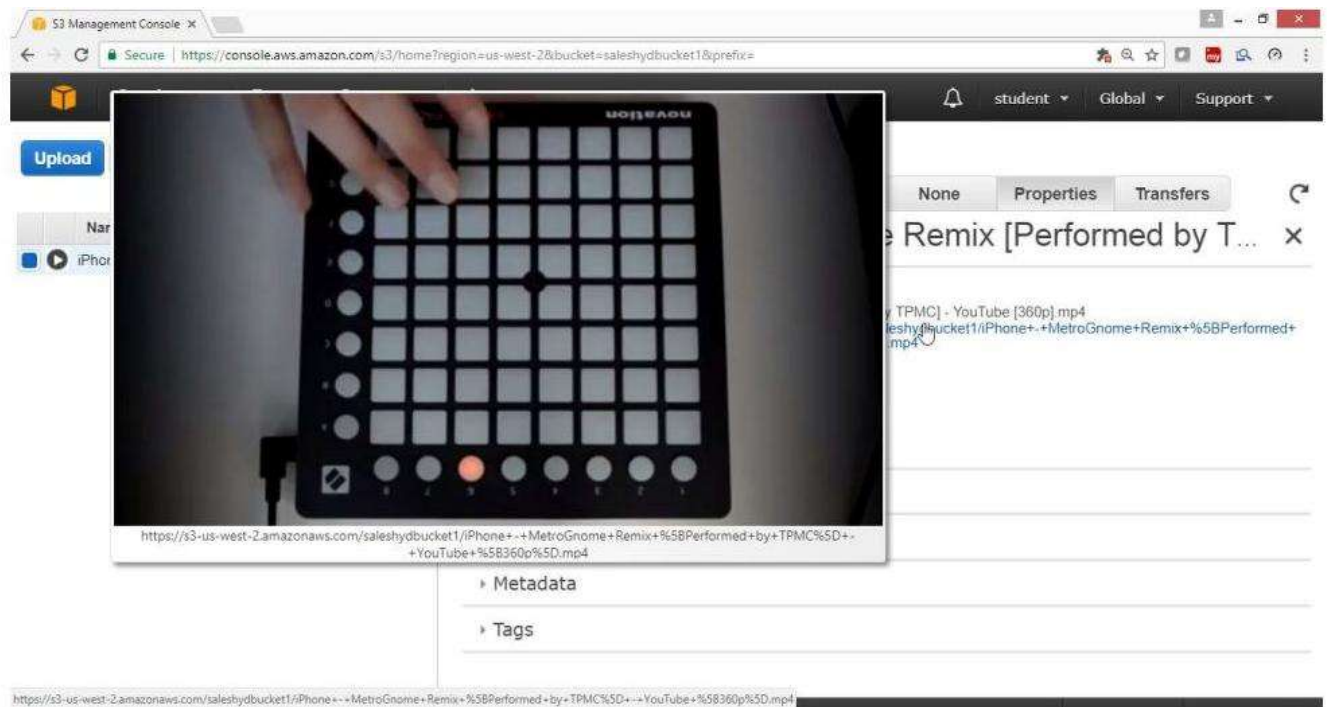
Drop down "Grantee" Button

- Select "Everyone" to make it public
- Enable the check box to Open/Download
- Enable the check box to View Permission
- Enable the check box the Edit View Permission

Click on "Save" Button



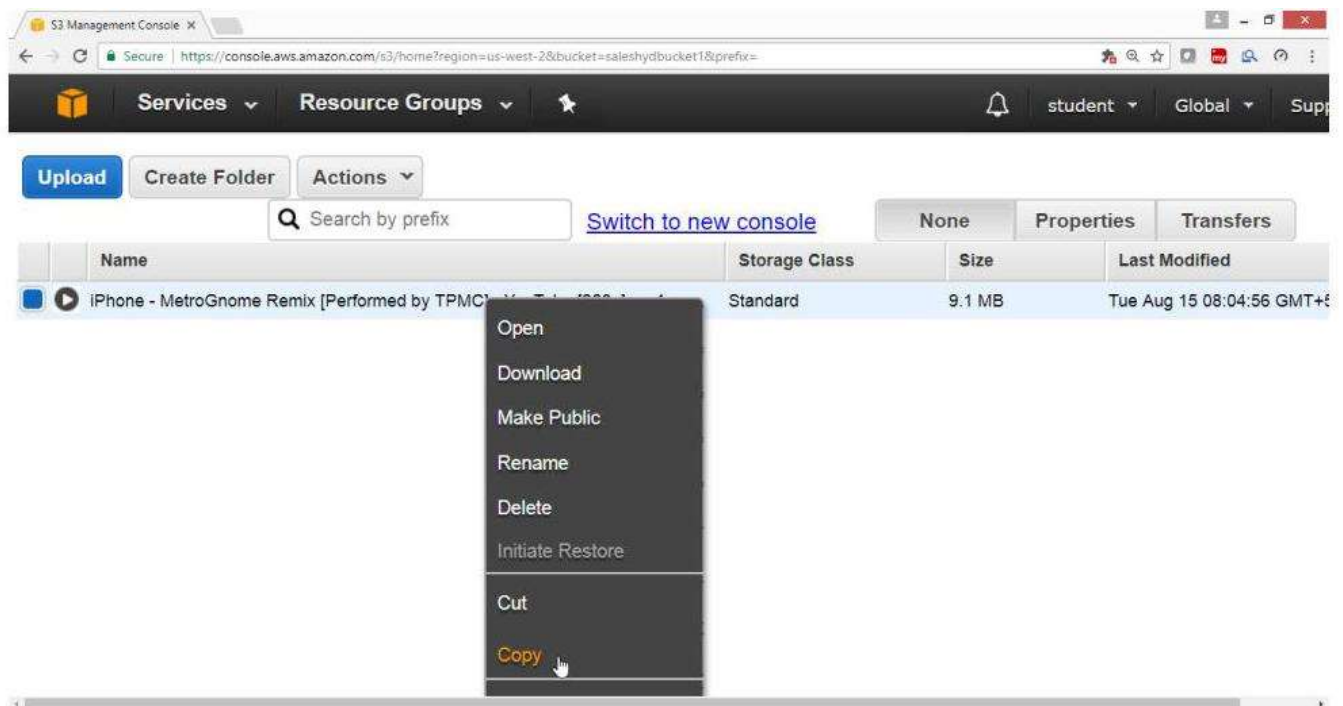
Verify file is accessible



To copy or move files from one bucket to another

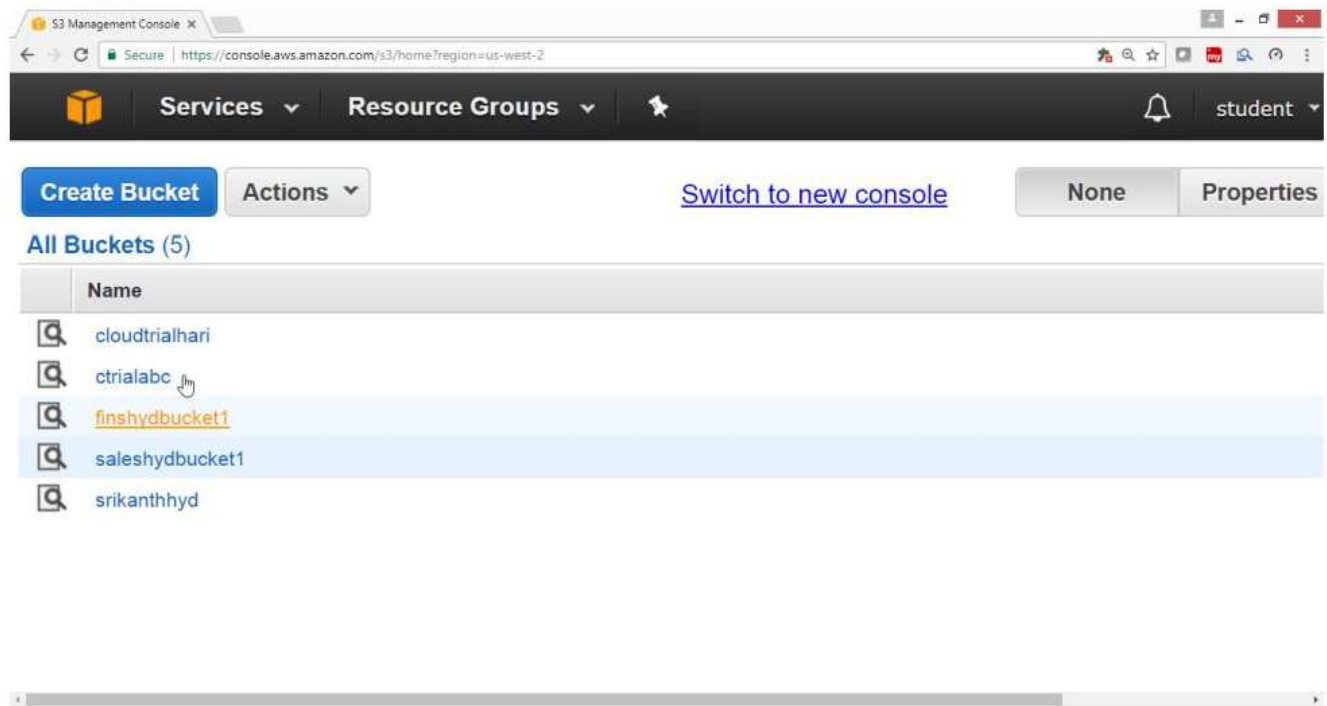
Select the file from Bucket or Folder, right click

now select copy/cut

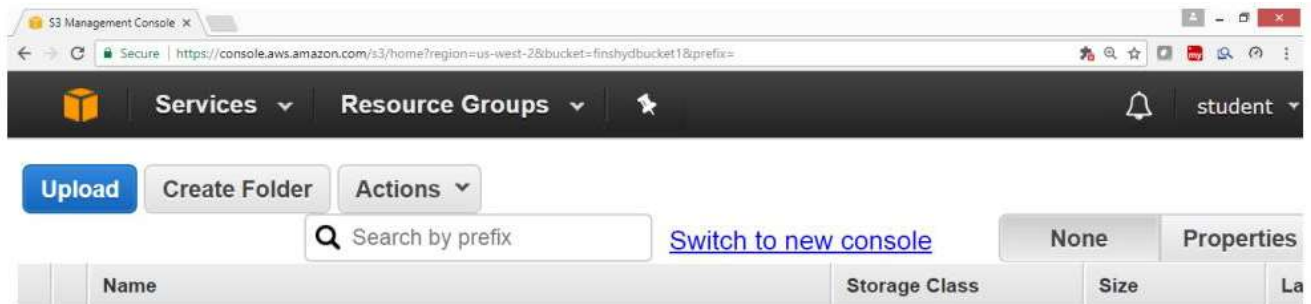


Select the Bucket or Folder, where you want to paste

Click on the Bucket ->finishydbucket1



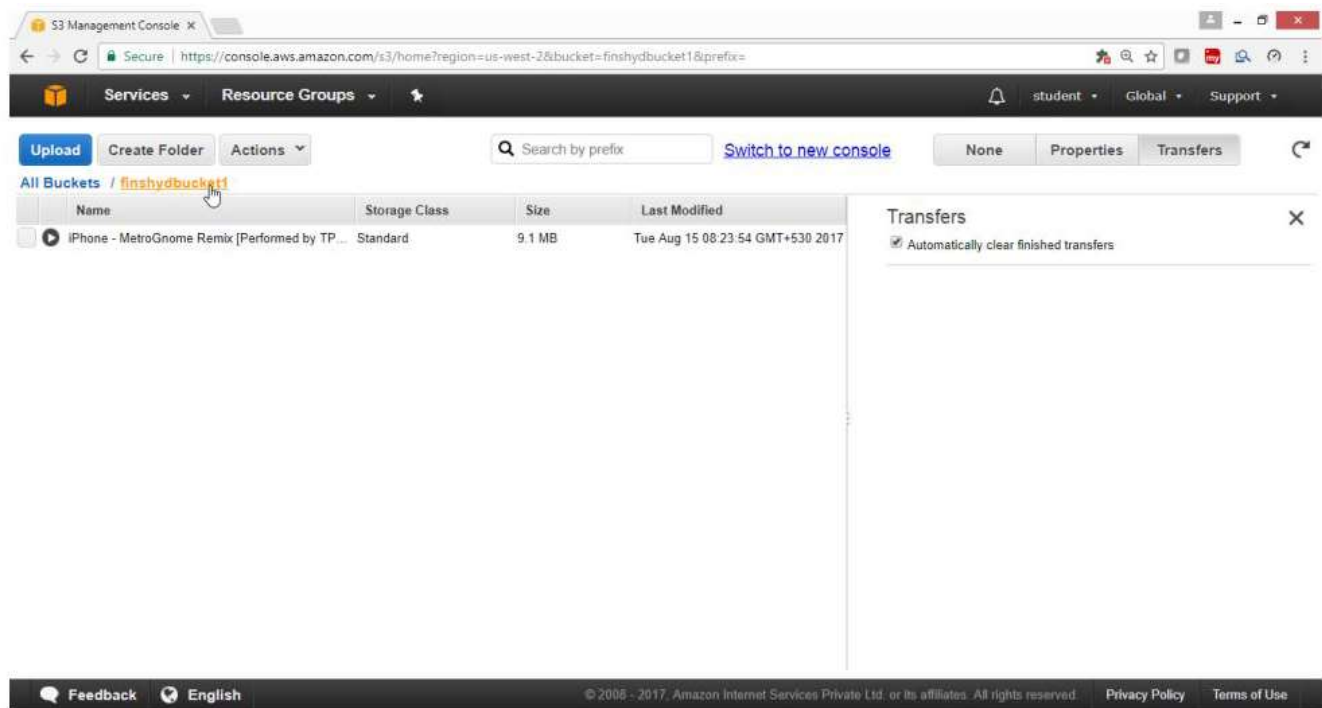
Click on "Paste"



The bucket 'finshydbucket1' is empty

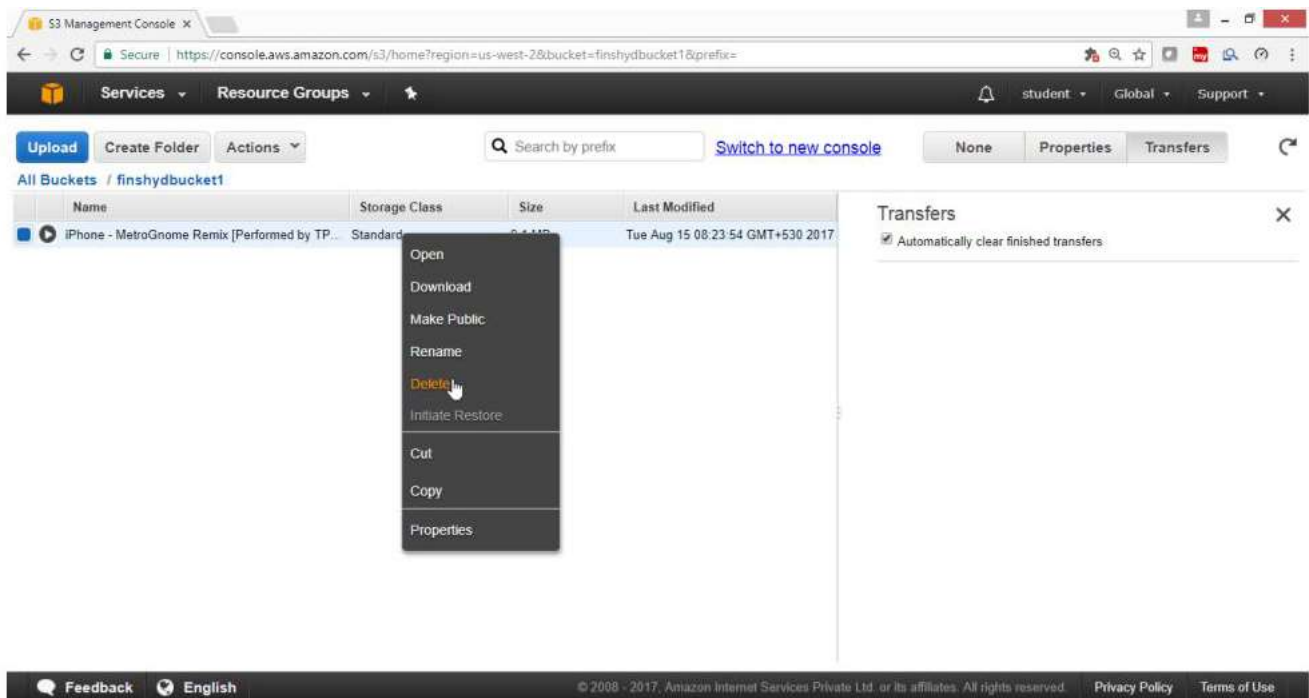


Verify that the file is copied in another bucket i.e., finshydbucket1



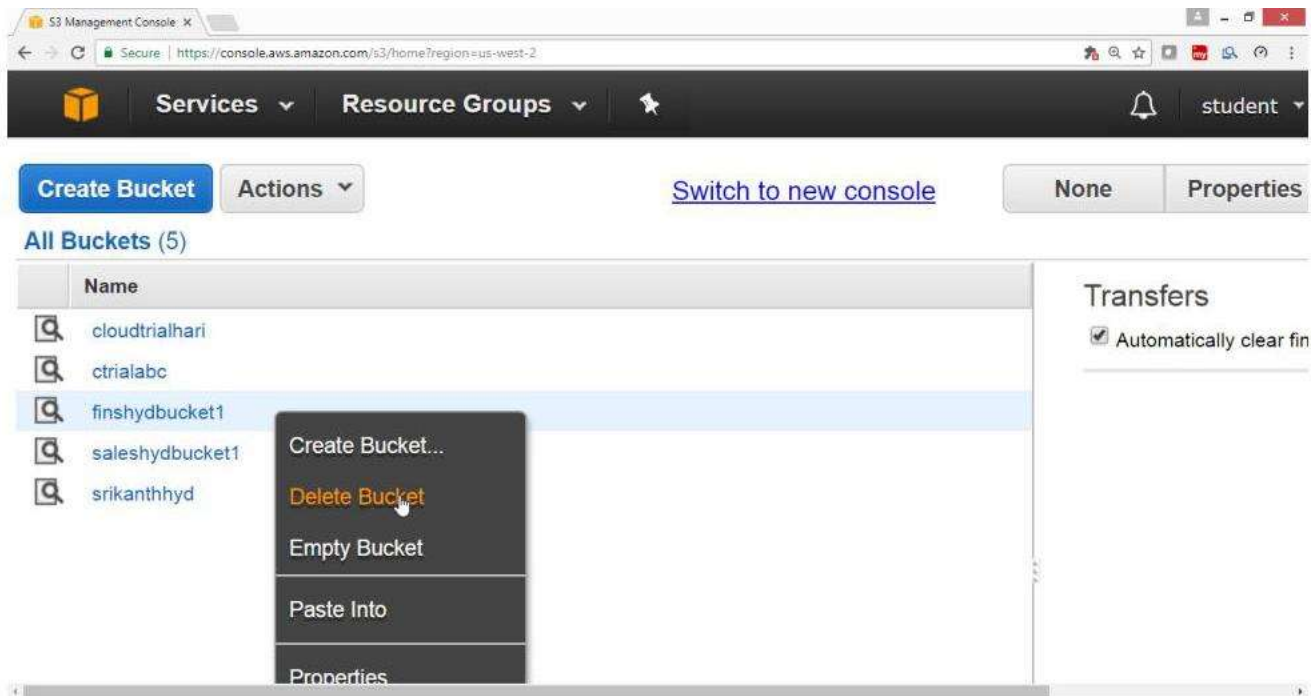
To delete a file from a Bucket

Right click on it, Select **Delete**



To Delete a bucket

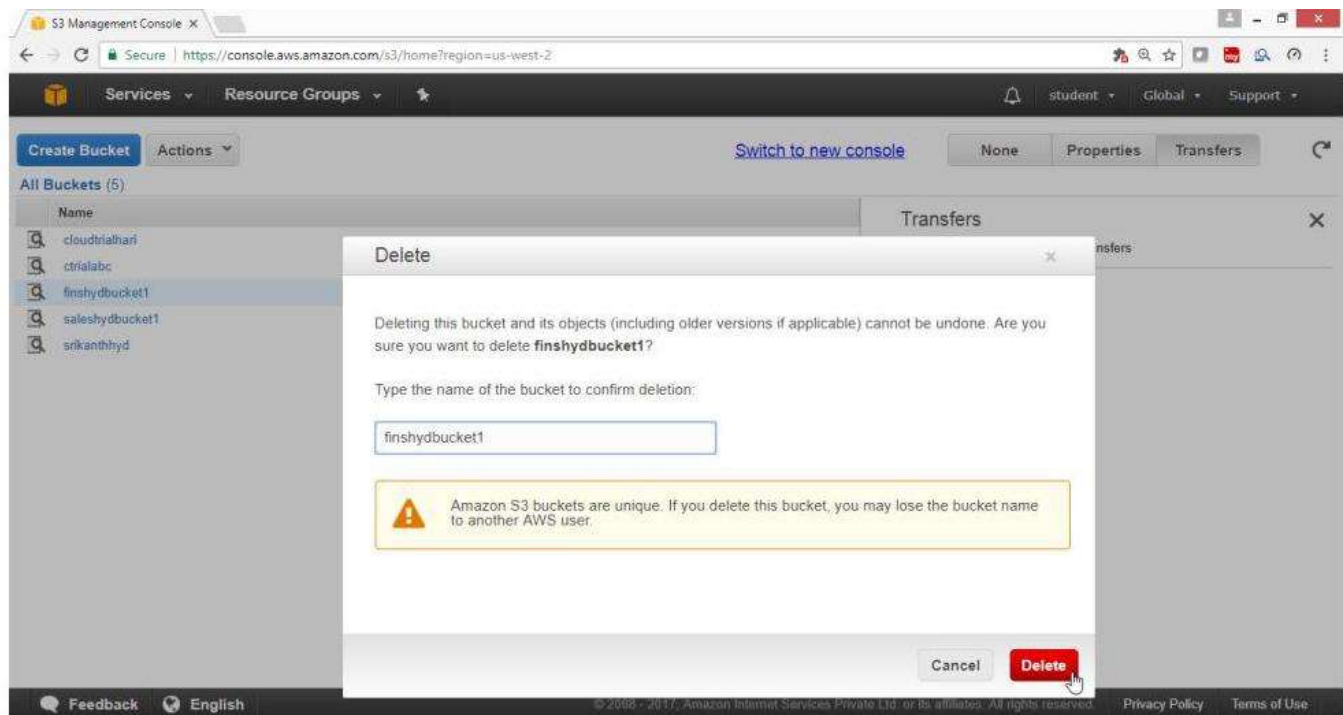
Select the bucket, right click select "**Delete Bucket**"



To "**Delete a bucket**"

Provide exact **bucket name**

Click on "**Delete**" Button



Verify that the bucket **finshydbucket1** is deleted

To host a Static Website using Amazon S3 Bucket

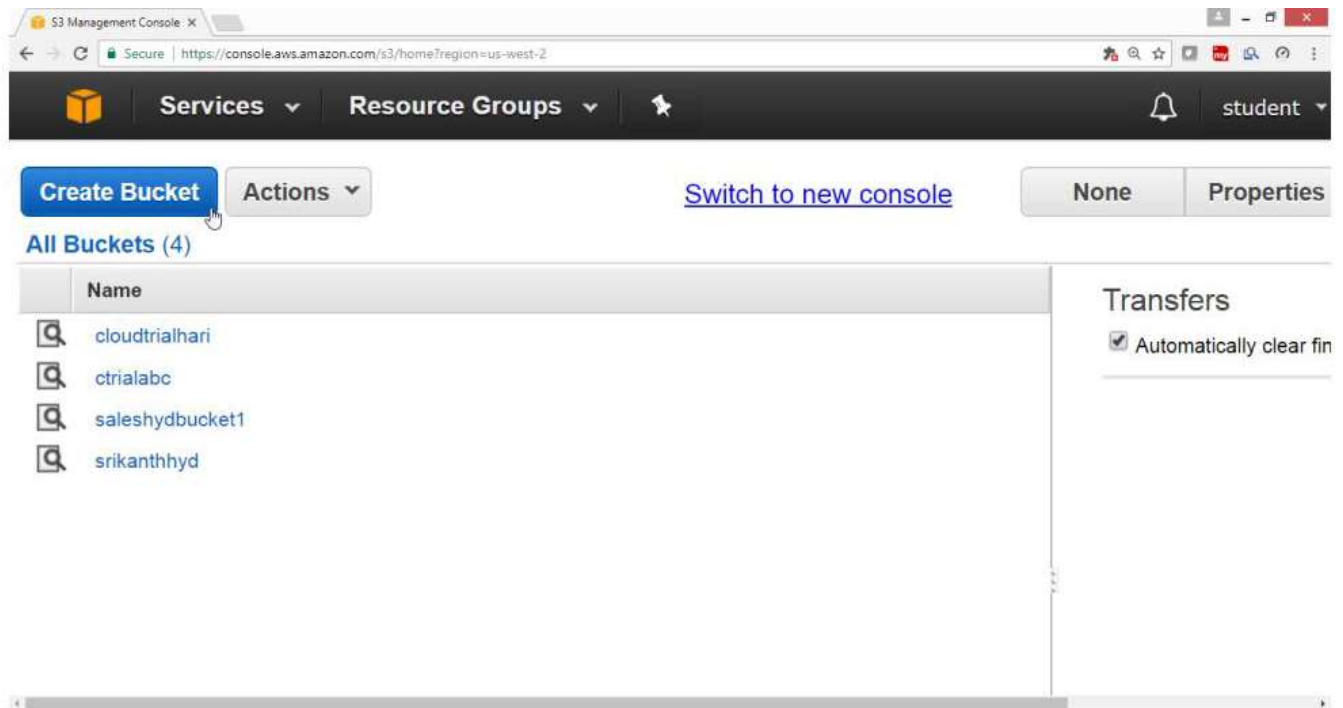
To host a Static Website using Amazon S3 Bucket

Open AWS console

Select "Storage"

Click on "S3" service

Click on "Create Bucket"

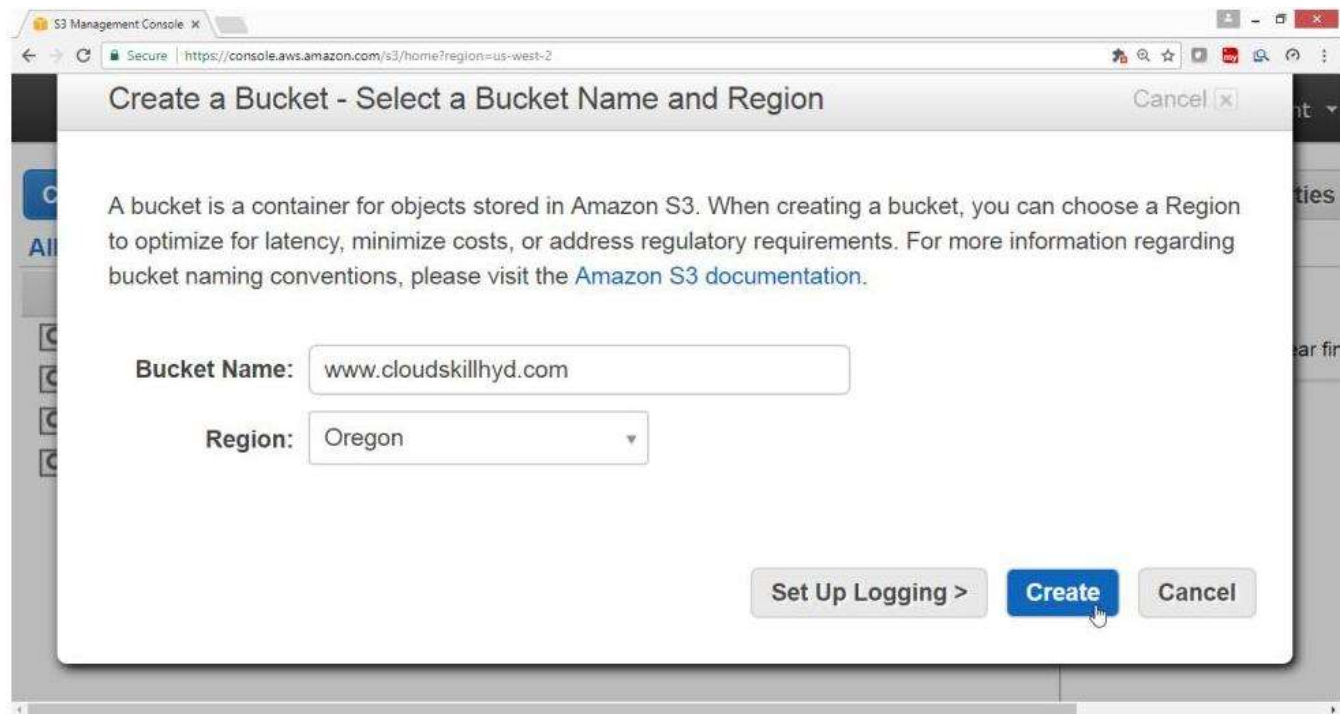


On "Create a Bucket - Select a Bucket Name and Region" Page

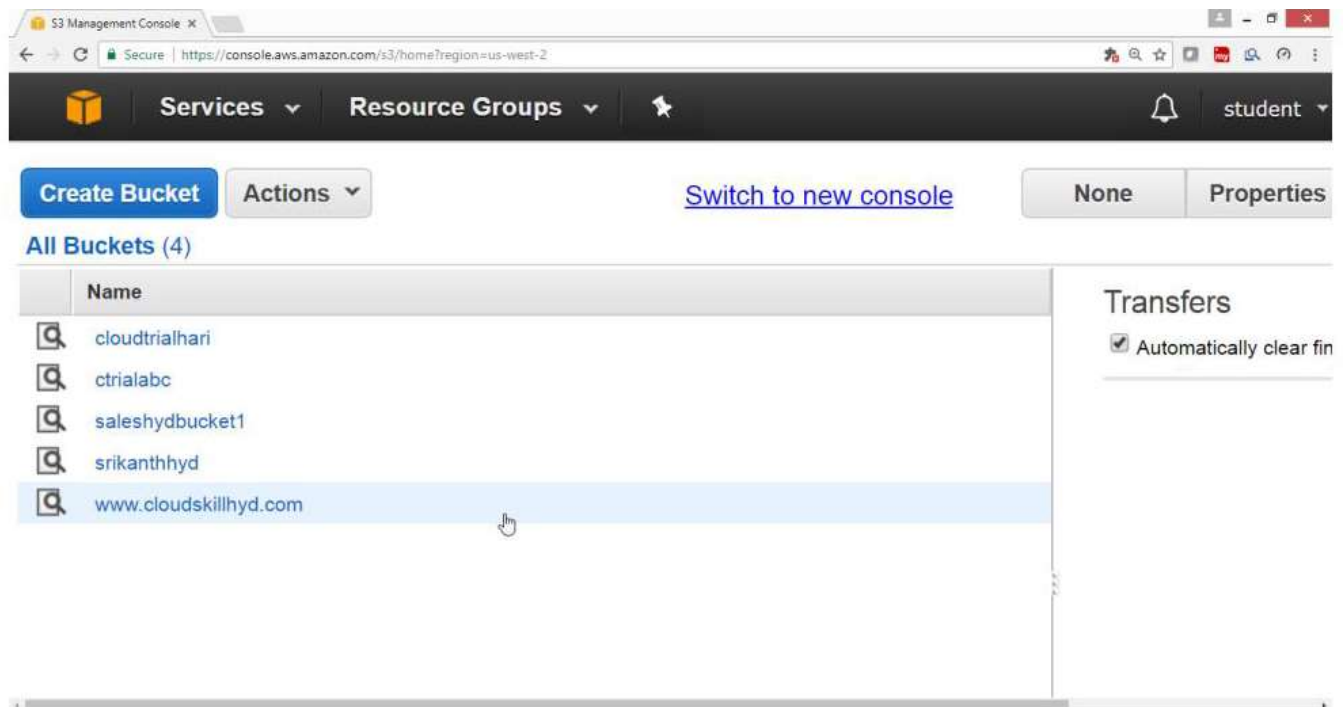
Provide following values for

- Bucket Name -> www.cloudskillhyd.com
- Region -> Oregon

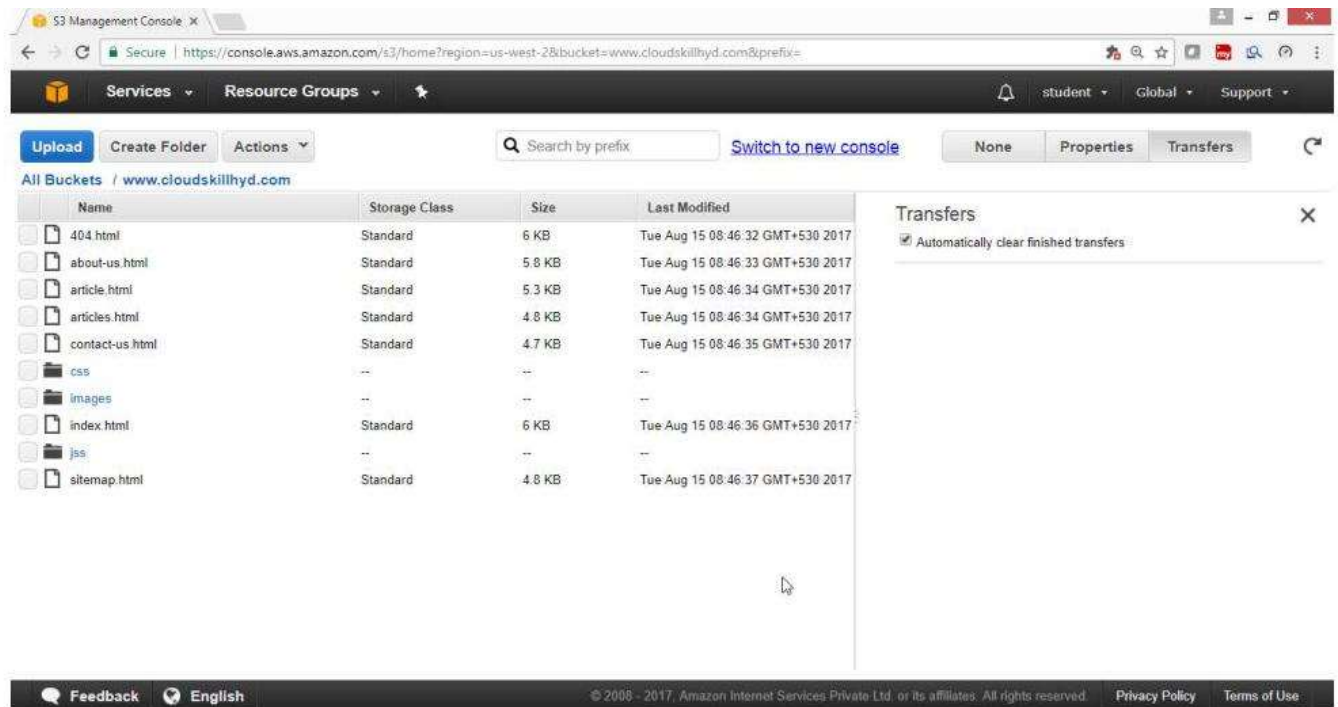
Click on Create button



Verify **Bucket** got created



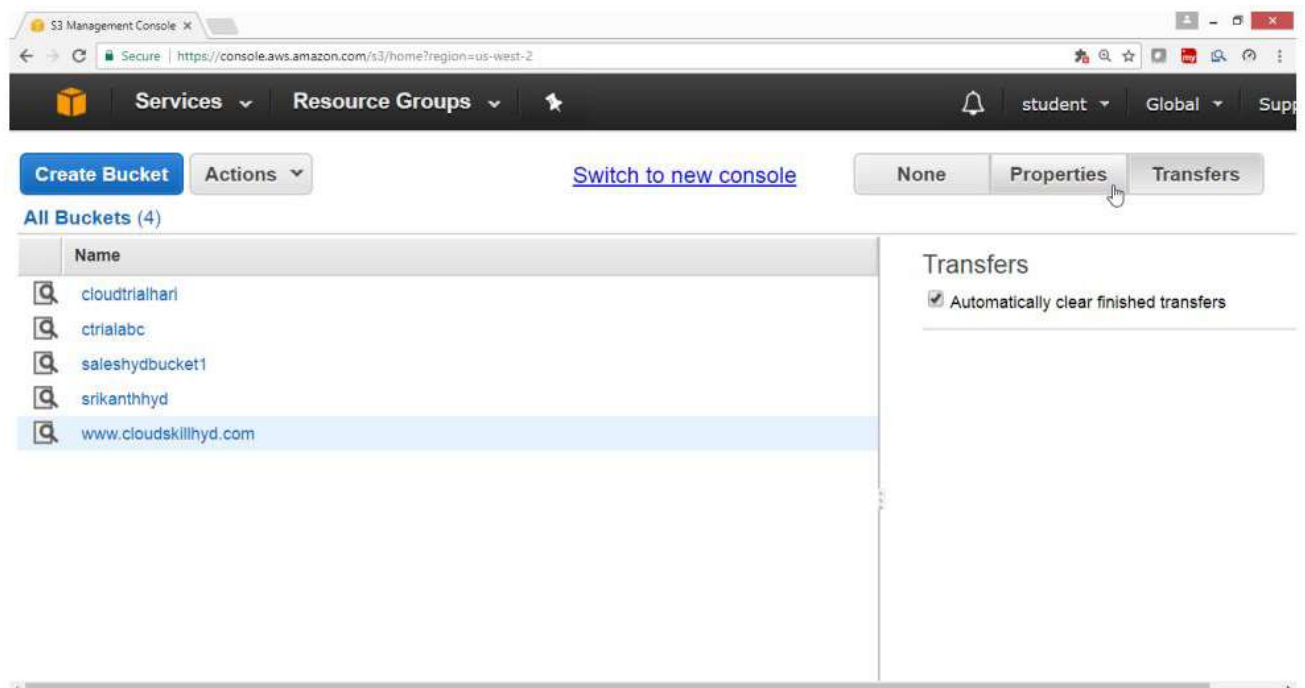
Upload all website content in this bucket



The screenshot shows the AWS S3 Management Console interface. The top navigation bar includes 'Services', 'Resource Groups', and a search bar. The main content area displays the bucket 'www.cloudskillhyd.com' with a table of its contents. The table has columns for Name, Storage Class, Size, and Last Modified. The contents include files like '404.html', 'about-us.html', 'article.html', 'articles.html', 'contact-us.html', 'css', 'images', 'index.html', 'js', and 'sitemap.html'. On the right side, there is a 'Transfers' panel with a checkbox for 'Automatically clear finished transfers'.

Name	Storage Class	Size	Last Modified
404.html	Standard	6 KB	Tue Aug 15 08:46:32 GMT+530 2017
about-us.html	Standard	5.8 KB	Tue Aug 15 08:46:33 GMT+530 2017
article.html	Standard	5.3 KB	Tue Aug 15 08:46:34 GMT+530 2017
articles.html	Standard	4.8 KB	Tue Aug 15 08:46:34 GMT+530 2017
contact-us.html	Standard	4.7 KB	Tue Aug 15 08:46:35 GMT+530 2017
css	--	--	--
images	--	--	--
index.html	Standard	6 KB	Tue Aug 15 08:46:36 GMT+530 2017
js	--	--	--
sitemap.html	Standard	4.8 KB	Tue Aug 15 08:46:37 GMT+530 2017

Select the bucket and click on properties button



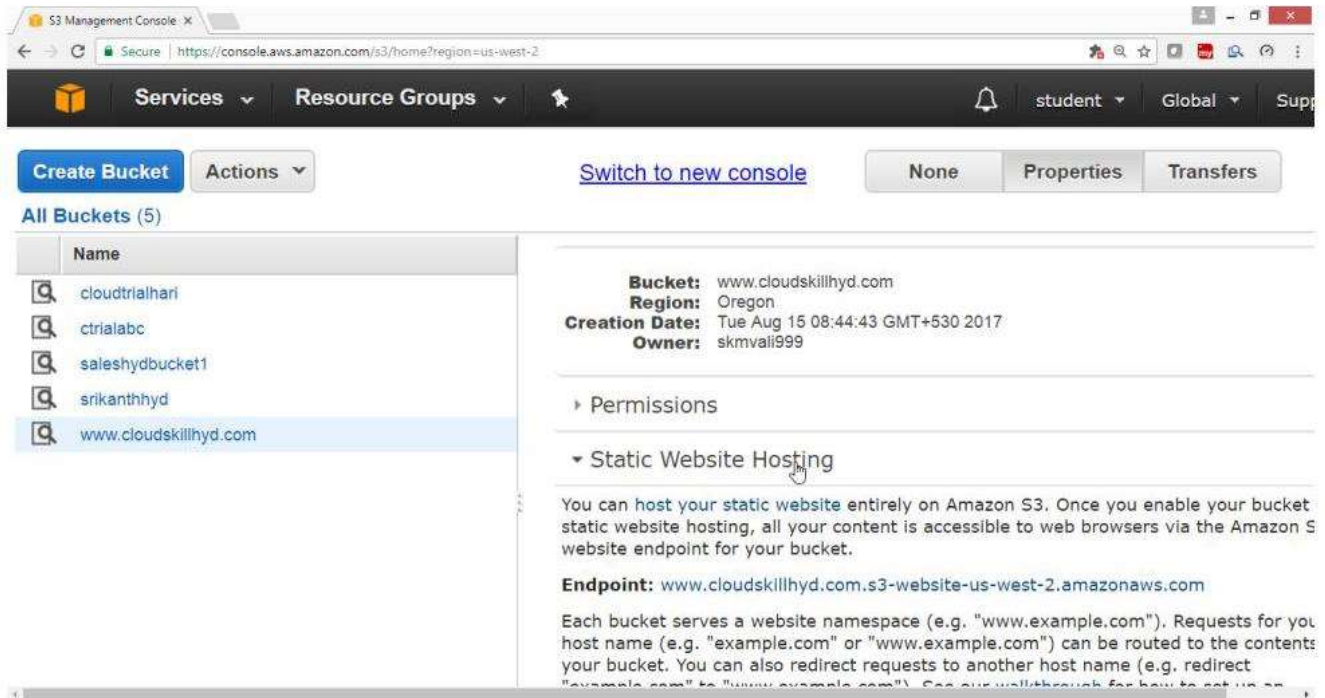
The screenshot shows the AWS S3 Management Console interface. The top navigation bar includes 'Services', 'Resource Groups', and a search bar. The main content area displays the 'All Buckets (4)' list. The list has a column for Name. The buckets listed are 'cloudtrialhari', 'ctrialabc', 'saleshydbucket1', 'srikanthhyd', and 'www.cloudskillhyd.com'. The bucket 'www.cloudskillhyd.com' is selected. On the right side, there is a 'Transfers' panel with a checkbox for 'Automatically clear finished transfers'.

Name
cloudtrialhari
ctrialabc
saleshydbucket1
srikanthhyd
www.cloudskillhyd.com

On the **Properties** panel

Click **Static Website Hosting**

Drag Down



Select the **Enable Website Hosting**

Provide following values for

- Index Document box -> index.html
- Error Documentation box -> 404.html

Click on **Save** button

S3 Management Console

Services Resource Groups

student Global Support

Bucket Actions

Switch to new console

None Properties Transfers

Buckets (5)

Name
cloudtrialhari
trialabc
saleshydbucket1
srikanthhyd
www.cloudskillhyd.com

Enable website hosting

Index Document:

Error Document:

Edit Redirection Rules: You can set custom rules to automatically redirect web page requests for specific content.

Redirect all requests to another host name

Save **Cancel**

Logging

Note down the Endpoint

Services Resource Groups

student Global Support

Create Bucket Actions

Switch to new console

None Properties Transfers

All Buckets (5)

Name
cloudtrialhari
trialabc
saleshydbucket1
srikanthhyd
www.cloudskillhyd.com

Endpoint: www.cloudskillhyd.com.s3-website-us-west-2.amazonaws.com

Each bucket serves a website namespace (e.g. "www.example.com"). Requests for your host name (e.g. "example.com" or "www.example.com") can be routed to the contents in your bucket. You can also redirect requests to another host name (e.g. redirect "example.com" to "www.example.com"). See our walkthrough for how to set up an Amazon S3 static website with your host name.

Do not enable website hosting

Enable website hosting

Index Document:

Error Document:

Edit Redirection Rules: You can set custom rules to automatically redirect web page requests for specific content.

Redirect all requests to another host name

Save **Cancel**

Logging

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

To add a bucket policy that makes your bucket content publicly available

In the Bucket properties, click on "Permission"

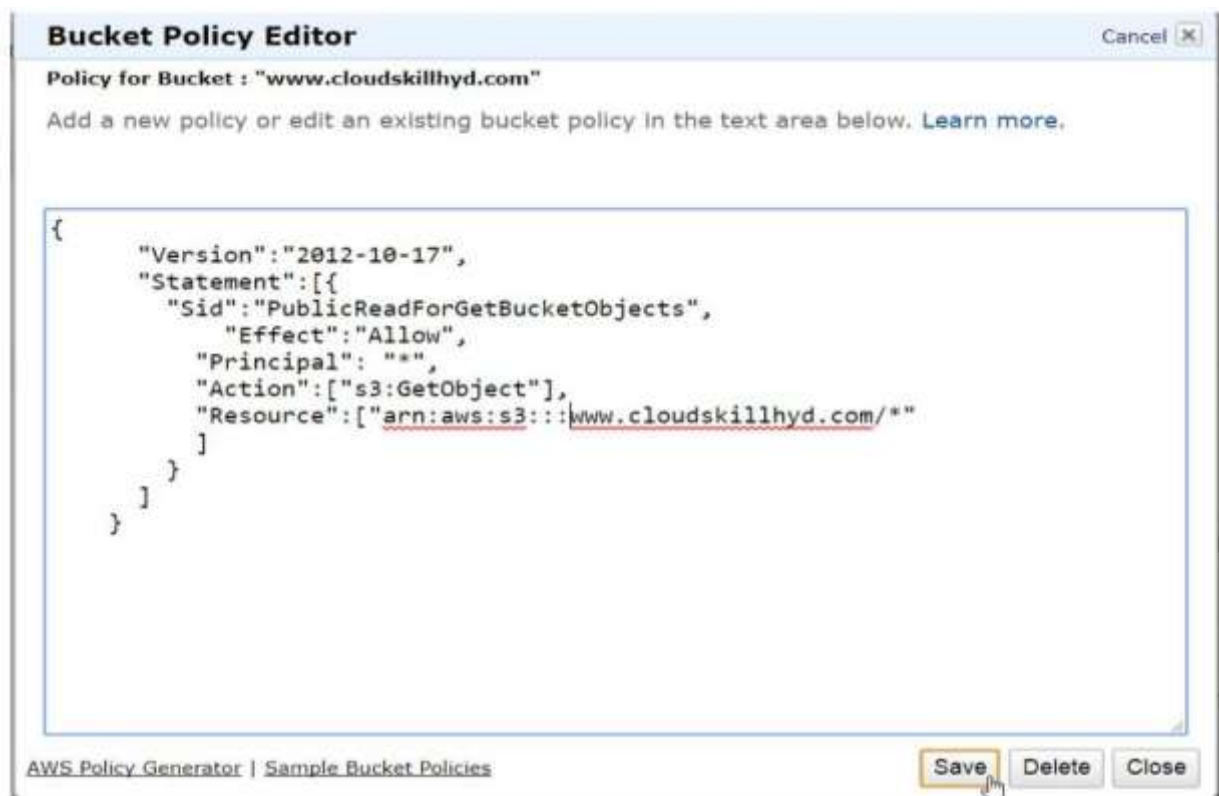
Click on "Add Bucket Policy"

The screenshot shows the AWS Management Console interface for a bucket named 'www.cloudskillhyd.com'. The 'Permissions' tab is selected, showing a list of buckets on the left and the bucket's details on the right. The 'Grantee' is 'skmvali999' and the permissions 'List', 'Upload/Delete', and 'View Permissions' are checked. The 'Add bucket policy' button is highlighted.

Copy the following bucket policy, and then paste it in the Bucket Policy Editor

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadForGetBucketObjects",
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["s3:GetObject"],
      "Resource": ["arn:aws:s3:::cloudskillhyd.com/*"]
    }
  ]
}
```

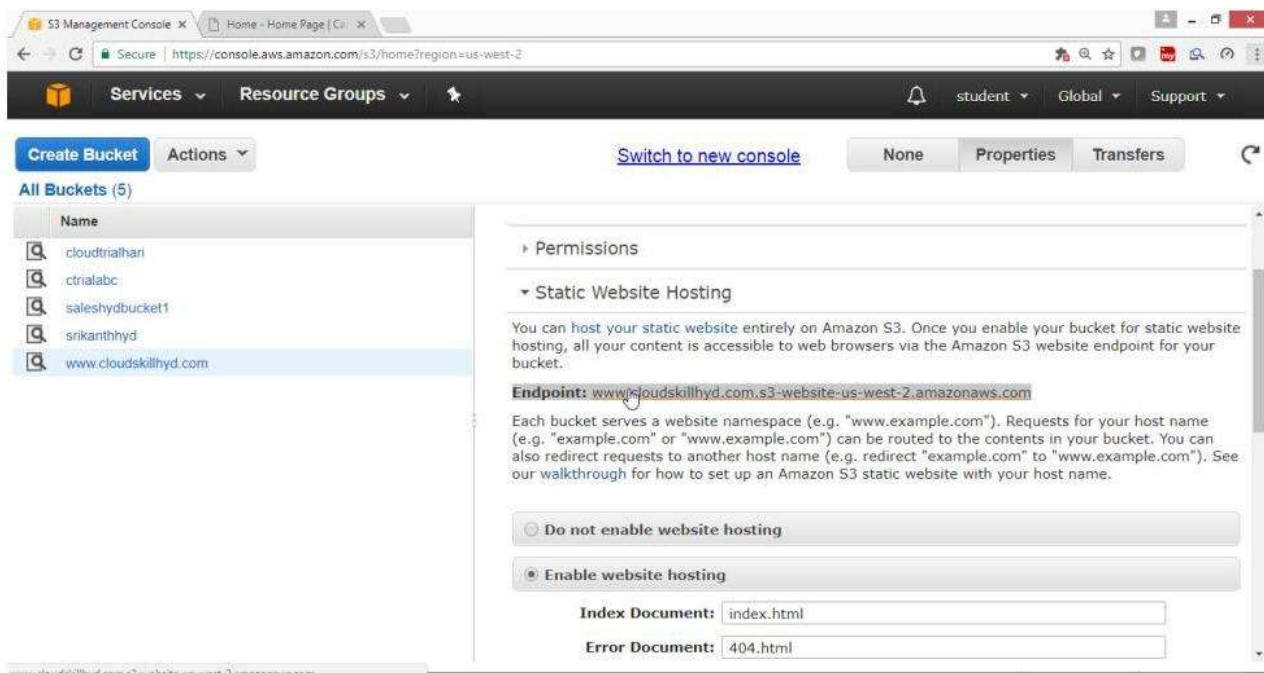
Click on "Save" button



Verify your website

Click on Endpoint under Static Website Hosting

Endpoint: www.cloudskillhyd.com.s3-website-us-west-2.amazonaws.com



Verify the website which is coming from S3 Bucket



Now you could give the link to the any one and through the link user can access the file which are stored in the folder of the S3 bucket.
















Welcome, Sriram

What is the need of Storage? What are the different storages available in AWS?

The need for storage is increasing every day, so building and maintaining your own repositories, therefore, becomes a tedious and tiresome job because knowing the amount of capacity you may need in the future is difficult to predict.

You may either over-utilize it leading to an application failure because of not having sufficient space or you may end up buying stacks of storage which will then be under-utilized. Keeping all these hassles in mind, Amazon came up with an internet storage service called AWS S3.

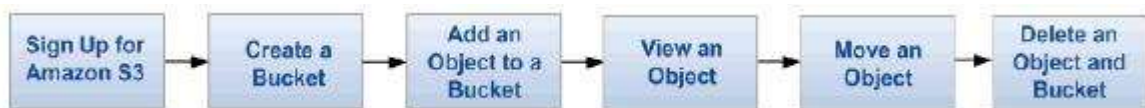
What are the storage available in S3?

	Amazon S3	Scalable storage in the cloud
	Amazon Glacier	Low-cost archive storage in the cloud
	Amazon EBS	Persistent block storage volumes for Amazon EC2 virtual machines
	Amazon EC2 Instance Storage	Temporary block storage volumes for Amazon EC2 virtual machines
	AWS Import/Export	Large volume data transfer
	AWS Storage Gateway	Integrates on-premises IT environments with cloud storage
	Amazon CloudFront	Global content delivery network (CDN)
	Amazon SQS	Message queue service
	Amazon RDS	Managed relational database server for MySQL, Oracle, and Microsoft SQL Server
	Amazon DynamoDB	Fast, predictable, highly-scalable NoSQL data store
	Amazon ElastiCache	In-memory caching service
	Amazon Redshift	Fast, powerful, full-managed, petabyte-scale data warehouse service
	Databases on Amazon EC2	Self-managed database on an Amazon EC2 instance

What is S3? What purpose S3 is designed for?

S3 stands for Simple Storage Service. You can use S3 interface to store and retrieve any amount of data, at any time and from anywhere on the web.

Also, we can host a website in Amazon S3. most of the companies storing the documents, images and other files to S3. For S3, the payment model is “pay as you go”. March 2006, Amazon launched Simple Storage Service (S3)



S3 is designed for:

- Remote data storage.
- Low cost, pay-as-you go.
- No up-front costs.
- High-availability.
- High bandwidth.



What are the core fundamentals of S3?

- Key (name)
- Value (data)
- Version ID
- Metadata
- Access Control Lists
- Object Based Storage File System
- Not Suitable to install an operating system on

What are the key Features of S3?

The key Features of S3 are: -

- 99.999999999% Durability.
- 99.99% Availability.

How is data organized in S3?

Data in S3 is organized in the form of buckets.

- A Bucket is a logical unit of storage in S3.
- A Bucket contains objects which contain the data and metadata.
- Before adding any data in S3 the user has to create a bucket which will be used to store objects.



What are the key concepts of S3 storage? What is bucket & What is object?

The Key concepts are S3 storage are Buckets & Objects

Buckets

- Equivalent to Directories, common namespace across S3
- A basic storage unit, Collection of Objects.
- Name of the bucket should be globally unique id a-z A-Z 0-9 . - .
- It is a single level container (no hierarchy), can contain multiple folders, or objects can be placed directly.
- Based on key-object associations
- Upload and download are easier.
- Allows maximum 100 buckets per user.
- No size restriction for Bucket.
- Data kept secured from unauthorized access through authentication mechanism.

Objects

- Equivalent to files.
- Allows max of 5TB for a single object.
- Identified by key (== filename)

What are the Object level properties?

The object level properties are:-

1. Details:

STANDARD: 99.99999999%

STAND.INFREQUENT: LESSER THAN STANDARD.

Reduced Redundancy: 99.99% , LESSER THAN S.IA.

AES 256 - ADVANCED ENCRYPTION STANDARD.

2. Permission:

- ME, ALL, RECOG.USER.

- OPEN, VIEW, EDIT.

3. Metadata: Data's Data

Format of the file. EG: HTML / JPEG.

4. Tags: INPROGRESS

What are the best practices on naming S3 bucket?

- DNS compatible
- FQDN ○ Allows for vhost ○ watch out for SSL: no dots :-(Objects)
- Blob
- Don't care about file formats
- Metadata can be added (like mimetype)
- Maximum 5 TB/object

How would you plan your data to be stored geographically?

You can self-choose where or in which region your data should be stored. Making a decision for the region is important and therefore it should be planned well.

These are the 4 parameters to choose the optimal region –

- Pricing
- User/Customer Location
- Latency
- Service Availability

We can clearly identify, that **N Virginia will be the best region** for this company because of the low latency and low price. Irrespective of your location, you can select any region which might suit your requirements, since you can access your S3 buckets from anywhere.

How to Access AWS S3 storage?

- Accessible using simple HTTP URLs
 - `http://s3.amazonaws.com/bucket/key`
 - `http://bucket.s3.amazonaws.com/key`
 - `http://bucket/key`where bucket is a DNS CNAME record pointing to s3.amazonaws.com)
- Use Amazon AWS Management Console
- Use Cloudberry Explorer
- S3 allows you to specify an Access Control List for every object in the database
- You can set permissions for the owner, for authenticated user, for specific users (e-mail & Amazon ID) for everybody
- It is even possible to create public URLs that expire at a given date

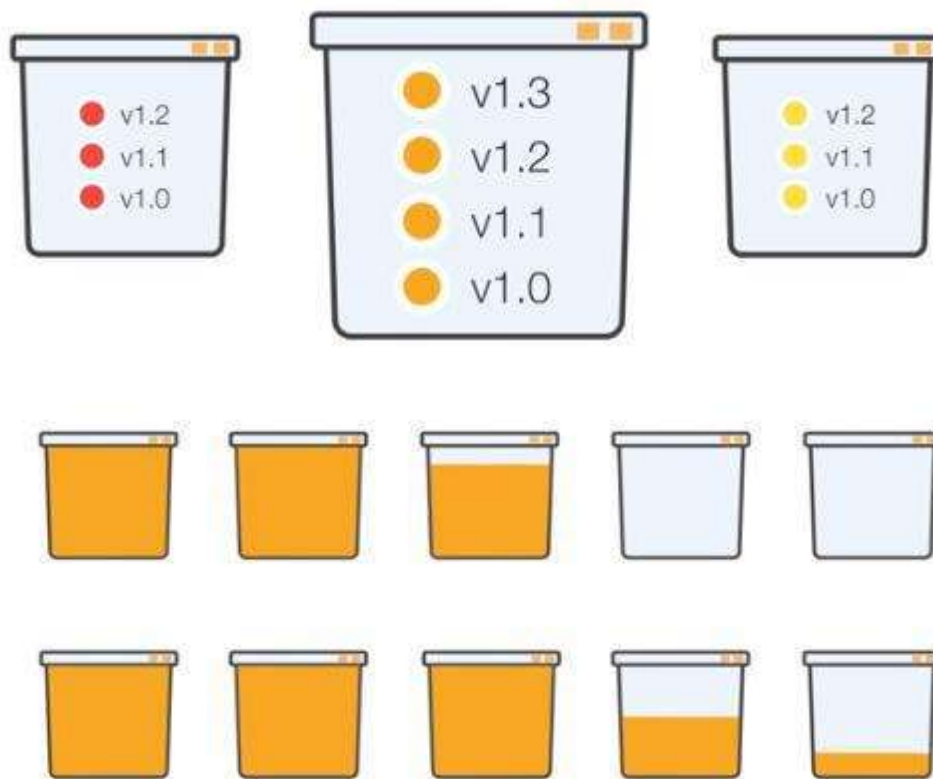


What is the Lifecycle Management in S3?

- Can be used in conjunction with versioning
- Can be applied current versions and previous versions
- Following actions can now be done: -
 - Transition to the standard - Infrequent Access Storage Class (128 Kb and 30 days after creation data)
 - Archive to the Glacier storage class (30 days after IA, if relevant)

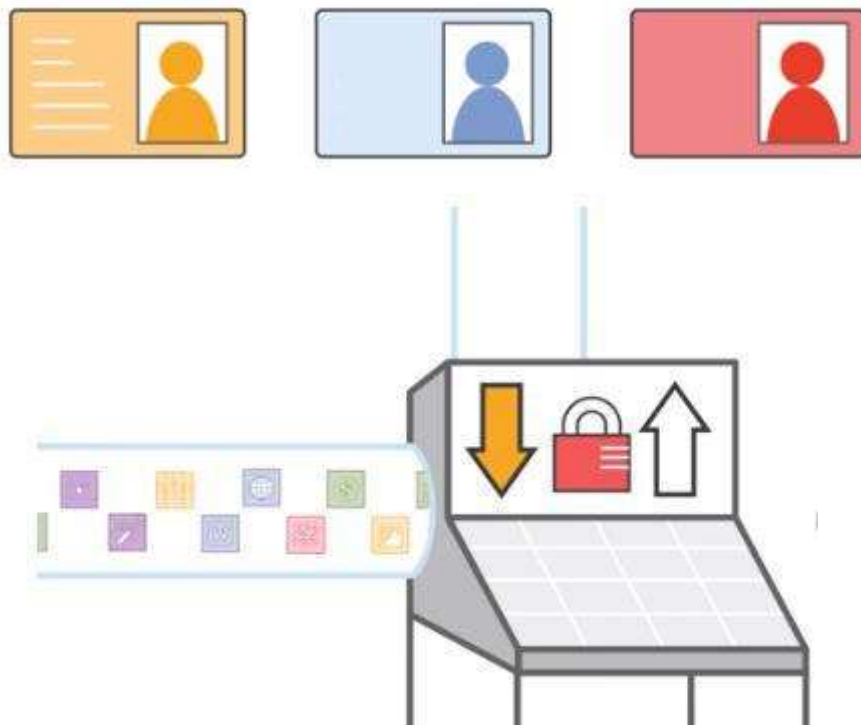
How Versioning is maintained in S3?

- Stores all versions of an object (including all writes and even if you delete an object)
- Great backup tool
- Once enabled, versioning cannot be disabled, only suspended.
- Integrates with Lifecycle rules
- Versioning's MFA Delete capability, which uses multi-factor authentication, can be used to provide an additional layer of security
- Cross Region Replication, requires versioning enabled on the source bucket
- It maintains the versions of Objects stored in S3 and recover in case of data loss



How Security is done in S3?

- By default, all newly created buckets are **PRIVATE**
- You can setup access control to your buckets using: -
 - **Bucket Policies**
 - **ACL's**
 - **Key Authentication**
 - **S3 also offer SSL encryption for data upload & download**
- **S3 Buckets can be configured to create access logs which log all requests made to the s3 bucket.**
This can be done to another bucket



What are the different tiers in Amazon S3 storage?

Different Storage tiers in Amazon S3 are as follows:

S3 Standard: In this tier, S3 supports durable storage of files that become immediately available. This is used for frequently used files.

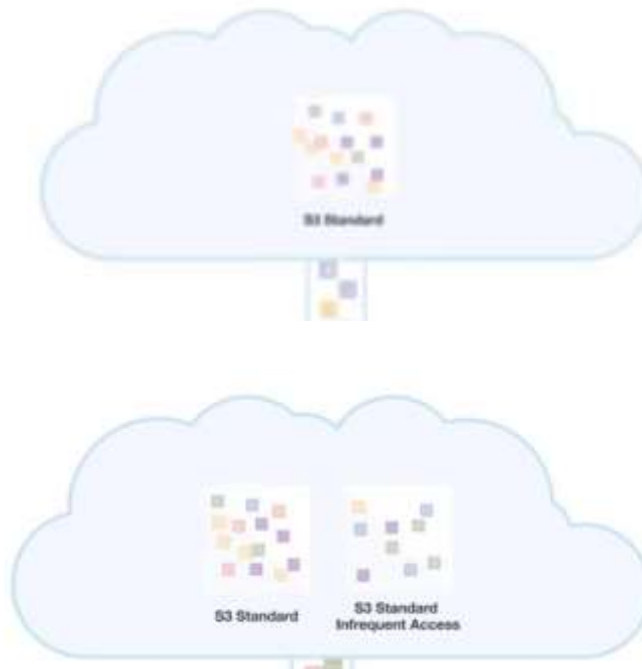
S3 Standard -Infrequent Access (IA): In this tier, S3 provides durable storage that is immediately available. But in this tier files are infrequently accessed.

S3 Reduced Redundancy Storage (RRS): In this tier, S3 provides the option to customers to store data at lower levels of redundancy. In this case data is copied to multiple locations but not on as many locations as standard S3.

What are the S3 range of classes? What are the S3 Storage Classes / Tiers?

- **S3 (Durable, immediately available, frequently accessed)**
- **S3 - IA (Durable, immediately available, infrequently accessed)**
- **S3 - Reduced Redundancy Storage (data that is easily reproducible, such as thumb nails, etc)**

- Glacier - archived data, where you can wait 3-5 hours before accessing
S3 Standard class for frequently accessed data



We can also setup auto policy to migrate data from one class to another class like standards to Glacier etc.,

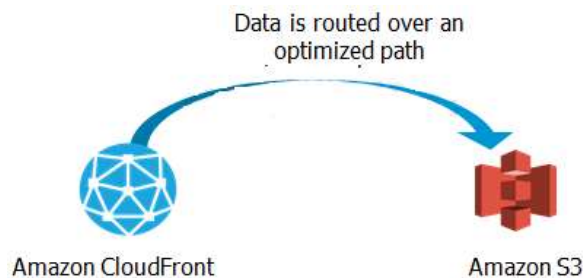


Characteristics	Standard	Standard - Infrequent Access	Glacier
Durability	99.99%	99.99%	99.99%
Availability	99.99%	99.90%	N/A
Minimum Object Size	No limit	128KB	No limit
Minimum Storage Duration	No minimum duration	30 Days	90 Days
First Byte Latency	milliseconds	milliseconds	4 hours
Retrieval Fee	No Fee	per GB retrieved	per GB retrieved

How the data gets transferred in S3?

Besides traditional transfer practices that is over the internet, AWS has 2 more ways to provide data transfer securely and at a faster rate:

- Transfer Acceleration
- Snowball



Transfer Acceleration enables fast, easy and secure transfers over long distances by exploiting Amazon's CloudFront edge technology.

CloudFront is a caching service by AWS, in which the data from client site gets transferred to the nearest edge location and from there the data is routed to your AWS S3 bucket over an optimized network path.

The Snowball is a way of transferring your data physically. In this Amazon sends an equipment to your premises, on which you can load the data. It has a kindle attached to it which has your shipping address when it is shipped from Amazon. When data transfer is complete on the Snowball, kindle changes the shipping address back to the AWS headquarters where the Snowball has to be sent.

The Snowball is ideal for customers who have large batches of data move. The average turnaround time for Snowball is 5-7 days, in the same time Transfer Acceleration can transfer up to 75 TB of data on a dedicated 1Gbps line. So, depending on the use case, a customer can decide. Obviously, there will be some cost around it, let's look at the overall costing around S3.

What are the Use cases of S3?

- Asset storage and CDN
- Data storage

- Static site
- Backups
- Mobile storage backend
- File Distribution

What are the advantages of S3?

- **Scalability:** The amount of storage & bandwidth you need scale as you like without any configuration changes needed.
- **Availability,** speed throughput, capacity and robustness are not affected even if you gain 10,000 users overnight
- **Unlimited storage.** You **Pay as you go.**
- **Inexpensive and no capital outlay.** Great for startups.
- **Data is accessible from any location**
- Since it is based on the Amazon Infrastructure, it is probably **more reliable** than other cheap data storage providers.

How about S3 Pricing?

- AWS S3 is affordable and flexible in its costing.
- There is no minimum fee to use S3. It works on Pay Per Use, meaning, you only pay what you use.
- Charges for using S3 is based on the location of your buckets
- You are billed according to storage (average), data transfer in and out and the number of requests per month
- You can view your current charges incurred almost immediately on the S3 portal
- Pricing in North Virginia region

Storage/month	Standard Storage	Standard – Infrequent Access Storage	Glacier Storage
First 1 TB / month	\$0.0300 per GB	\$0.0125 per GB	\$0.007 per GB
Next 49 TB / month	\$0.0295 per GB	\$0.0125 per GB	\$0.007 per GB
Next 450 TB / month	\$0.0295 per GB	\$0.0125 per GB	\$0.007 per GB
Next 500 TB / month	\$0.0285 per GB	\$0.0125 per GB	\$0.007 per GB

Cross Region Replication is billed in the following way:

If you replicate 1,000 1 GB objects (1,000 GB) between regions you will incur a request charge of \$0.005 (1,000 requests x \$0.005 per 1,000 requests) for replicating 1,000 objects and a charge of \$20 (\$0.020 per GB transferred x 1,000 GB) for inter-region data transfer. After replication, the 1,000 GB will incur storage charges based on the destination region.

Snowball, there are 2 variants:

- Snowball 50 TB: 200\$
- Snowball 80 TB: 250\$

This is the fixed service fee that they charge.

Apart from this there are on-site, charges which are exclusive of shipping days, the shipping days are free.

The first 10 on-site days are also free, meaning when the Snowball reaches your premises from then, till the day it is shipped back, they are the on-site days. The day it arrives, and the day it is shipped gets counted as shipping days, therefore are free.

Transfer Acceleration pricing is shown in the following table:

Data Transfer IN to Amazon S3 from the Internet:	
Accelerated by AWS Edge Locations in the United States, Europe, and Japan	\$0.04/GB
Accelerated by all other AWS Edge Locations	\$0.08/GB
Data Transfer OUT from Amazon S3 to the Internet:	
Accelerated by any AWS Edge Location	\$0.04/GB
Data Transfer between Amazon S3 and another AWS region:	
Accelerated by any AWS Edge Location	\$0.04/GB

How Encryption is done in S3?

- In Transit: SSL/TLS
- At Rest
- Server-Side Encryption
 - S3 Managed Keys - SSE-S3
 - AWS Key Management Service, Managed Keys - SSE-KMS
 - Server-Side Encryption with Customer Provided Keys - SSE-C
- Client-Side Encryption

What are the different Storage Gateway on S3?

- File Gateway - For flat files, stored directly on S3
- Volume Gateway
 - Stored Volumes - Entire Dataset is stored on site and is asynchronously backed up to S3
 - Cached Volumes - Entire Dataset is stored on S3 and the most frequently accessed data is cached on site
- Gateway Virtual Tape Library (VTL)
 - Used for backup and uses popular backup applications like NetBackup, Backup Exec, Veeam etc.,

What are the important features of Amazon S3?

Some of the important features of Amazon S3 are as follows:

- Amazon S3 provides unlimited storage for files.
- File size in Amazon S3 can vary from 0 Bytes to 5 Terabytes.
- We have store files in Buckets in Amazon S3.
- In Amazon S3, names of buckets have to be unique globally. Amazon S3 is Object Based storage.

What is the maximum length of a file-name in S3?

Names are the object keys. The name for a key is a sequence of Unicode characters whose UTF-8 encoding is at most 1024 bytes long.

What is the scale of durability in Amazon S3?

Amazon S3 supports durability at the scale of 99.999999999% of time. This is 9 nines after decimal.

How can you check the disk space used by S3 bucket?

We can use s3cmd utility for this purpose. We can run s3cmd du command for this. We can also pass the bucket name as an argument to this command.

What are the Consistency levels supported by Amazon S3?

Amazon S3 supports Read after Write consistency when we create a new object by PUT. It means as soon as we Write a new object, we can access it. Amazon S3 supports Eventual Consistency when we overwrite an existing object by PUT. Eventual Consistency means that the effect of overwrite will not be immediate but will happen after some time. For deletion of an object, Amazon S3 supports Eventual Consistency after DELETE.

How can you send request to Amazon S3?

Amazon S3 is a REST service, you can send request by using the REST API or the AWS SDK wrapper libraries that wrap the underlying Amazon S3 REST API.

Mention what is the difference between Amazon S3 and EC2?

The difference between EC2 and Amazon S3 is that

EC2

It is a cloud web service used for hosting your application

It is like a huge computer machine which can run either Linux or Windows and can handle application like PHP, Python, Apache or any databases

S3

It is a data storage system where any amount of data can be stored

It has a REST interface and uses secure HMAC-SHA1 authentication keys

How many buckets can you create in AWS by default?

By default, you can create up to 100 buckets in each of your AWS accounts.

What is the command to copy all files from a S3 bucket to another bucket?

We can use `s3cmd` for this purpose. The command would be as follows: `s3cmd sync s3://source/foo/bucket/ s3://destination/foo/bucket/`

How will you upload a file greater than 100 megabytes in Amazon S3?

Amazon S3 supports storing objects or files up to 5 terabytes. To upload a file greater than 100 megabytes, we have to use Multipart upload utility from AWS.

By using Multipart upload we can upload a large file in multiple parts. Each part will be independently uploaded. It doesn't matter in what order each part is uploaded.

It even supports uploading these parts in parallel to decrease overall time. Once all the parts are uploaded, this utility makes these as one single object or file from which the parts were created.

What happens to an Object when we delete it from Amazon S3?

Amazon S3 provides DELETE API to delete an object.

If the bucket in which the object exists is version controlled, then we can specify the version of the object that we want to delete. The other versions of the Object still exist within the bucket.

If we do not specify the version, and just pass the key name, Amazon S3 will delete the object and return the version id. And the object will not appear on the bucket. In case the bucket is Multi-factor authentication (MFA) enabled, then the DELETE request will fail if we do not specify a MFA token.

Mention what are the differences between Amazon S3 and EC2?

S3: Amazon S3 is just a storage service, typically used to store large binary files. Amazon also has other storage and database services, like RDS for relational databases and DynamoDB for NoSQL.

EC2: An EC2 instance is like a remote computer running Windows or Linux and on which you can install whatever software you want, including a Web server running PHP code and a database server.

How many buckets can you create in AWS by default?

By default, you can create up to 100 buckets in each of your AWS accounts.

How step you follow to make 10,000 files as public in S3?

I will generate a bucket policy which gives access to all the files in the bucket. The bucket policy can be added to a bucket through AWS console.

```
{
  "Id": "...",
  "Statement": [ {
    "Sid": "...",
    "Action": [
      "s3:GetObject"
    ],
    "Effect": "Allow",
    "Resource": "arn:aws:s3:::bucket/*",
    "Principal": {
      "AWS": [ "*" ]
    }
  }
]
```

How do you see how much disk space is using by S3 bucket?

s3cmd can show you this by running s3cmd du, optionally passing the bucket name as an argument.

Write down the command you will use to copy all files from one S3 bucket to another with s3cmd?

```
s3cmd sync s3://from/this/bucket/ s3://to/this/bucket/
```

How many objects you can put in a S3 bucket? is there a limit to the number of objects I can put in an S3 bucket?

Write, read, and delete objects containing from 1 byte to 5 terabytes of data each. The number of objects you can store is unlimited.

How to delete files recursively from an S3 bucket?

```
aws s3 rm --recursive s3://your_bucket_name/fool
```


Or delete everything under the bucket:

```
aws s3 rm --recursive s3://your_bucket_name
```

If what you want is to actually delete the bucket, there is one-step shortcut:

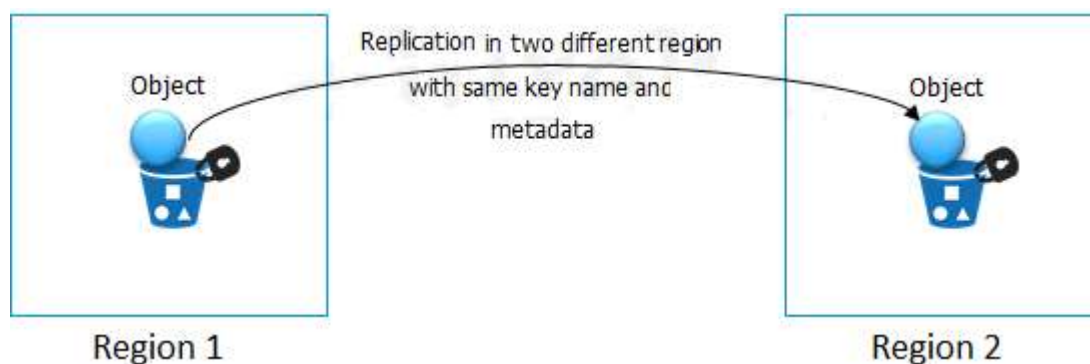
```
aws s3 rb --force s3://your_bucket_name
```

Can we disable versioning on a version-enabled bucket in Amazon S3?

No, we cannot disable versioning on a version-enabled bucket in Amazon S3. We can just suspend the versioning on a bucket in S3. Once we suspend versioning, Amazon S3 will stop creating new versions of the object. It just stores the object with null version ID. On overwriting an existing object, it just replaces the object with null version ID. So any existing versions of the object still remain in the bucket. But there will be no more new versions of the same object except for the null version ID object.

What are the use cases of Cross Region Replication Amazon S3?

We can use Cross Region Replication Amazon S3 to make copies of an object across buckets in different AWS Regions. This copying takes place automatically and in an asynchronous mode.



We have to add replication configuration on our source bucket in S3 to make use of Cross Region Replication. It will create exact replicas of the objects from source bucket to destination buckets in different regions.

Some of the main use cases of Cross Region Replication are as follows: **Compliance:** Sometimes there are laws/regulatory requirements that ask for storing data at farther geographic locations. This kind of compliance can be achieved by using AWS Regions that are spread across the world. **Failover:** At times, we want to minimize the probability of system failure due to complete blackout in a region.

We can use Cross-Region Replication in such a scenario. **Latency:** In case we are serving multiple geographies, it makes sense to replicate objects in the geographical Regions that are closer to end customer. This helps in reducing the latency.

Can we do Cross Region replication in Amazon S3 without enabling versioning on a bucket?

No, we have to enable versioning on a bucket to perform Cross Region Replication.

What are the different types of actions in Object Lifecycle Management in Amazon S3?

There are mainly two types of Object Lifecycle Management actions in Amazon S3. Transition Actions: These actions define the state when an Object transitions from one storage class to another storage class. E.g. a new object may transition to STANDARD_IA (infrequent access) class after 60 days of creation. And it can transition to GLACIER after 180 days of creation. Expiration Actions: These actions specify what happens when an Object expires. We can ask S3 to delete an object completely on expiration.

What are the security mechanisms available in Amazon S3?

Amazon S3 is a very secure storage service. Some of the main security mechanisms available in Amazon S3 are as follows: -

Access: When we create a bucket or an object, only the owner gets the access to the bucket and objects.

Authentication: Amazon S3 also support user authentication to control who has access to a specific object or bucket.

Access Control List: We can create Access Control Lists (ACL) to provide selective permissions to users and groups.

HTTPS: Amazon S3 also supports HTTPS protocol to securely upload and download data from cloud.

Encryption: We can also use Server-Side Encryption (SSE) in Amazon S3 to encrypt data.

You need to configure an Amazon S3 bucket to serve static assets for your public-facing web application. Which method will ensure that all objects uploaded to the bucket are set to public read?

- A. Set permissions on the object to public read during upload.
- B. Configure the bucket policy to set all objects to public read.
- C. Use AWS Identity and Access Management roles to set the bucket to public read.
- D. Amazon S3 objects default to public read, so no action is needed.

Answer B

Explanation: Rather than making changes to every object, its better to set the policy for the whole bucket. IAM is used to give more granular permissions, since this is a website, all objects would be public by default.

A customer wants to leverage Amazon Simple Storage Service (S3) and Amazon Glacier as part of their backup and archive infrastructure. The customer plans to use third-party software to support this integration. Which approach will limit the access of the third party software to only the Amazon S3 bucket named “company-backup”?

- A. A custom bucket policy limited to the Amazon S3 API in three Amazon Glacier archive “company-backup”
- B. A custom bucket policy limited to the Amazon S3 API in “company-backup”
- C. A custom IAM user policy limited to the Amazon S3 API for the Amazon Glacier archive “company-backup”.
- D. A custom IAM user policy limited to the Amazon S3 API in “company-backup”.

Answer D

Explanation: Taking queue from the previous questions, this use case involves more granular permissions, hence IAM would be used here.

Can S3 be used with EC2 instances, if yes, how?

Yes, it can be used for instances with root devices backed by local instance storage. By using Amazon S3, developers have access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure that Amazon uses to run its own global network of web sites. In order to execute systems in the Amazon EC2 environment, developers use the tools provided to load their Amazon Machine Images (AMIs) into Amazon S3 and to move them between Amazon S3 and Amazon EC2. Another use case could be for websites hosted on EC2 to load their static content from S3.

A customer implemented AWS Storage Gateway with a gateway-cached volume at their main office. An event takes the link between the main and branch office offline. Which methods will enable the branch office to access their data?

- A. Restore by implementing a lifecycle policy on the Amazon S3 bucket.
- B. Make an Amazon Glacier Restore API call to load the files into another Amazon S3 bucket within four to six hours.
- C. Launch a new AWS Storage Gateway instance AMI in Amazon EC2 and restore from a gateway snapshot.
- D. Create an Amazon EBS volume from a gateway snapshot and mount it to an Amazon EC2 instance.

Answer C

Explanation: The fastest way to do it would be launching a new storage gateway instance. Why? Since time is the key factor which drives every business, troubleshooting this problem will take more time. Rather than we can just restore the previous working state of the storage gateway on a new instance.

When you need to move data over long distances using the internet, for instance across countries or continents to your Amazon S3 bucket, which method or service will you use?

- A. Amazon Glacier
- B. Amazon CloudFront
- C. Amazon Transfer Acceleration
- D. Amazon Snowball

Answer C

Explanation: You would not use Snowball, because for now, the snowball service does not support cross region data transfer, and since, we are transferring across countries, Snowball cannot be used.

Transfer Acceleration shall be the right choice here as it throttles your data transfer with the use of optimized network paths and Amazon's content delivery network upto 300% compared to normal data transfer speed.

How can you speed up data transfer in Snowball?

The data transfer can be increased in the following way:

By performing multiple copy operations at one time i.e. if the workstation is powerful enough, you can initiate multiple cp commands each from different terminals, on the same Snowball device.

Copying from multiple workstations to the same snowball.

Transferring large files or by creating a batch of small file, this will reduce the encryption overhead.

Eliminating unnecessary hops i.e. make a setup where the source machine(s) and the snowball are the only machines active on the switch being used, this can hugely improve performance.

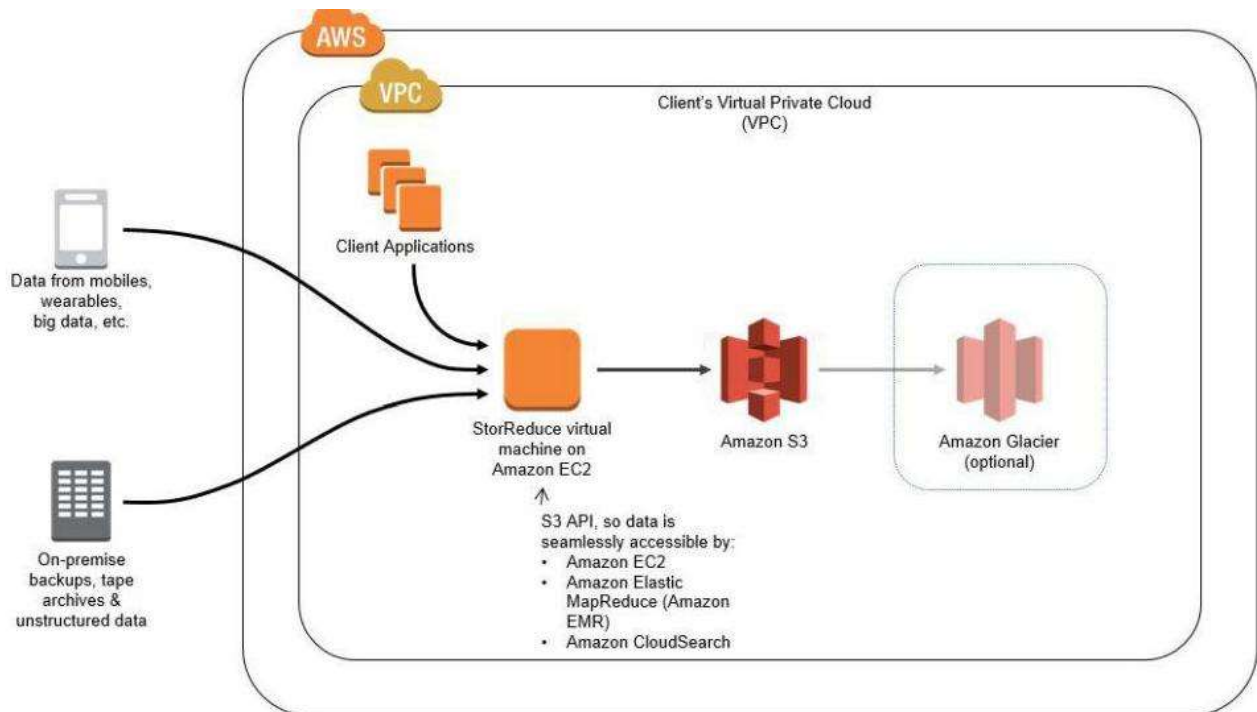


Glacier

Share the S3 Configuration Step by Step?

To Configure and use [AWS Glacier Service](#)

Topology



Pre-requisites

User should have AWS account, IAM user with AmazonGlacierFullAccess Policy

To Configure Glacier with following task

Transfer files from S3 to Glacier

Note: Amazon does not allow files to be directly loaded on Glacier

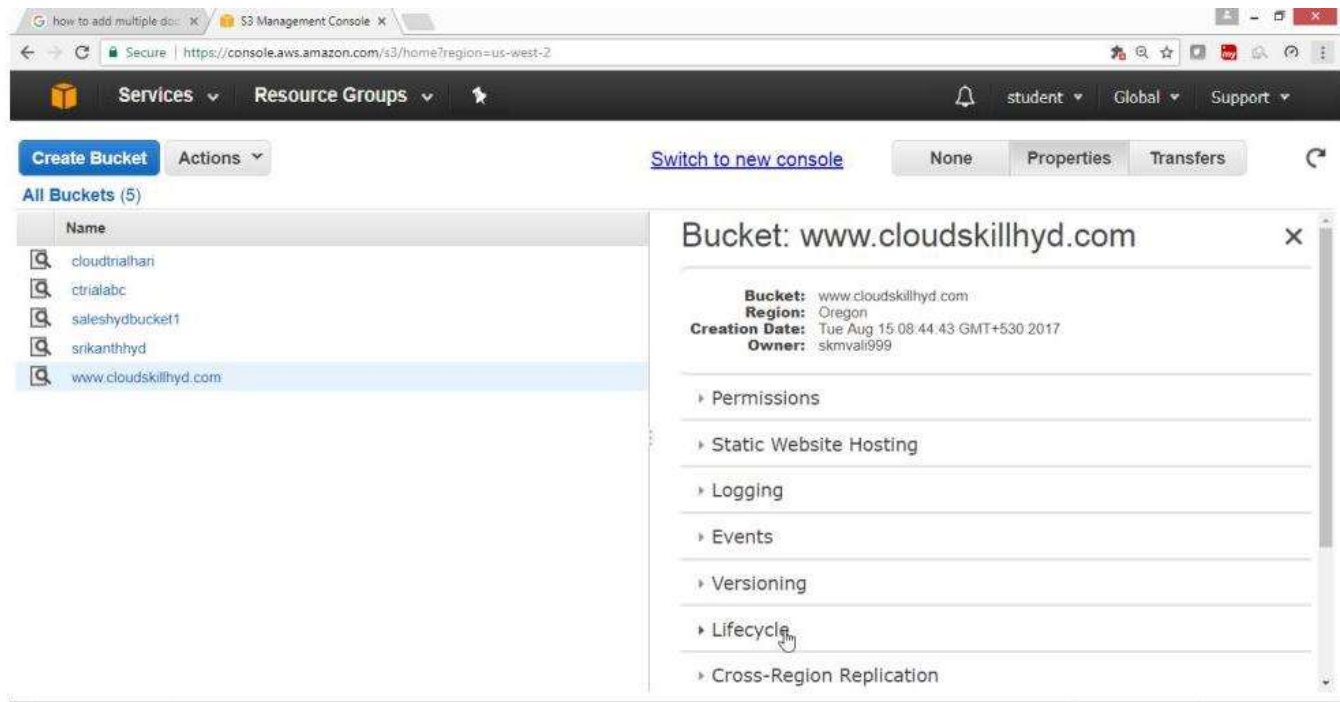
Use S3 or third-party tools to archive or restore

Step-1) Using S3 bucket & S3 life cycle permission to archive in glacier

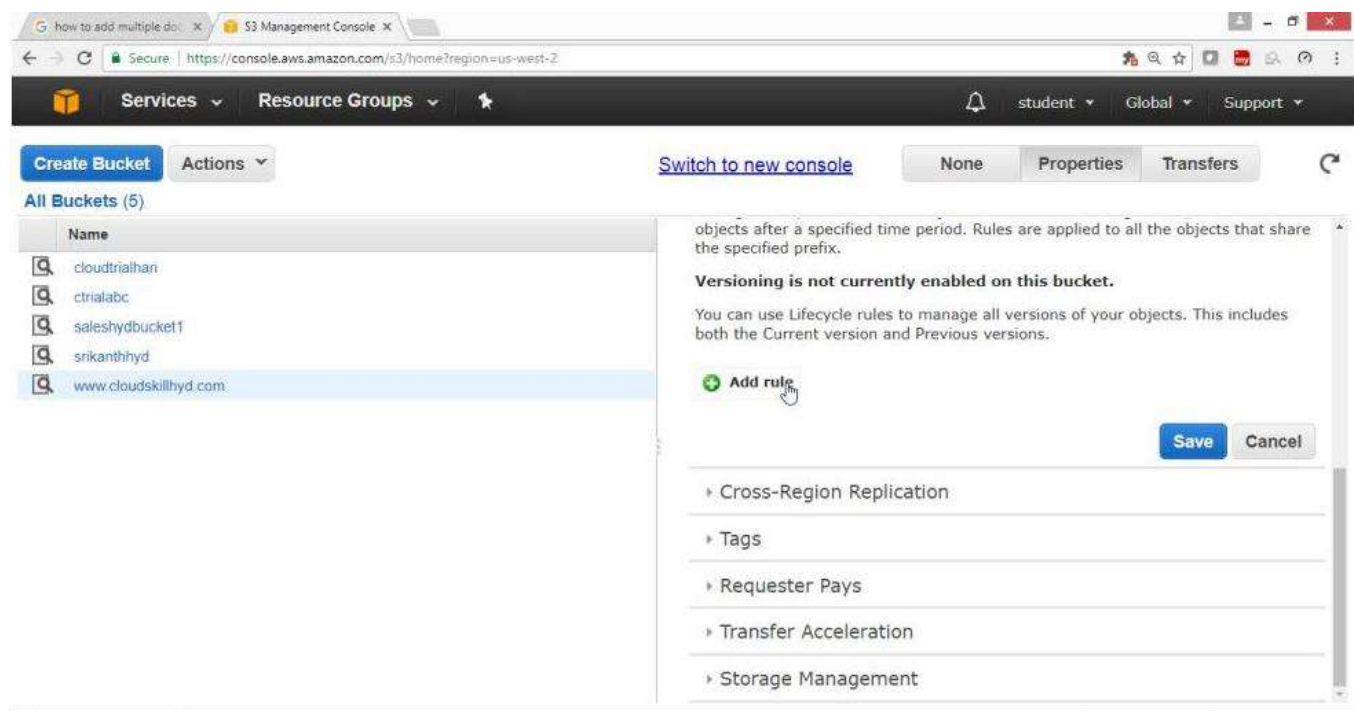
Select S3 bucket

Go to properties

Click on Lifecycle



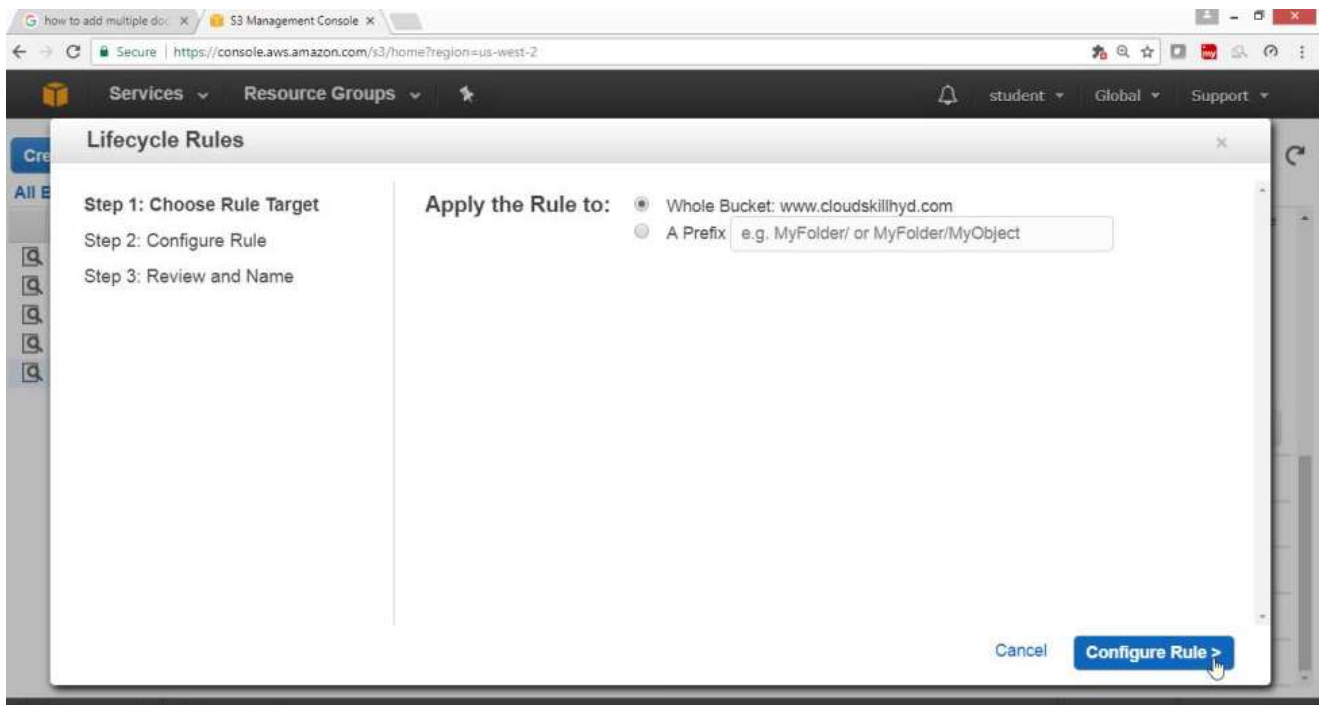
Click on **Add rule**



Under Lifecycle Rules

Select Choose **Rule Target**

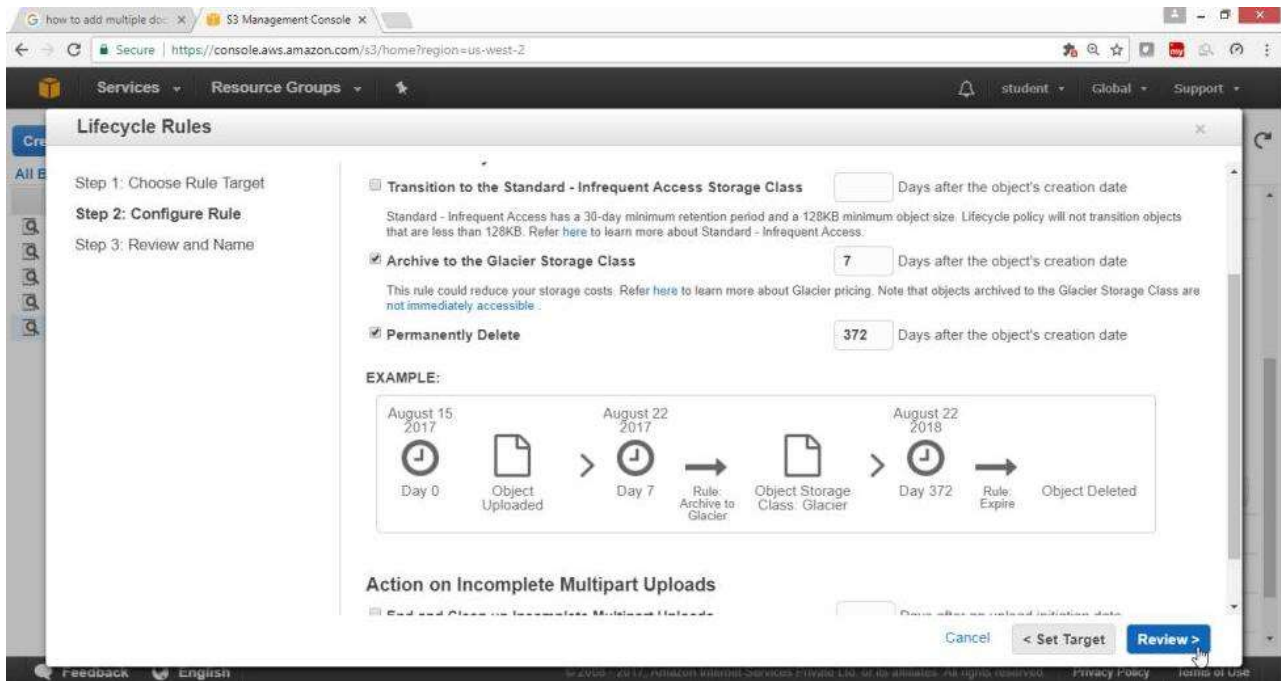
Apply the Rule to -> **Whole Bucket**



Select checkbox Archive to the Glacier Storage Class -> 7

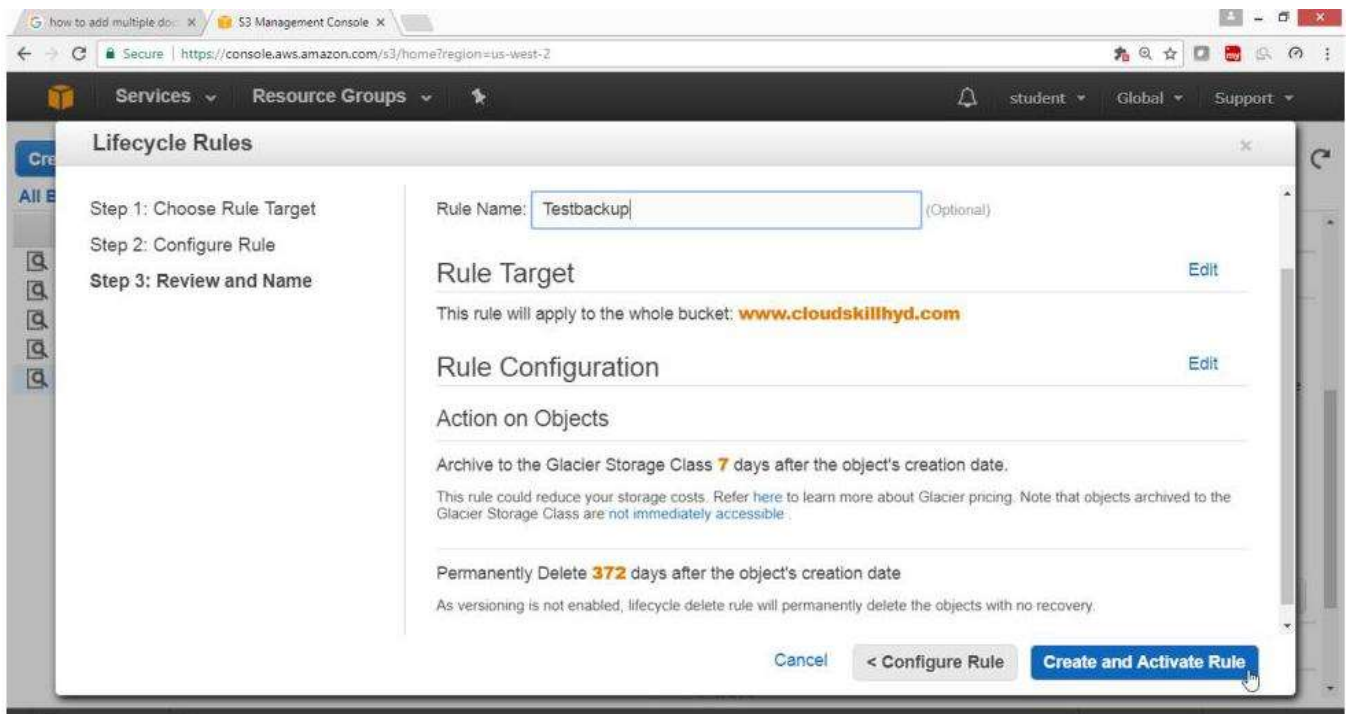
Select the check box Permanently Delete -> 372

Click on Review

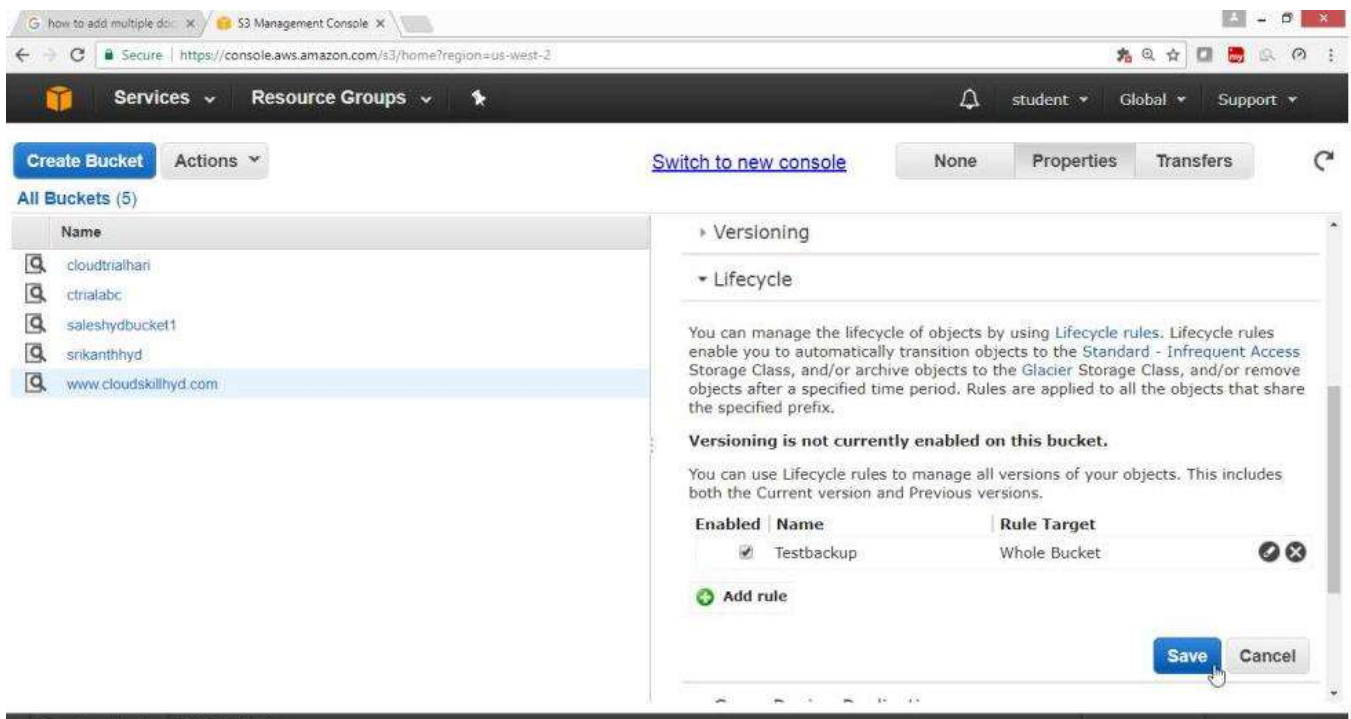


Provide Rule Name -> Testbackup

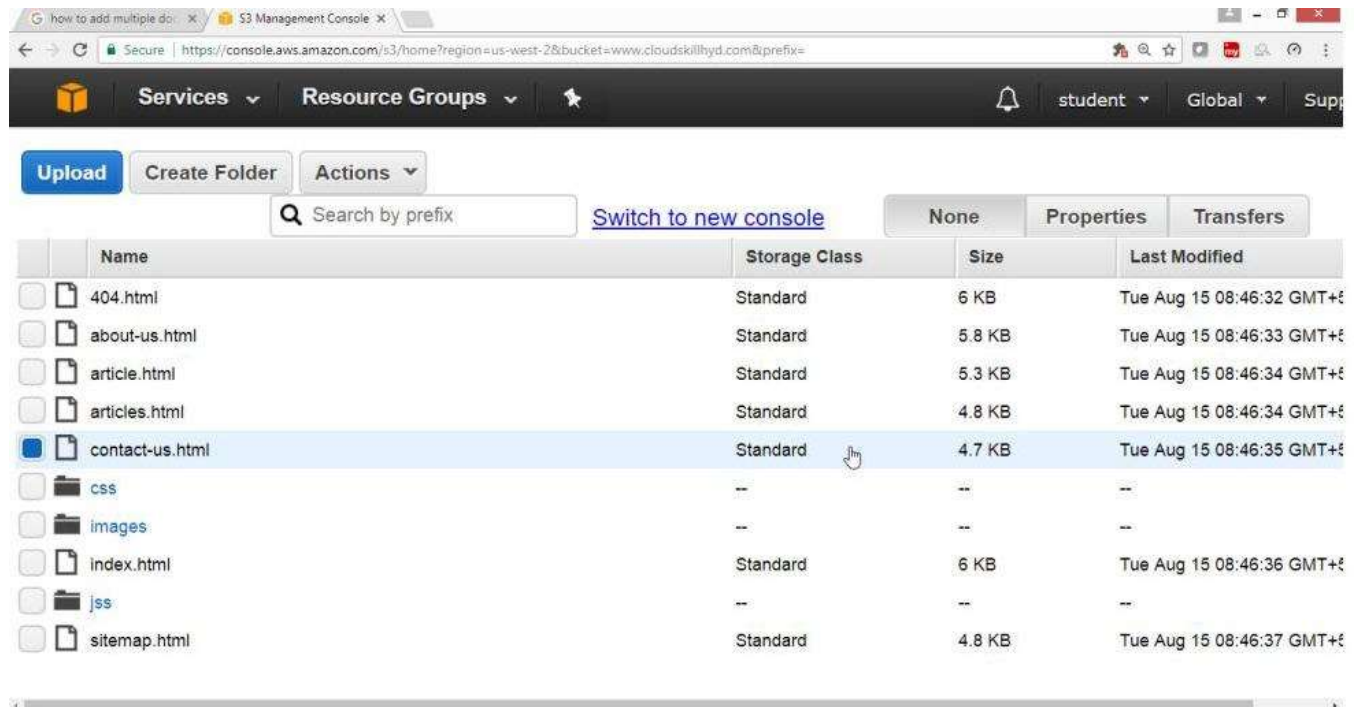
Click on "Create and Activate Rule" button



Click on "Save" button



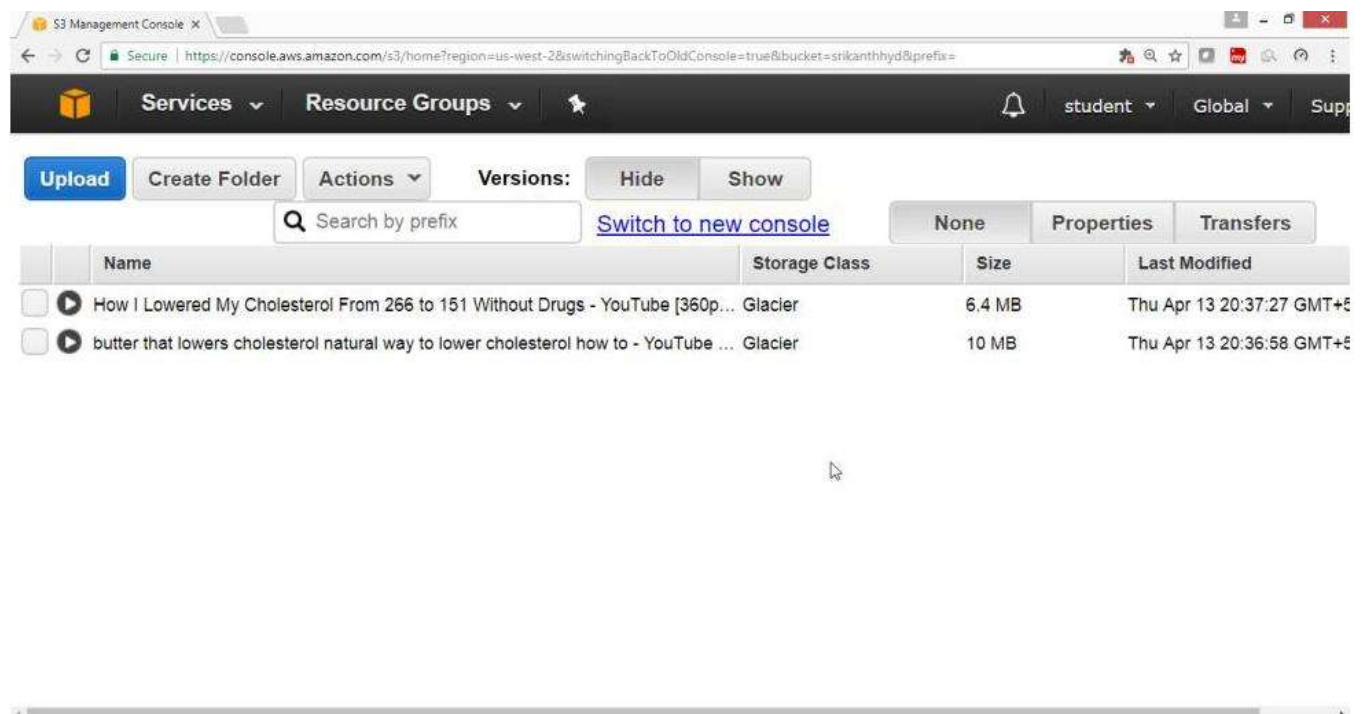
Verify Storage Class is Standard



The screenshot shows the AWS S3 Management Console interface. At the top, there's a navigation bar with 'Services', 'Resource Groups', and a user profile 'student'. Below the navigation bar, there are buttons for 'Upload', 'Create Folder', and 'Actions'. A search bar with the placeholder 'Search by prefix' and a link 'Switch to new console' are also present. The main content area displays a table of objects in the bucket. The table has columns for 'Name', 'Storage Class', 'Size', and 'Last Modified'. The 'Storage Class' column for all listed files is 'Standard'.

Name	Storage Class	Size	Last Modified
404.html	Standard	6 KB	Tue Aug 15 08:46:32 GMT+5
about-us.html	Standard	5.8 KB	Tue Aug 15 08:46:33 GMT+5
article.html	Standard	5.3 KB	Tue Aug 15 08:46:34 GMT+5
articles.html	Standard	4.8 KB	Tue Aug 15 08:46:34 GMT+5
contact-us.html	Standard	4.7 KB	Tue Aug 15 08:46:35 GMT+5
css	--	--	--
images	--	--	--
index.html	Standard	6 KB	Tue Aug 15 08:46:36 GMT+5
js	--	--	--
sitemap.html	Standard	4.8 KB	Tue Aug 15 08:46:37 GMT+5

Verify Once the file goes to Glacier then Storage Class is Glacier

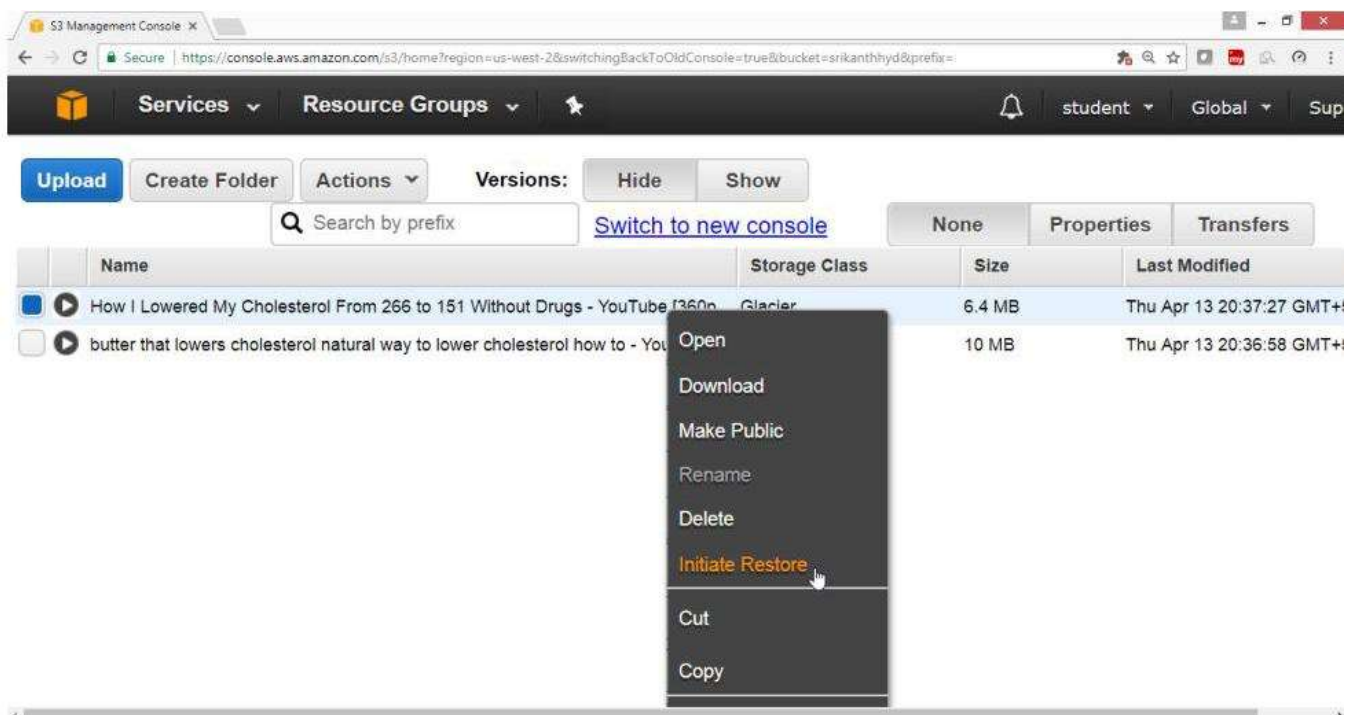


The screenshot shows the AWS S3 Management Console interface for a different bucket. The navigation bar is the same. Below the navigation bar, there are buttons for 'Upload', 'Create Folder', and 'Actions'. A search bar with the placeholder 'Search by prefix' and a link 'Switch to new console' are also present. The main content area displays a table of objects in the bucket. The table has columns for 'Name', 'Storage Class', 'Size', and 'Last Modified'. The 'Storage Class' column for all listed files is 'Glacier'.

Name	Storage Class	Size	Last Modified
How I Lowered My Cholesterol From 266 to 151 Without Drugs - YouTube [360p...]	Glacier	6.4 MB	Thu Apr 13 20:37:27 GMT+5
butter that lowers cholesterol natural way to lower cholesterol how to - YouTube ...	Glacier	10 MB	Thu Apr 13 20:36:58 GMT+5

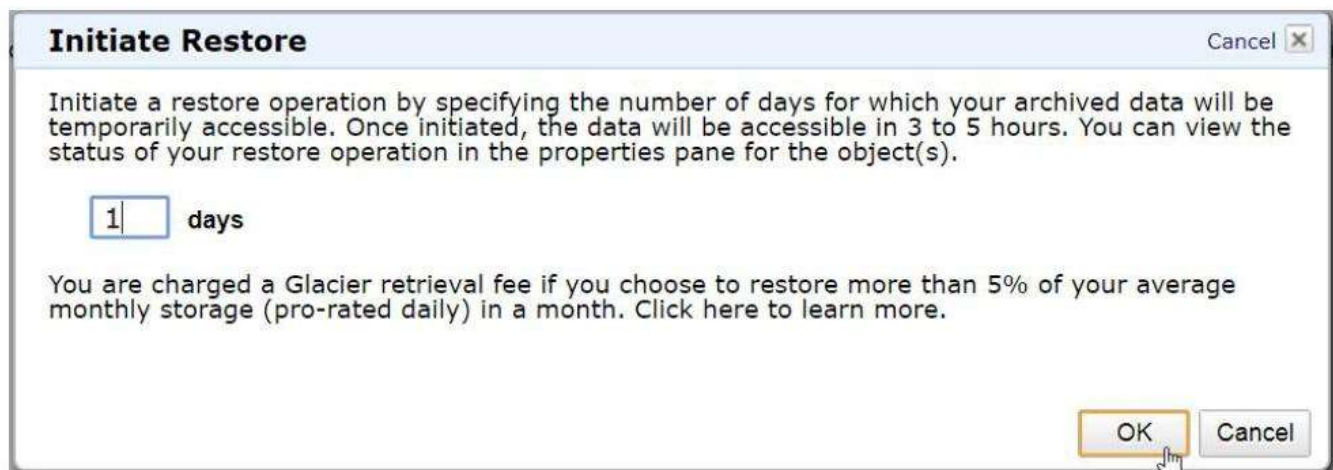
To Restore go to the bucket select the file

Right click and select Initiate Restore



Provide number of days ->1

Click on OK



Verify

File will get restored after 1 Day

Storage class will become Standard

What is the use of Amazon Glacier?

Amazon Glacier is an extremely low-cost cloud-based storage service provided by Amazon. We mainly use Amazon Glacier for long-term backup purpose. Amazon Glacier can be used for storing data archives for months, years or even decades.

It can also be used for long term immutable storage based on regulatory and archiving requirements. It provides Vault Lock support for this purpose. In this option, we write once but can read many times same data. One use case is for storing certificates that can be issued only once and only the original person keeps the main copy.

Suppose that you are working with a customer who has 10 TB of archival data that they want to migrate to glacier. The customer has a 1-Mbps connection to the internet. Which service or feature provides the fastest method of getting data in to Amazon Glacier?

AWS Import | Export

I created a key in Oregon region to encrypt my data in North Virginia region for security purposes. I added two users to the key and an external AWS account. I wanted to encrypt an object in S3, so when I tried, the key that I just created was not listed. What could be the reason?

External aws accounts are not supported.

AWS S3 cannot be integrated KMS.

The Key should be in the same region.

New keys take some time to reflect in the list.

Answer C.

Explanation: The key created and the data to be encrypted should be in the same region. Hence the approach taken here to secure the data is incorrect.



Storage Gateway

What are the benefits of AWS Storage Gateway?

We can use AWS Storage Gateway (ASG) service to connect our local infrastructure for files etc with Amazon cloud services for storage. Some of the main benefits of AWS Storage Gateway are as follows:

Local Use: We can use ASG to integrate our data in multiple Amazon Storage Services like- S3, Glacier etc with our local systems. We can continue to use our local systems seamlessly.

Performance: ASG provides better performance by caching data in local disks. Though data stays in cloud, but the performance we get is similar to that of local storage.

Easy to use: ASG provides a virtual machine to use it by an easy to use interface. There is no need to install any client or provision rack space for using ASG. These virtual machines can work in local system as well as in AWS.

Scale: We get the storage at a very high scale with ASG. Since backend in ASG is Amazon cloud, it can handle large amounts of workloads and storage needs.

Optimized Transfer: ASG performs many optimizations, due to which only the changes to data are transferred. This helps in minimizing the use of bandwidth.

What are the main use cases for AWS Storage Gateway?

AWS Storage Gateway (ASG) is very versatile in its usage. It solves a variety of problems at an enterprise.

Some of the main use cases of ASG are as follows:

Backup systems: We can use ASG to create backup systems. From local storage data can be backed up into cloud services of AWS. On demand, we can also restore the data from this backup solution. It is a replacement for Tape based backup systems.

Variable Storage: With ASG, we can grow or shrink our Storage as per our needs. There is no need to add racks, disks etc to expand our storage systems. We can manage the fluctuations in our storage needs gracefully by using ASG.

Disaster Recovery: We can also use ASG for disaster recovery mechanism. We can create snapshots of our local volumes in Amazon EBS. In case of a local disaster we can use our applications in cloud and recover from the snapshots created in EBS. **Hybrid Cloud:** At times we want to use our local applications with cloud services. ASG helps in implementing Hybrid cloud solutions in which we can utilize cloud storage services with us on premises local applications.



Snowball

What is AWS Snowball?

AWS provides a very useful service called Snowball for transporting very large amounts of data at the scale of petabytes. With Snowball, we can securely transfer data without any network cost. It is a physical data transfer solution to store data in AWS cloud.

Once we create a Snowball job in AWS console, Amazon ships a physical storage device to our location. We can copy our data to this storage device and ship it back. Amazon services will take the Snowball device and transfer the data to Amazon S3.

- Snowball can
 - Import to S3, Export from S3
- Snowball Edge
- Snowmobile

What is Transfer Acceleration in S3?

You can speed up transfers to S3 using S3 transfer acceleration. This costs extra and has the greatest impact on people who are in faraway location.

What is the purpose of Static Websites in S3?

You can use S3 to host static websites

- Serverless
- Very cheap, scales automatically
- STATIC only, cannot host dynamic sites



Database

Amazon Aurora High Performance Managed Relational Database	Amazon RDS Managed Relational Database Service for MySQL, PostgreSQL, Oracle, SQL Server and MariaDB	Amazon DynamoDB Managed NoSQL Database	
Amazon Elastic Cache In-memory Caching System	AWS Redshift Fast, Simple, Cost-effective Data Warehousing	Amazon Neptune Fully Managed Graph Database Service	
AWS Database Migration Service Migrate Database with Minimal Downtime			



Database

RDS

Database Highlights

AWS Data Types

- RDS - OLTP
 - SQL, MySQL, PostgreSQL, Oracle, Aurora, MariaDB
- DynamoDB - NoSQL
- RedShift – OLAP
- Elastic Cache – In Memory Caching
 - Memcached, Redis

Aurora Scaling

- Two copies of your data are contained in each availability zone, with minimum of 3 availability zones. Six copies of your data.
- Aurora is designed to transparently handle the loss of up to two copies of data without affecting database write availability and up to three copies without affecting read availability.
- Aurora storage is also self-healing. Data blocks and disks are continuously scanned for errors and repaired automatically

Aurora Replica

- 2 Types of Replicas are available
- Aurora Replicas (Currently 15)
- MySQL Read Replicas (Currently 15)

DynamoDB Vs RDS

- DynamoDB offers "Push Button" scaling, meaning that you can scale your database on the fly, without any down time.
- RDS is not so easy and you usually have to use a bigger instance size or to add a read replica

DynamoDB

- Stored on SSD storage

- Spread Across 3 geographically distinct data Centre's
- Eventual Consistent Reads (Default)
- Strongly Consistent Reads

Redshift Configuration

- Single Node (160 GB)
- Multi-Node
 - Leader Node (Manages Client Connections and receives queries)
 - Compute Node (Store Data and Perform queries and computations) up to 128 Compute Nodes

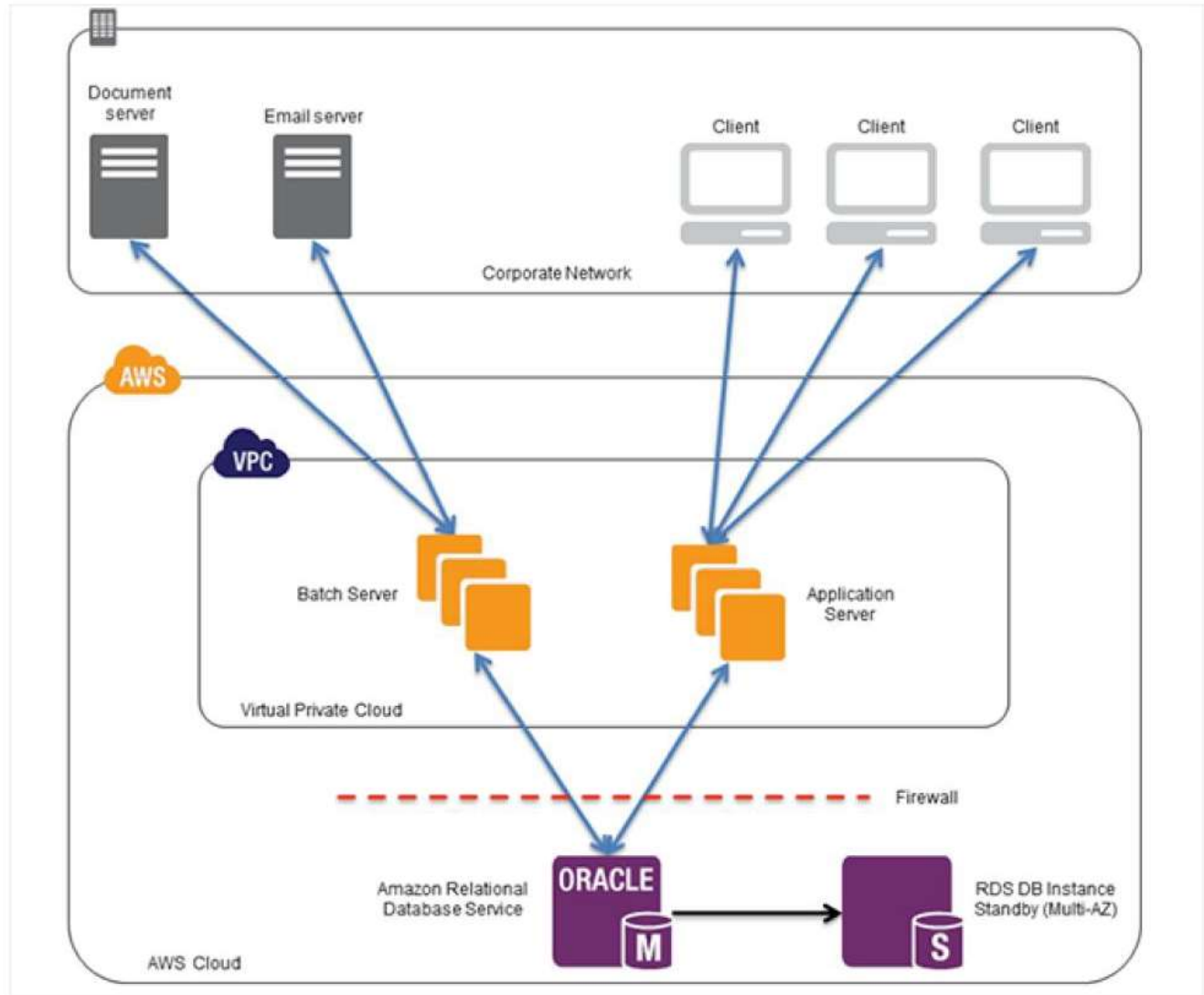
What is Amazon RDS?

RDS stand for **Relational Database Service** is a web service that makes it easier to setup, operate, and scale a relational database in the cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

Share the Amazon RDS Configuration Step by Step?

To Configure [Amazon Relational Database Service](#)

Topology



Pre-requisites

User should have AWS account, or IAM user with AmazonRDSFullAccess

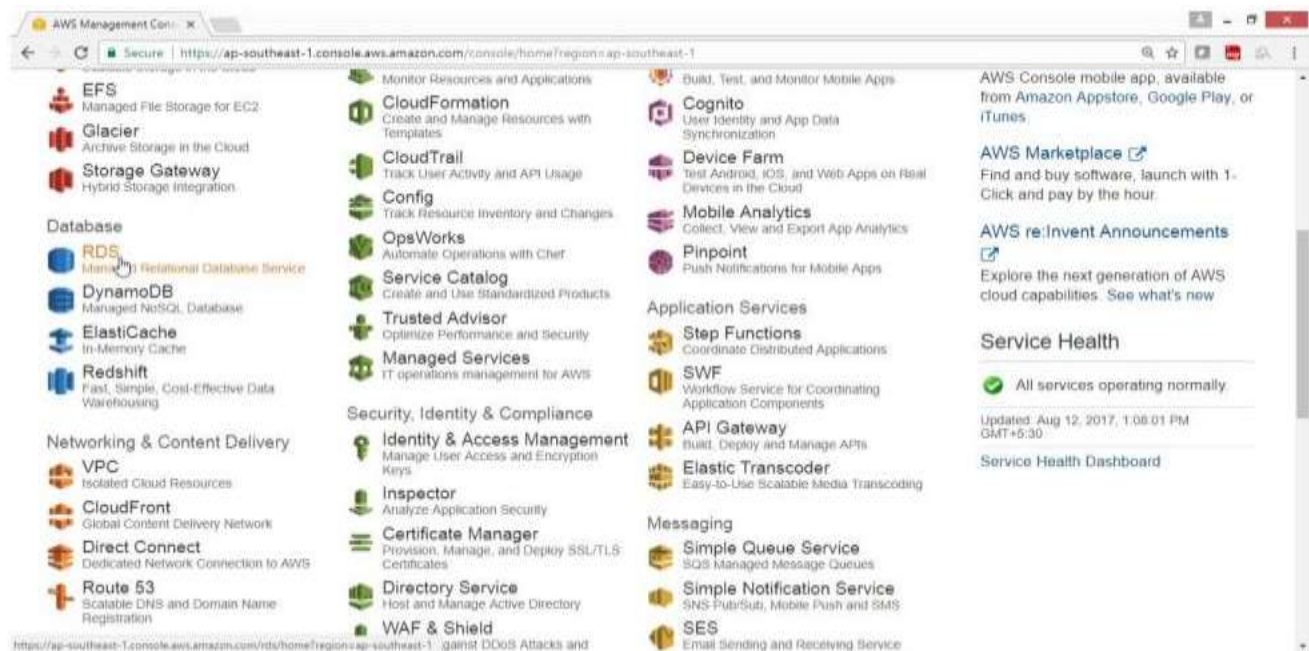
Task

- Create Amazon Relational Database Service
- Verify connection from MySQL client command line tool
- Verify Connection using MySQL Workbench client application

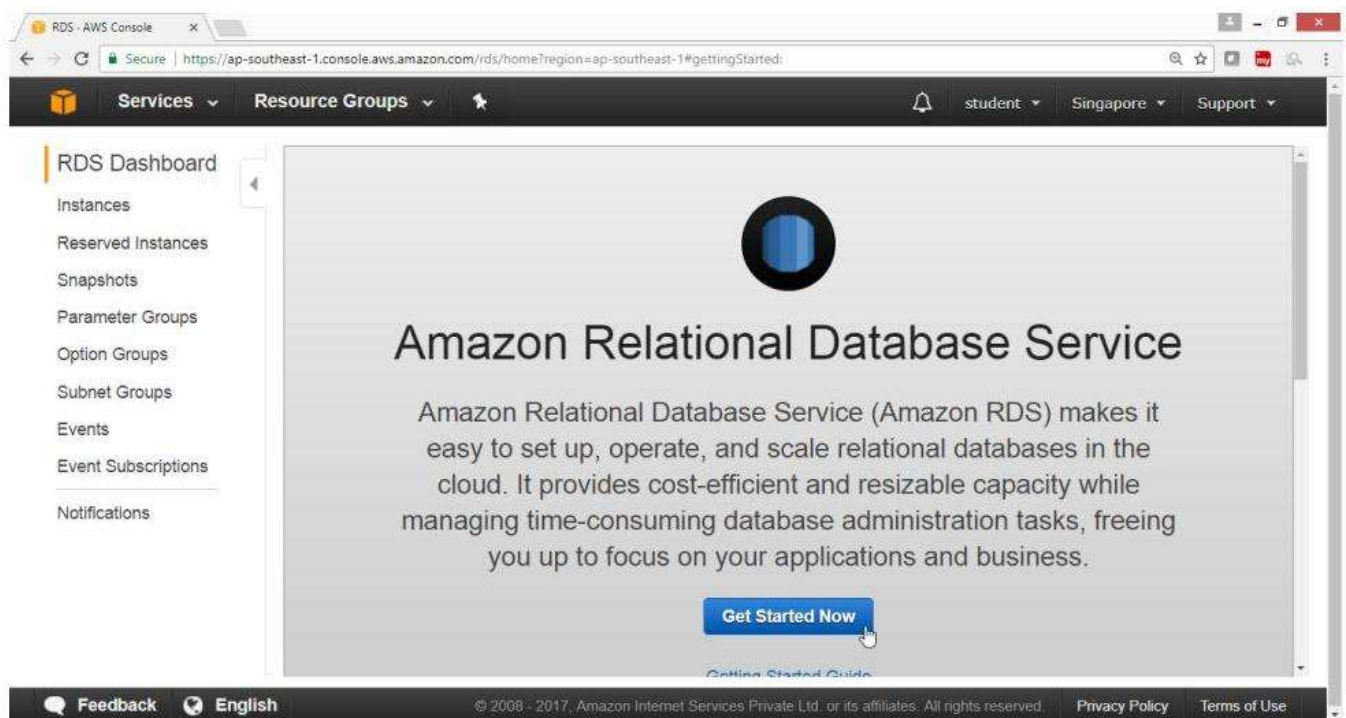
Step-1) To create Amazon Relational Database Service

From the [AWS Console](#)

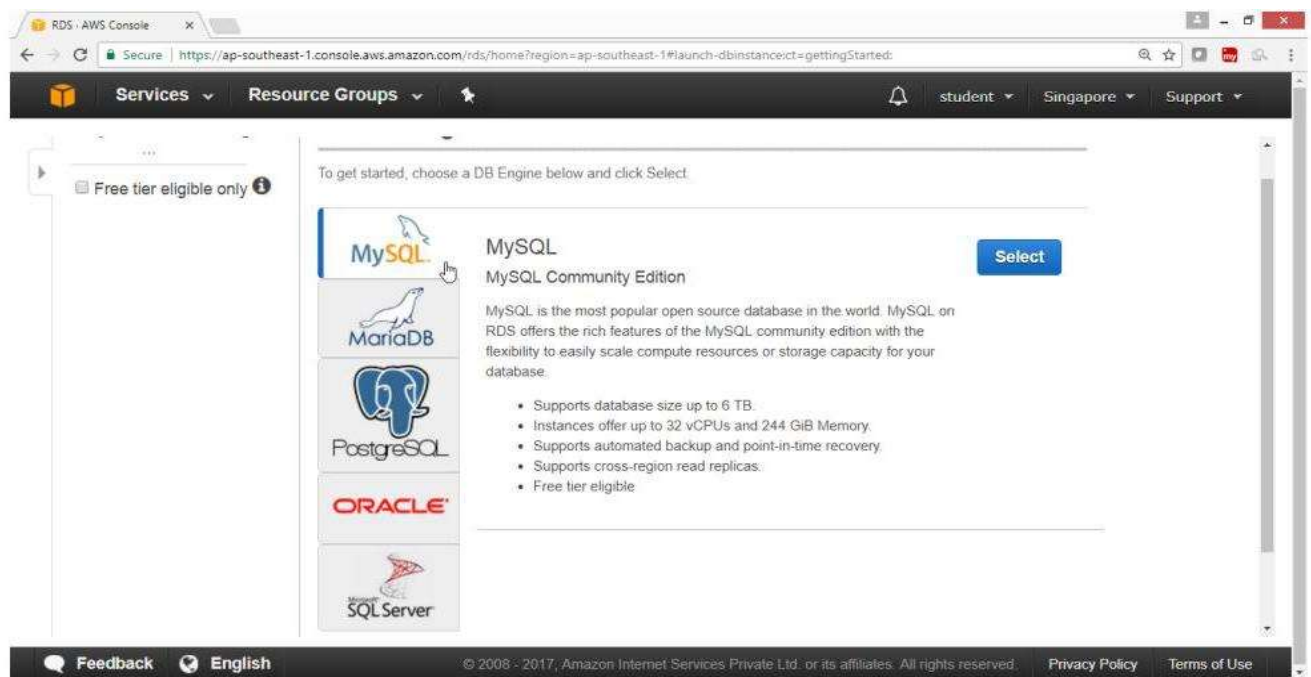
- Select "[Database](#)"
- Click on "[RDS](#)" Service



In "RDS Dashboard", wizard
Click "Get Started Now", button

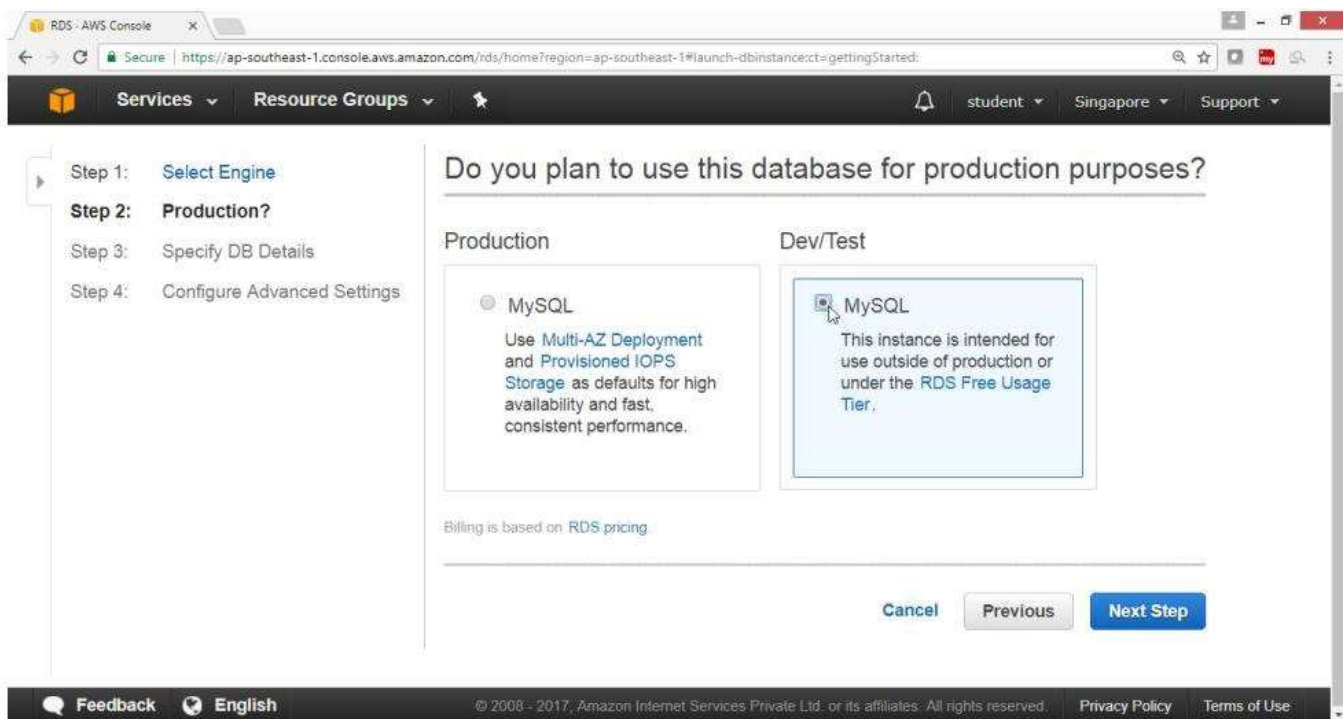


In "Select Engine", Wizard
Click on "MySQL"
Click on "Select" Button



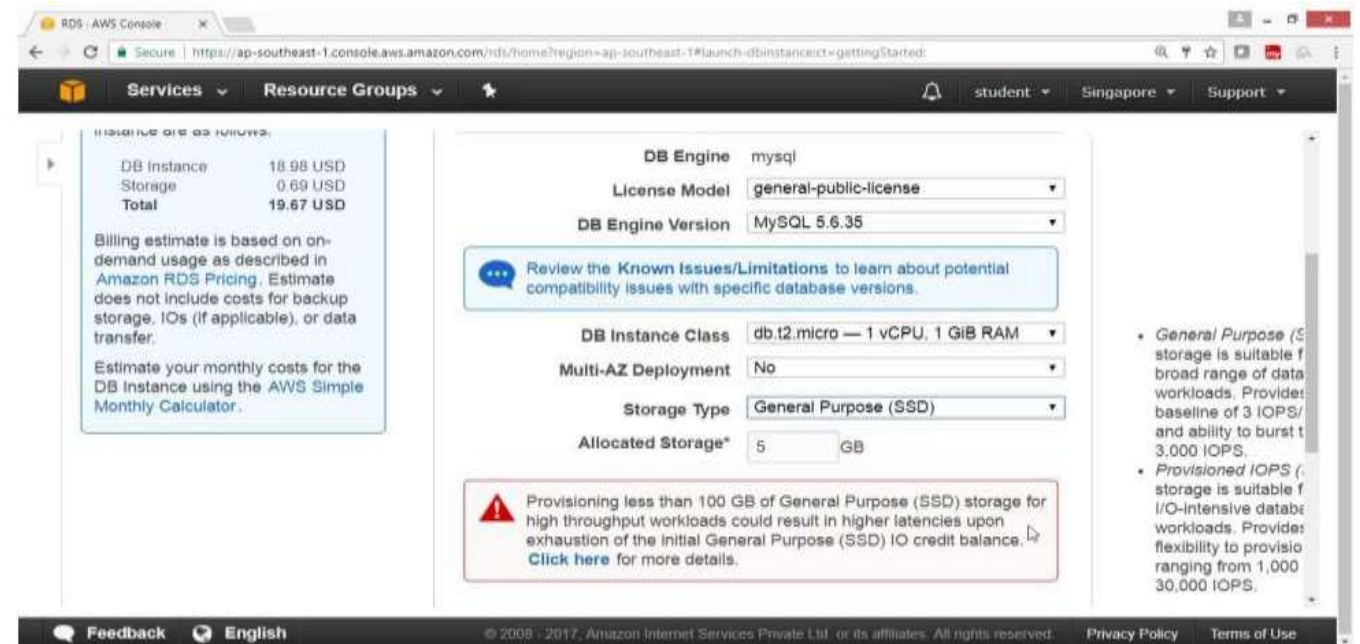
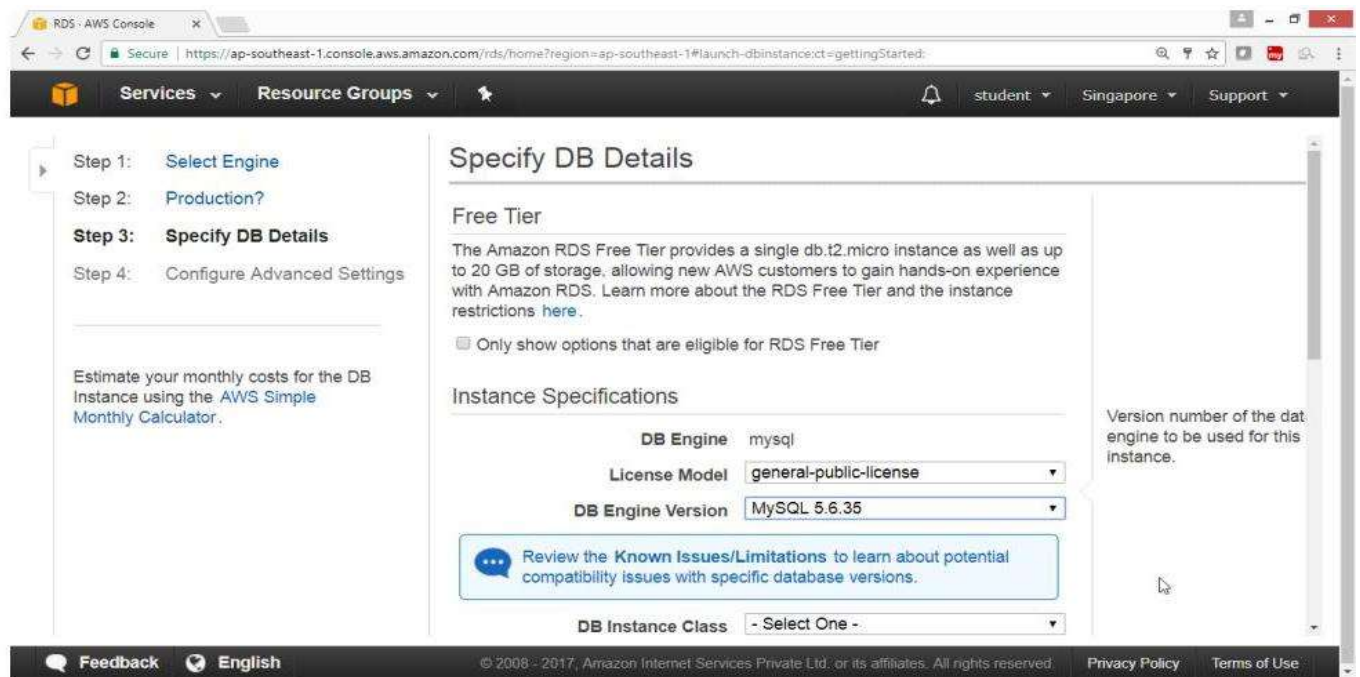
"In Production Wizard"

Select Dev/Test, Choose **MySQL**



In "**Specify DB Details**", wizard provide following values in "**Instance Specifications**"

- For DB Engine -> **MySQL**
- For License Model -> **general-public-license**
- For DB Engine Version -> **5.6.27** (Leave default)
- For DB Instance Class -> **db.t2.micro**
- For Multi-AZ Deployment -> **No**
- For Storage Type -> **General Purpose SSD**
- For Allocated Storage -> **5GB**



"Under Settings"

For Allocated Storage* -> **5 GB**

For DB Instance Identifier -> **rdsdatabase**

For Master Username -> **testuser**

For Master Password* -> *********

For Confirm Password* -> *********

Click on **"Next"** Button

RDS: AWS Console

Secure | https://ap-southeast-1.console.aws.amazon.com/rds/home?region=ap-southeast-1#launch-dbinstance:ct=gettingStarted:

ServicesResource GroupsstudentSingaporeSupport

Provisioning less than 100 GB of General Purpose (SSD) storage for high throughput workloads could result in higher latencies upon exhaustion of the initial General Purpose (SSD) IO credit balance. [Click here](#) for more details.

Settings

DB Instance Identifier*

rdsdatabase

Master Username*

testuser

Master Password*

.....

Confirm Password*

.....

Retype the value you specified for Master Password.

* Required

CancelPreviousNext Step

FeedbackEnglish

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

In **Configure Advanced Settings**, Wizard, Under **Network & Security**

Provide the following Values

- **VPC*** -> **Default VPC**
- **Subnet Group** -> **default**
- **Publicly Accessible** -> **Yes**
- **Availability Zone** -> **No Preference**
- **VPC Security Group(s)** -> **Create new Security Group**

The screenshot shows the AWS RDS console 'Configure Advanced Settings' wizard, Step 4. The left sidebar lists the steps: Step 1: Select Engine, Step 2: Production?, Step 3: Specify DB Details, and Step 4: Configure Advanced Settings. The main content area is titled 'Configure Advanced Settings' and has a tab for 'Network & Security'. The configuration options are as follows:

- VPC***: Default VPC (vpc-ec2fe388)
- Subnet Group**: default
- Publicly Accessible**: Yes
- Availability Zone**: No Preference
- VPC Security Group(s)**: Create new Security Group (selected), default (VPC), launch-wizard-1 (VPC), rds-launch-wizard (VPC)

Below the 'Network & Security' section is the 'Database Options' section, which includes a 'Database Name' input field. On the right side of the 'Network & Security' section, there is a help text box that reads: 'Select the security groups that have rule authorizing connections of the EC2 instances that need to data stored in the DB. By default, security groups not authorize any connections, so you must specify rule instances and device connect to the DB instance. [Learn More](#).' At the bottom of the console, there is a footer with 'Feedback', 'English', '© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.', 'Privacy Policy', and 'Terms of Use'.

Under Database Options

Provide the following values

- Database Name -> **salesdba**
- Database Port -> **3306**
- DB Parameter Group -> **default.mysql5.6**
- Option Group -> **default.mysql5.6**
- Copy Tags To Snapshots -> **leave blank**
- Enable IAM DB Authentication -> **No Preference**
- Enable Encryption -> **No**

Database Options

Database Name:

Note: if no database name is specified then no initial MySQL database will be created on the DB Instance.

Database Port:

DB Parameter Group:

Option Group:

Copy Tags To Snapshots: ☐

Enable IAM DB Authentication:

Enable Encryption:

Backup

Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to detail [here](#).

Specify a string of up to 63 alpha-numeric characters. Do not use spaces or special characters. Define the name give database that Amazon RDS creates when it creates the instance, as in "mydb". Do not specify a database name when you create the instance.

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

"Provider Following Values"

Under "Backup"

- Backup Retention Period -> 7 days
- Backup Window -> No Preference

Under "Monitoring"

- Enable Enhanced Monitoring -> No

Under "Maintenance"

- Auto Minor Version Upgrade -> No
- Maintenance Window -> No Preference

Click on "Launch DB Instance"

RDS : AWS Console

Services Resource Groups

student Singapore Support

Backup

Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to detail [here](#).

Backup Retention Period 7 days

Backup Window No Preference

Monitoring

Enable Enhanced Monitoring No

Maintenance

Auto Minor Version Upgrade No

Maintenance Window No Preference

* Required

Cancel Previous **Launch DB Instance**

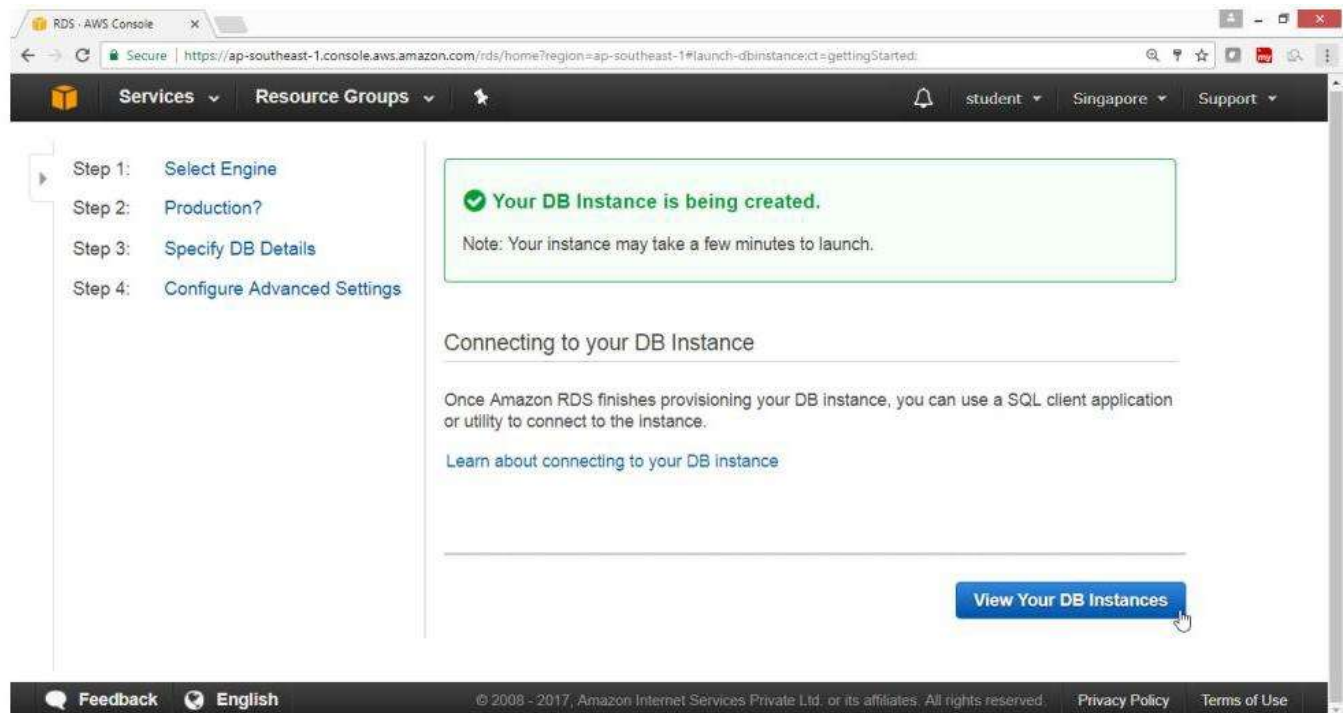
Select the period in which you want pending modifications, (such as changing the DB instance class) or patches applied to the DB instance by Amazon RDS. Any such maintenance should be started and completed within the selected period. If you do not select a period, Amazon RDS will assign a period randomly. [Learn More](#).

Feedback English

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

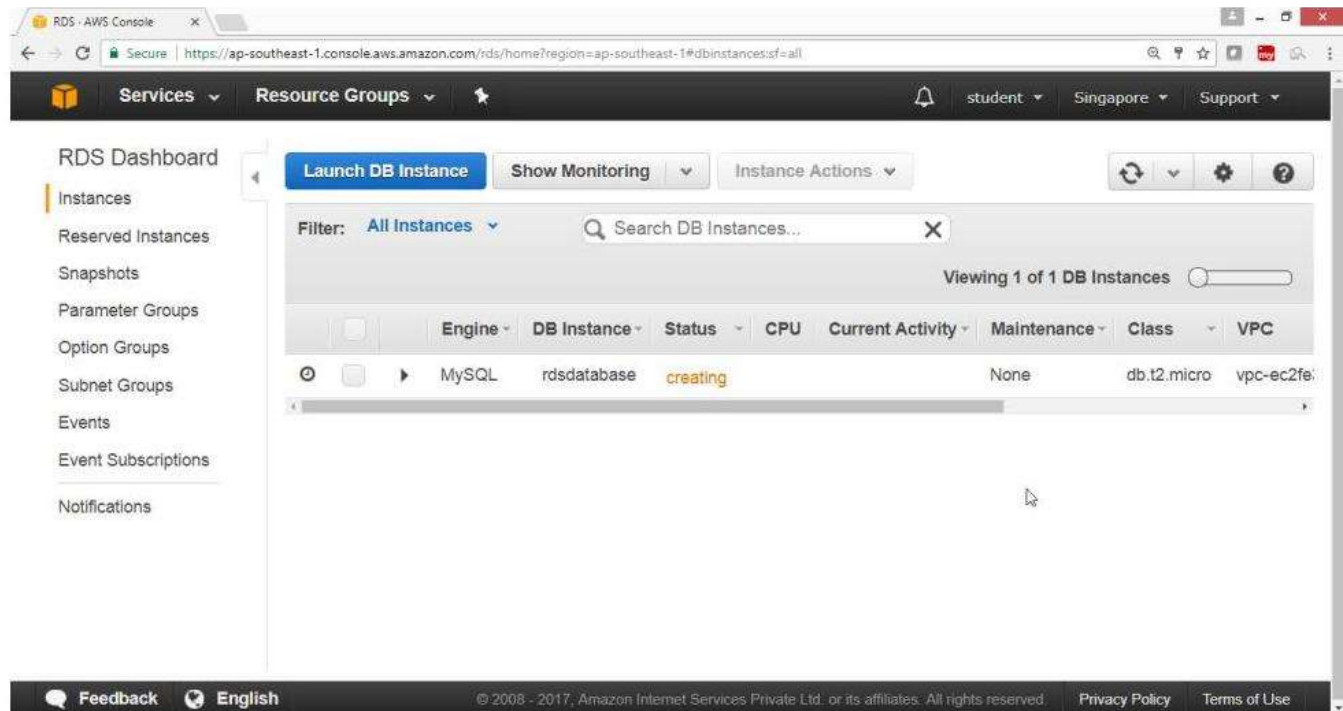
Your DB Instance is being created.

Click on "View Your DB Instances" Button



Under status column

Verify "creating"



Select "MySQL Engine"

The screenshot shows the AWS RDS console with a MySQL instance named 'rdsdatabase' in the 'creating' state. The instance is of class 'db.t2.micro' in VPC 'vpc-ec2'. The 'Endpoint' is 'Not available yet'. The 'Alarms and Recent Events' table shows three events: 'DB instance deleted' at 10:07 PM, 'DB instance shutdown' at 10:03 PM, and 'Finished DB Instance backup' at 9:05 PM. The 'Monitoring' section shows 'CPU' and 'Memory' with 'No Data'.

TIME (UTC+5:30)	EVENT
Aug 11 10:07 PM	DB instance deleted
Aug 11 10:03 PM	DB instance shutdown
Aug 11 9:05 PM	Finished DB Instance backup

	CURRENT VALUE	THRESHOLD	LAST HOUR
CPU	No Data		
Memory	No Data		

Under status column

Verify "backing-up"

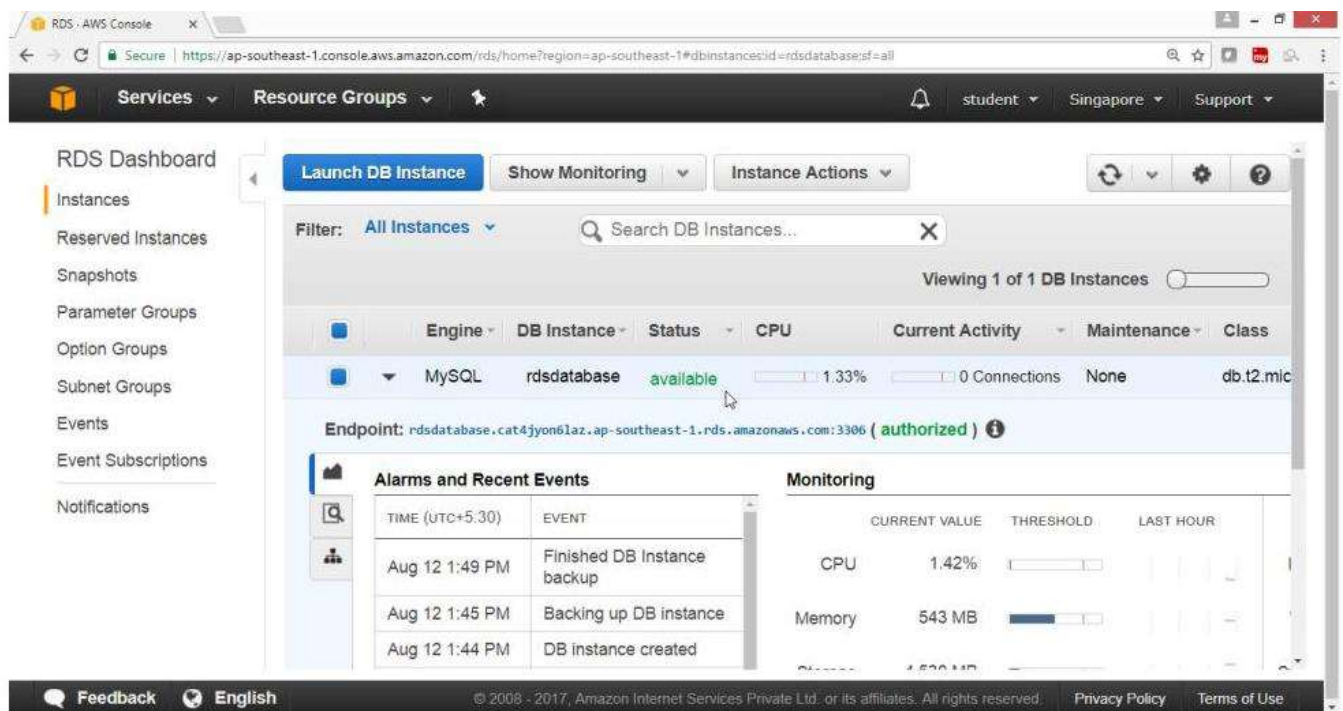
The screenshot shows the AWS RDS console with the same MySQL instance 'rdsdatabase' now in the 'backing-up' state. The 'Endpoint' is 'rdsdatabase.cat4jyon6laz.ap-southeast-1.rds.amazonaws.com:3306 (authorized)'. The 'Alarms and Recent Events' table shows the same three events as before. The 'Monitoring' section shows 'CPU' and 'Memory' with 'No Data'.

TIME (UTC+5:30)	EVENT
Aug 11 10:07 PM	DB instance deleted
Aug 11 10:03 PM	DB instance shutdown
Aug 11 9:05 PM	Finished DB Instance backup

	CURRENT VALUE	THRESHOLD	LAST HOUR
CPU	No Data		
Memory	No Data		

Under status column

Verify "Available"



Client Side

Go to Linux Box

Run **MYSQL** client command to connect to RDS database

```
$mysql -u <username> -h <End_point_of_RDS_Instance> -p <password>
```

```
shaikh@shaikh-virtual-machine:~$ mysql -u testuser -h rdsdatabase.clkyahad3ggx.ap-southeast-1.rds.amazonaws.com -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 31
Server version: 5.6.35-log MySQL Community Server (GPL)

Copyright (c) 2000, 2017, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

To see the list of databases;

show databases;

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| innodb |
| mysql |
| performance_schema |
| salesdba |
| sys |
+-----+
6 rows in set (0.02 sec)

mysql> █
```

Use the database

Create table

Insert values in tables

```
mysql>
mysql> use salesdba;
Database changed
mysql>
mysql> create table tutorials_tbl(tutorial_id INT NOT NULL AUTO_INCREMENT,tutorial_title VARCHAR(100) NOT NULL,tutorial_author VARCHAR(40) NOT NULL,submission_date DATE,PRIMARY KEY ( tutorial_id ));
Query OK, 0 rows affected (0.04 sec)

mysql>
mysql> INSERT INTO tutorials_tbl(tutorial_title, tutorial_author, submission_date) VALUES("Learn PHP", "John Poul", NOW());
Query OK, 1 row affected, 1 warning (0.02 sec)

mysql>
mysql> INSERT INTO tutorials_tbl(tutorial_title, tutorial_author, submission_date) VALUES("Learn MySQL", "Abdul S", NOW());
Query OK, 1 row affected, 1 warning (0.03 sec)

mysql>
mysql> INSERT INTO tutorials_tbl(tutorial_title, tutorial_author, submission_date) VALUES("JAVA Tutorial", "Sanjay", '2007-05-06');
Query OK, 1 row affected (0.02 sec)

mysql>
mysql>
```

To see the structure of table

desc <table_name>

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| innodb |
| mysql |
| performance_schema |
| salesdba |
| sys |
+-----+
6 rows in set (0.02 sec)

mysql> use salesdba;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> desc tutorials_tbl;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| tutorial_id | int(11) | NO | PRI | NULL | auto_increment |
| tutorial_title | varchar(100) | NO | | NULL | |
| tutorial_author | varchar(40) | NO | | NULL | |
| submission_date | date | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.02 sec)

mysql>
```

To see records in the tables

select * from <table_name>

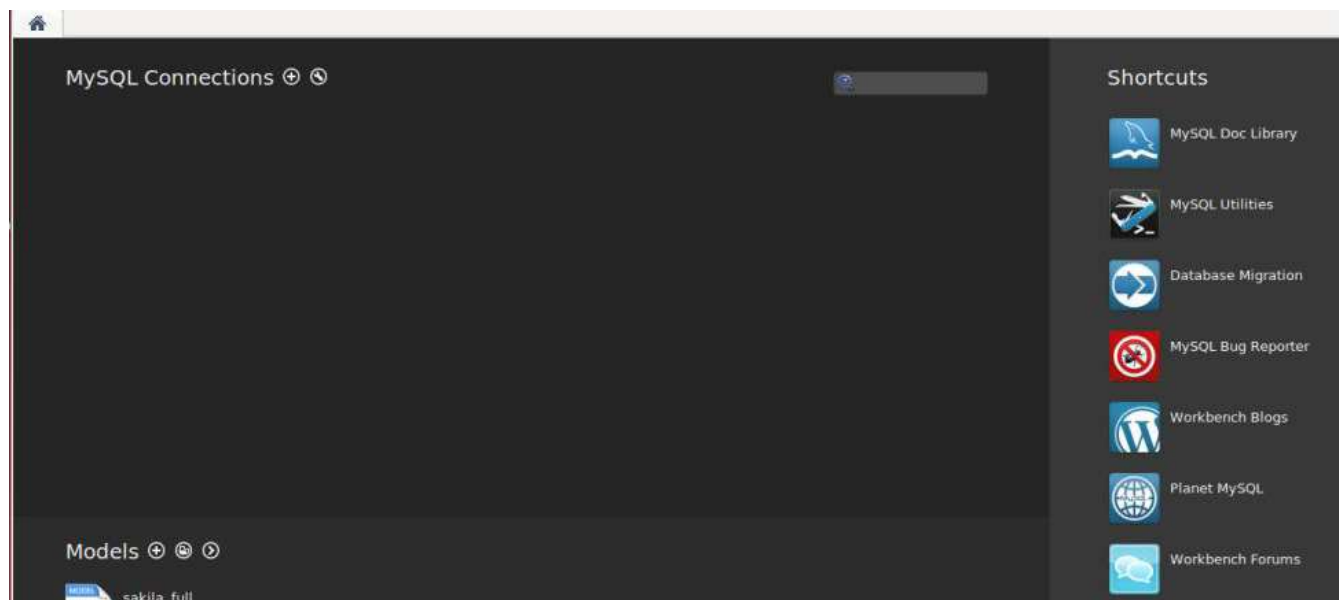
```
mysql> select * from tutorials_tbl;
+-----+-----+-----+-----+
| tutorial_id | tutorial_title | tutorial_author | submission_date |
+-----+-----+-----+-----+
| 1 | Learn PHP | John Poul | 2017-08-12 |
| 2 | Learn MySQL | Abdul S | 2017-08-12 |
| 3 | JAVA Tutorial | Sanjay | 2007-05-06 |
+-----+-----+-----+-----+
3 rows in set (0.02 sec)

mysql>
```

Step-2) To access RDS database through MYSQL WorkBench Client application

Open MYSQL WorkBench Client application, provide the following details

On MYSQL Connection Tag, click plus radio button



Provide the following values for

Connection Name: ->testcon1

Connection Method: ->Standard (TCP/IP)

Parameters

- Hostname->copy RDS url
- (rdsdatabase.clkyahad3ggx.ap-south-1.rds.amazonaws.com)
- Port->3306
- Username->testuser
- Password->*****

Setup New Connection

Connection Name: Type a name for the connection

Connection Method: Method to use to connect to the RDBMS

Parameters

Hostname: Port: Name or IP address of the server host. - TCP/IP port.

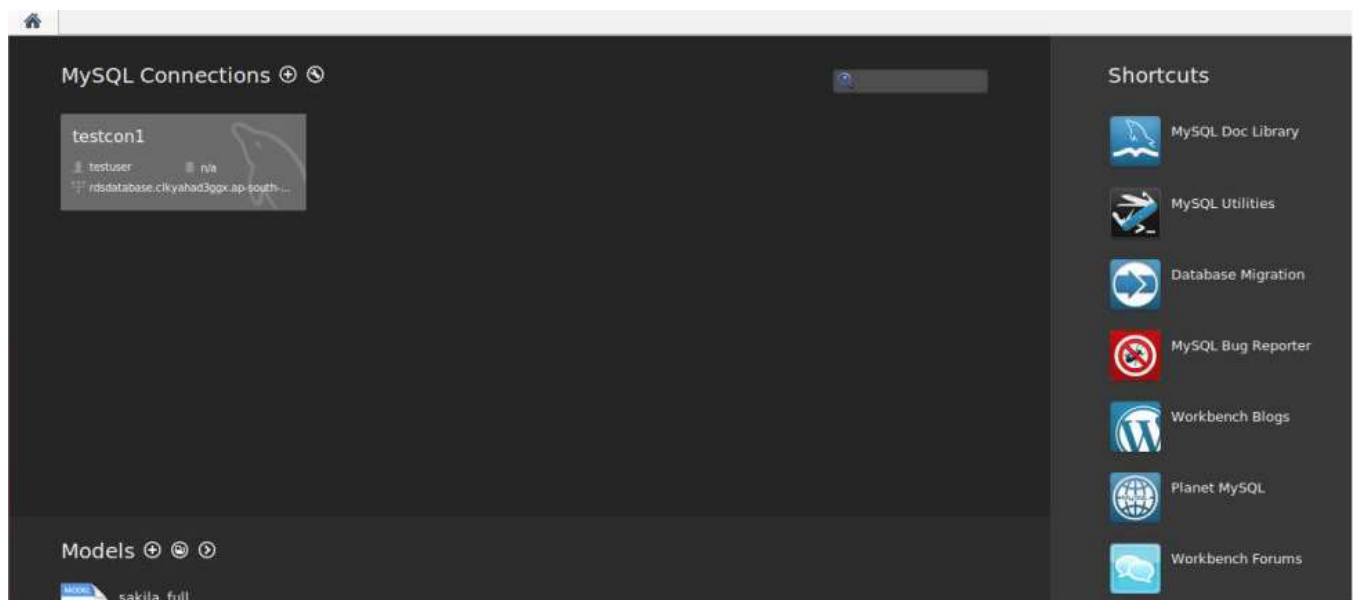
Username: Name of the user to connect with.

Password: The user's password. Will be requested later if it's not set.

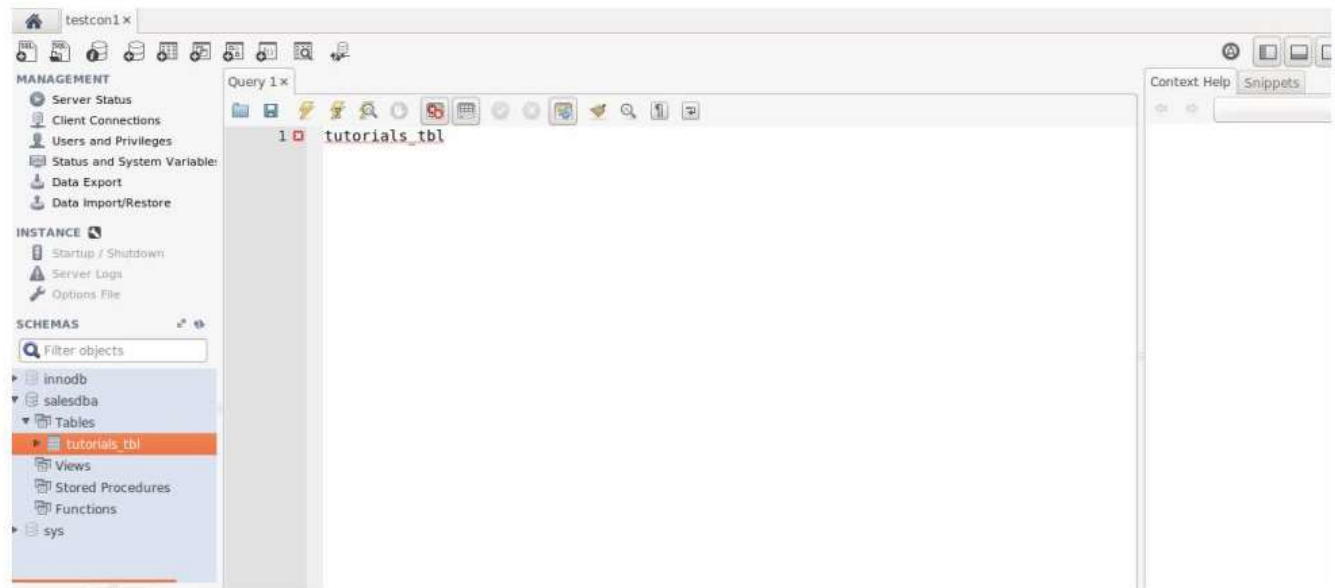
Default Schema: The schema to use as default schema. Leave blank to select it later

Verify

Connection is getting established



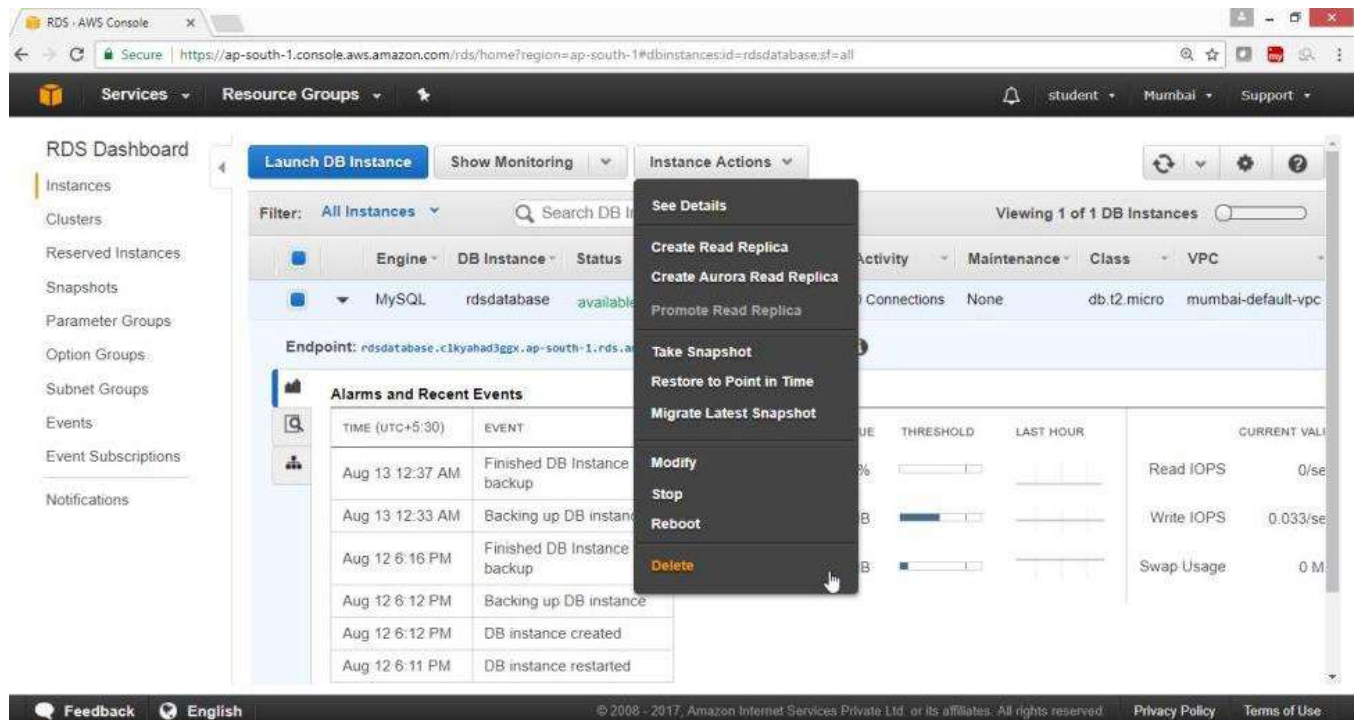
So, we can see that tables are listed in MYSQL clients



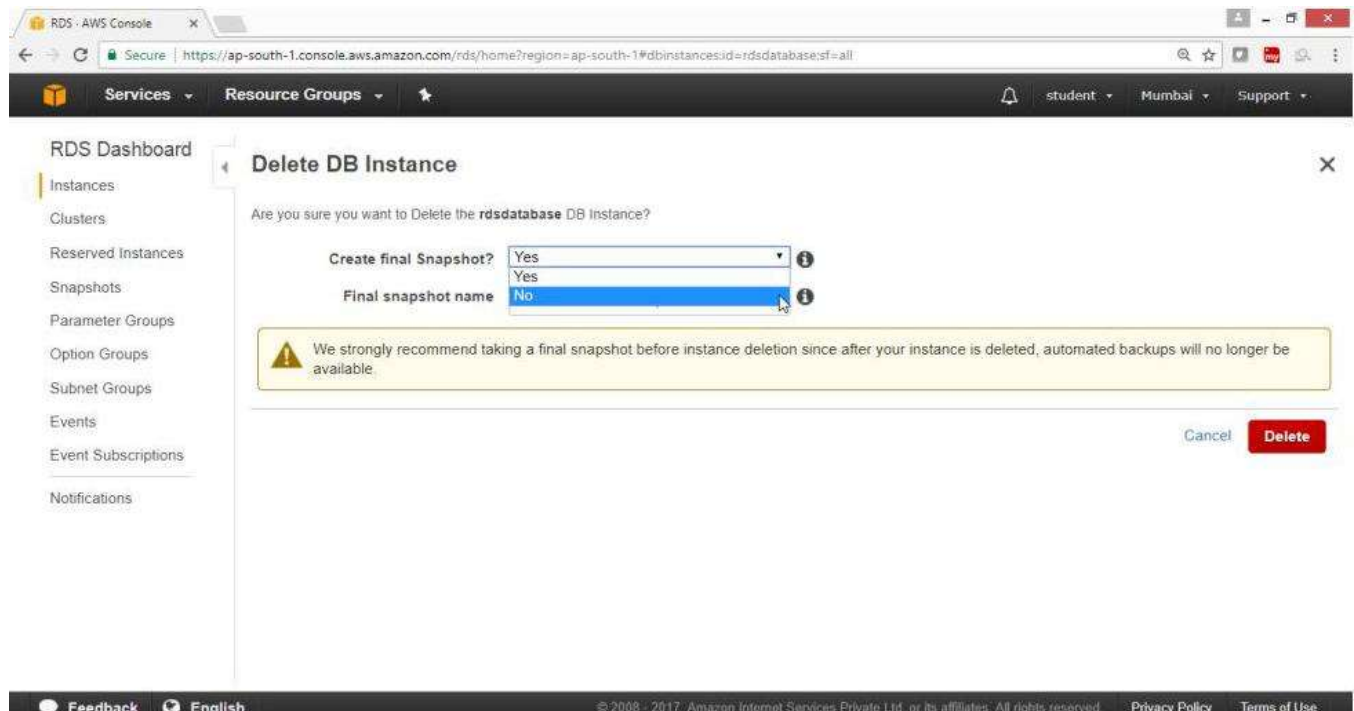
Step-3) To Delete the RDS instance

3.1 Open RDS Dashboard, select an instance

From Drop Down "Instance Action" Button, Select **Delete**

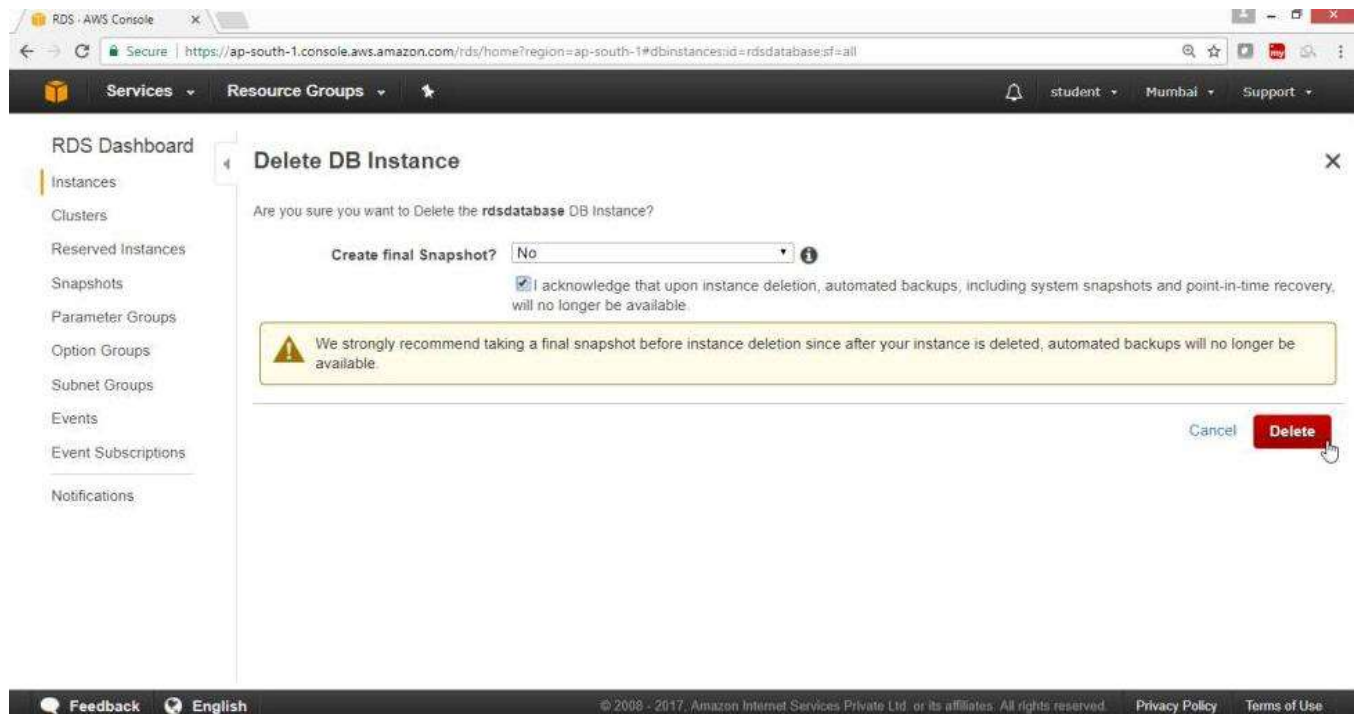


From Create final snapshot ->No



Select Acknowledge Check Box

Click on "Delete" Button



Verify

In status column->deleting

The screenshot shows the AWS RDS console with a single DB instance in the 'deleting' status. The instance is a MySQL database named 'rdsdatabase' in the 'ap-south-1' region. The status column shows 'deleting' with a red progress bar. The CPU usage is 1.00% and there are 0 connections. The instance class is 'db.t2.micro' and it is in the 'mumbai-default-vpc'.

TIME (UTC+5:30)	EVENT
Aug 13 12:37 AM	Finished DB Instance backup
Aug 13 12:33 AM	Backing up DB instance
Aug 12 6:16 PM	Finished DB Instance backup
Aug 12 6:12 PM	Backing up DB instance
Aug 12 6:12 PM	DB instance created
Aug 12 6:11 PM	DB instance restarted

Metric	Current Value	Threshold	Last Hour
CPU	0.915%		
Memory	536 MB		
Storage	4,530 MB		

Metric	Current Value
Read IOPS	0/sec
Write IOPS	0.208/sec
Swap Usage	0 MB

Delete Confirmed

The screenshot shows the AWS RDS console after the deletion of the DB instance. The status column now shows 'No DB Instances'. The console displays a message about the Relational Database Service (RDS) and a note that DB instances will launch in the Asia Pacific (Mumbai) region.

elational Database Service (RDS) is a web service that makes it easy to set up, operate, and scale a relational database in the cloud. We currently offer MySQL, PostgreSQL, and Oracle engines, allowing you to use the code, application and tools you already use with your existing database with Amazon RDS. You can find pricing information for RDS [here](#). Click the Launch DB Instance button to get started.

Note: Your DB Instances will launch in the Asia Pacific (Mumbai) region.

In RDS, what is the maximum value you can set for my backup retention period?

35 days

In RDS, Automated backups are enabled by default for new DB instance, true or false?

True.

What is Elastic Cache?

Elastic Cache is a web service that makes it easy to deploy, operate, and scale an in-memory cache in the cloud. The service improves the performance of web applications by allowing you to retrieve information from fast, managed, in-memory caches, instead of relying entirely on slower disk-based databases. Elastic Cache supports two open source in-memory engines namely: - [Memcached & Redis](#)

If you want to run a database on an EC2 instance, which is the most recommended Amazon storage option, S3, RDS or EBS?

EBS

In S3, what does RRS stand for?

Reduced Redundancy Storage

If I launch a standby RDS instance, will it be in the same Availability Zone as my primary?

- A. Only for Oracle RDS types
- B. Yes
- C. Only if it is configured at launch
- D. No

Answer D

Explanation: No, since the purpose of having a standby instance is to avoid an infrastructure failure (if it happens), therefore the standby instance is stored in a different availability zone, which is a physically different independent infrastructure.

When would I prefer Provisioned IOPS over Standard RDS storage?

- A. If you have batch-oriented workloads
- B. If you use production online transaction processing (OLTP) workloads.
- C. If you have workloads that are not sensitive to consistent performance
- D. All of the above

Answer A

Explanation: Provisioned IOPS deliver high IO rates but on the other hand it is expensive as well. Batch processing workloads do not require manual intervention they enable full utilization of systems, therefore a provisioned IOPS will be preferred for batch-oriented workload.

How is Amazon RDS, DynamoDB and Redshift different?

Amazon RDS is a database management service for relational databases, it manages patching, upgrading, backing up of data etc. of databases for you without your intervention. RDS is a Db management service for structured data only.

DynamoDB, on the other hand, is a NoSQL database service, NoSQL deals with unstructured data. Redshift, is an entirely different service, it is a data warehouse product and is used in data analysis.

If I am running my DB Instance as a Multi-AZ deployment, can I use the standby DB Instance for read or write operations along with primary DB instance?

- A. Yes
- B. Only with MySQL based RDS
- C. Only for Oracle RDS instances
- D. No**

Answer D

Explanation: No, Standby DB instance cannot be used with primary DB instance in parallel, as the former is solely used for standby purposes, it cannot be used unless the primary instance goes down.

Your company's branch offices are all over the world, they use a software with a multi-regional deployment on AWS, they use MySQL 5.6 for data persistence.

The task is to run an hourly batch process and read data from every region to compute cross-regional reports which will be distributed to all the branches. This should be done in the shortest time possible. How will you build the DB architecture in order to meet the requirements?

- A. For each regional deployment, use RDS MySQL with a master in the region and a read replica in the HQ region**
- B. For each regional deployment, use MySQL on EC2 with a master in the region and send hourly EBS snapshots to the HQ region
- C. For each regional deployment, use RDS MySQL with a master in the region and send hourly RDS snapshots to the HQ region
- D. For each regional deployment, use MySQL on EC2 with a master in the region and use S3 to copy data files hourly to the HQ region

Answer A

Explanation: For this we will take an RDS instance as a master, because it will manage our database for us and since we have to read from every region, we'll put a read replica of this instance in every region where the data has to be read from. Option C is not correct since putting a read replica would be more efficient than putting a snapshot, a read replica can be promoted if needed to an independent DB instance, but with a Db snapshot it becomes mandatory to launch a separate DB Instance.

Can I run more than one DB instance for Amazon RDS for free?

Yes. You can run more than one Single-AZ Micro database instance, that too for free! However, any use exceeding 750 instance hours, across all Amazon RDS Single-AZ Micro DB instances, across all eligible database engines and regions, will be billed at standard Amazon RDS prices.

For example: if you run two Single-AZ Micro DB instances for 400 hours each in a single month, you will accumulate 800 instance hours of usage, of which 750 hours will be free. You will be billed for the remaining 50 hours at the standard Amazon RDS price.

Which AWS services will you use to collect and process e-commerce data for near real-time analysis?

- A. Amazon ElastiCache
- B. Amazon DynamoDB
- C. Amazon Redshift
- D. Amazon Elastic MapReduce

Answer B, C

Explanation: DynamoDB is a fully managed NoSQL database service. DynamoDB, therefore can be fed any type of unstructured data, which can be data from e-commerce websites as well, and later, an analysis can be done on them using Amazon Redshift. We are not using Elastic MapReduce, since a near real time analyses is needed.

Can I retrieve only a specific element of the data, if I have a nested JSON data in DynamoDB?

Yes. When using the GetItem, BatchGetItem, Query or Scan APIs, you can define a Projection Expression to determine which attributes should be retrieved from the table. Those attributes can include scalars, sets, or elements of a JSON document.

A company is deploying a new two-tier web application in AWS. The company has limited staff and requires high availability, and the application requires complex queries and table joins.

Which configuration provides the solution for the company's requirements?

- A. MySQL Installed on two Amazon EC2 Instances in a single Availability Zone
- B. Amazon RDS for MySQL with Multi-AZ
- C. Amazon ElastiCache
- D. Amazon DynamoDB

Answer D

Explanation: DynamoDB has the ability to scale more than RDS or any other relational database service, therefore DynamoDB would be the apt choice.

What happens to my backups and DB Snapshots if I delete my DB Instance?

When you delete a DB instance, you have an option of creating a final DB snapshot, if you do that you can restore your database from that snapshot. RDS retains this user-created DB snapshot along with all other manually created DB snapshots after the instance is deleted, also automated backups are deleted and only manually created DB Snapshots are retained.

Which of the following use cases are suitable for Amazon DynamoDB? (Choose 2 answers)

- A. Managing web sessions.
- B. Storing JSON documents.
- C. Storing metadata for Amazon S3 objects.
- D. Running relational joins and complex updates.

Answer C, D

Explanation: If all your JSON data have the same fields eg [id,name,age] then it would be better to store it in a relational database, the metadata on the other hand is unstructured, also running relational joins or complex updates would work on DynamoDB as well.

How can I load my data to Amazon Redshift from different data sources like Amazon RDS, Amazon DynamoDB and Amazon EC2?

You can load the data in the following two ways: -

You can use the COPY command to load data in parallel directly to Amazon Redshift from Amazon EMR, Amazon DynamoDB, or any SSH-enabled host.

AWS Data Pipeline provides a high performance, reliable, fault tolerant solution to load data from a variety of AWS data sources. You can use AWS Data Pipeline to specify the data source, desired data transformations, and then execute a pre-written import script to load your data into Amazon Redshift.

Your application has to retrieve data from your user's mobile every 5 minutes and the data is stored in DynamoDB, later every day at a particular time the data is extracted into S3 on a per user basis and then your application is later used to visualize the data to the user. You are asked to optimize the architecture of the backend system to lower cost, what would you recommend?

- A. Create a new Amazon DynamoDB (able each day and drop the one for the previous day after its data is on Amazon S3.
- B. Introduce an Amazon SQS queue to buffer writes to the Amazon DynamoDB table and reduce provisioned write throughput.
- C. Introduce Amazon ElastiCache to cache reads from the Amazon DynamoDB table and reduce provisioned read throughput.
- D. Write data directly into an Amazon Redshift cluster replacing both Amazon DynamoDB and Amazon S3.

Answer C

Explanation: Since our work requires the data to be extracted and analyzed, to optimize this process a person would use provisioned IO, but since it is expensive, using a ElastiCache memoryinsread to cache the results in the memory can reduce the provisioned read throughput and hence reduce cost without affecting the performance.

You are running a website on EC2 instances deployed across multiple Availability Zones with a Multi-AZ RDS MySQL Extra Large DB Instance. The site performs a high number of small reads and writes per second and relies on an eventual consistency model. After comprehensive tests you discover that there is read contention on RDS MySQL. Which are the best approaches to meet these requirements? (Choose 2 answers)

- A. Deploy ElastiCache in-memory cache running in each availability zone
- B. Implement Sharding to distribute load to multiple RDS MySQL instances

C. Increase the RDS MySQL Instance size and Implement provisioned IOPS

D. Add an RDS MySQL read replica in each availability zone

Answer A, C

Explanation: Since it does a lot of read writes, provisioned IO may become expensive. But we need high performance as well, therefore the data can be cached using ElastiCache which can be used for frequently reading the data. As for RDS since read contention is happening, the instance size should be increased and provisioned IO should be introduced to increase the performance.

A startup is running a pilot deployment of around 100 sensors to measure street noise and air quality in urban areas for 3 months. It was noted that every month around 4GB of sensor data is generated. The company uses a load balanced auto scaled layer of EC2 instances and a RDS database with 500 GB standard storage. The pilot was a success and now they want to deploy at least 100K sensors which need to be supported by the backend. You need to store the data for at least 2 years to analyze it. Which setup of the following would you prefer?

A. Add an SQS queue to the ingestion layer to buffer writes to the RDS instance

B. Ingest data into a DynamoDB table and move old data to a Redshift cluster

C. Replace the RDS instance with a 6 node Redshift cluster with 96TB of storage

D. Keep the current architecture but upgrade RDS storage to 3TB and 10K provisioned IOPS

Answer C

Explanation: A Redshift cluster would be preferred because it easy to scale, also the work would be done in parallel through the nodes, therefore is perfect for a bigger workload like our use case. Since each month 4 GB of data is generated, therefore in 2 year, it should be around 96 GB. And since the servers will be increased to 100K in number, 96 GB will approximately become 96TB. Hence option C is the right answer.



DynamoDB

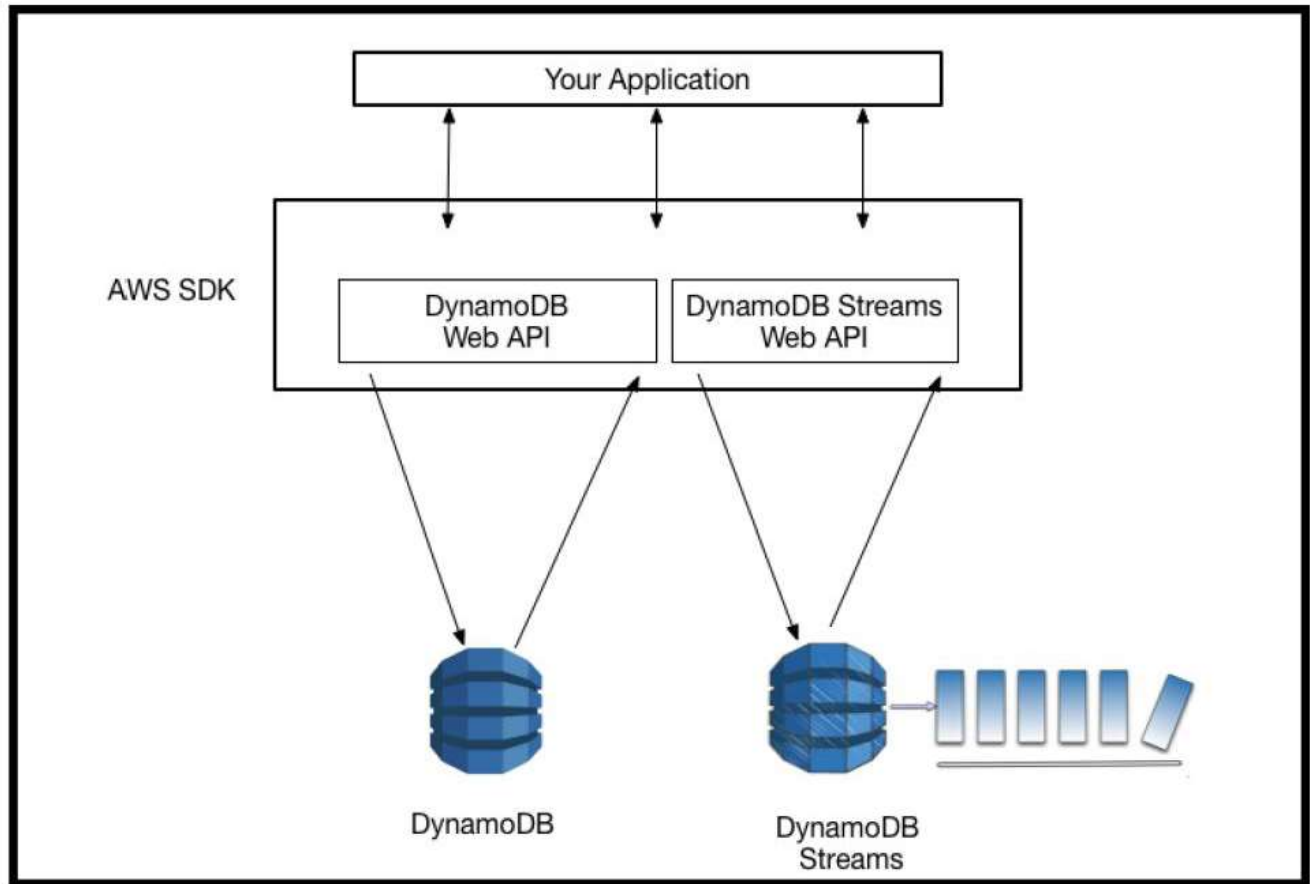
DynamoDB Highlights

- AWS provides a NoSQL database called Amazon DynamoDB.
- It can be used to store data in a **NoSQL environment**.
- DynamoDB gives very **fast and predictable performance**.
- It is highly **scalable**.
- We can use Amazon DynamoDB to create a database table to **store and retrieve any amount of data**.
- It is capable of serving very high volume of request **traffic**. Any level of request traffic. Amazon DynamoDB also provides support for automatically distributing the data and traffic of a table on multiple servers to handle the spikes in request traffic. Even after distributing the load it provides consistent performance.
- Stored on **SSD storage**
- Spread **Across 3 geographically distinct data centers**
- Eventually **Consistent Reads** (By Default)
- Strongly Consistent Reads

Share the CloudWatch Configuration Step by Step?

To configure a table, create records in Amazon DynamoDB

Topology



Pre-requisites

User should have AWS account, or IAM user with AmazonDynamoDBFullAccess

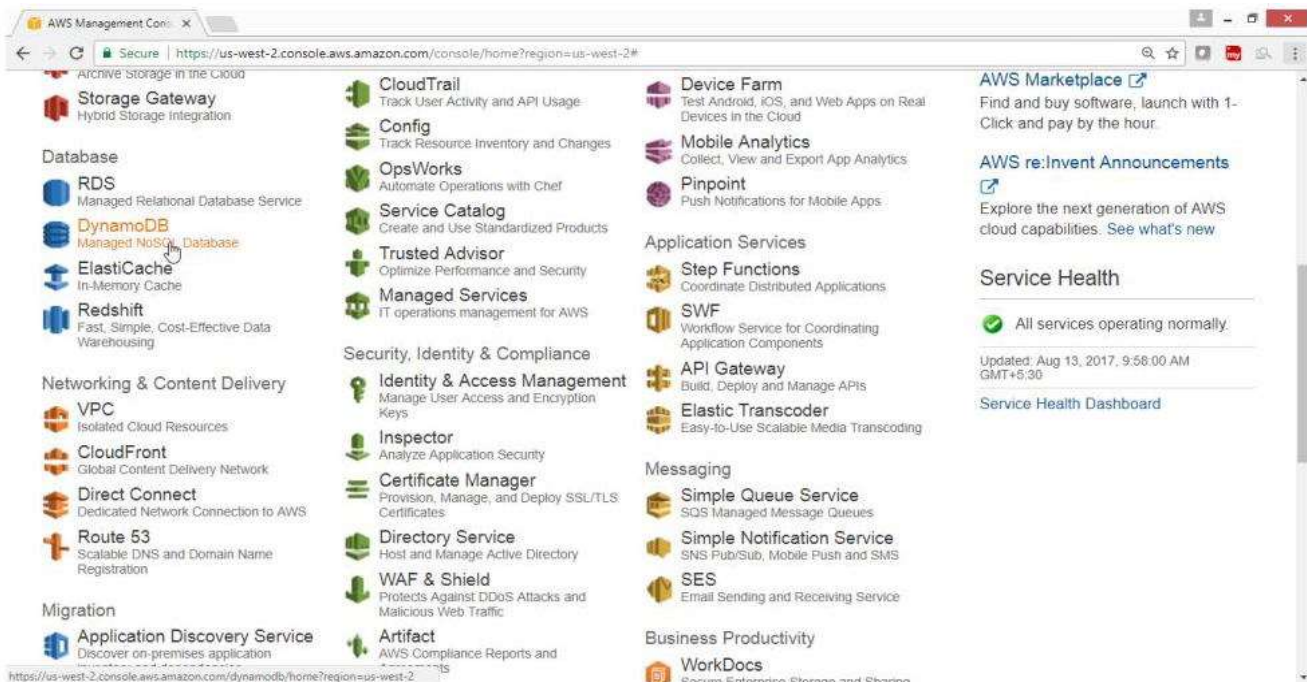
Task

- Create DynamoDB table
- Provide Provisioned Read/Write capacity
- Add the values to a table
- Scan the table
- Query table
- Delete the table

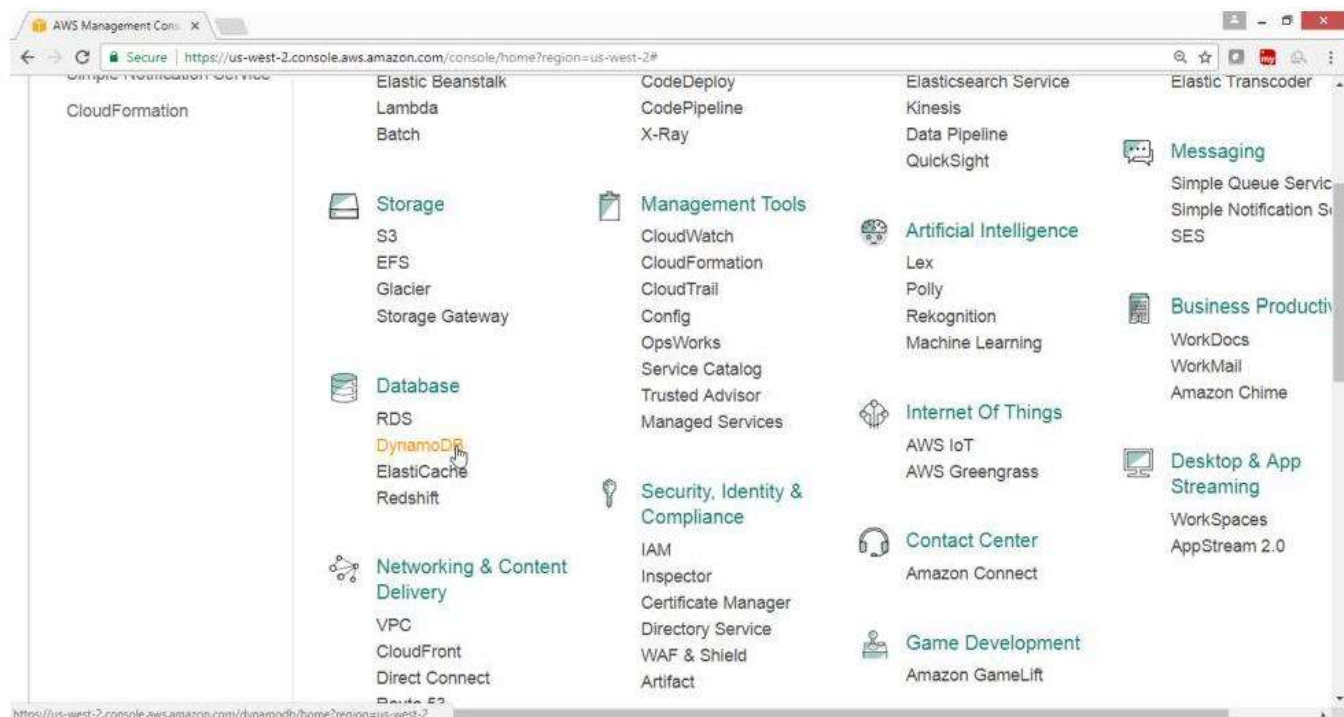
Step -1) To create an Amazon DynamoDB Table

Open AWS console

- Select services Database
- Click on “DynamoDB”

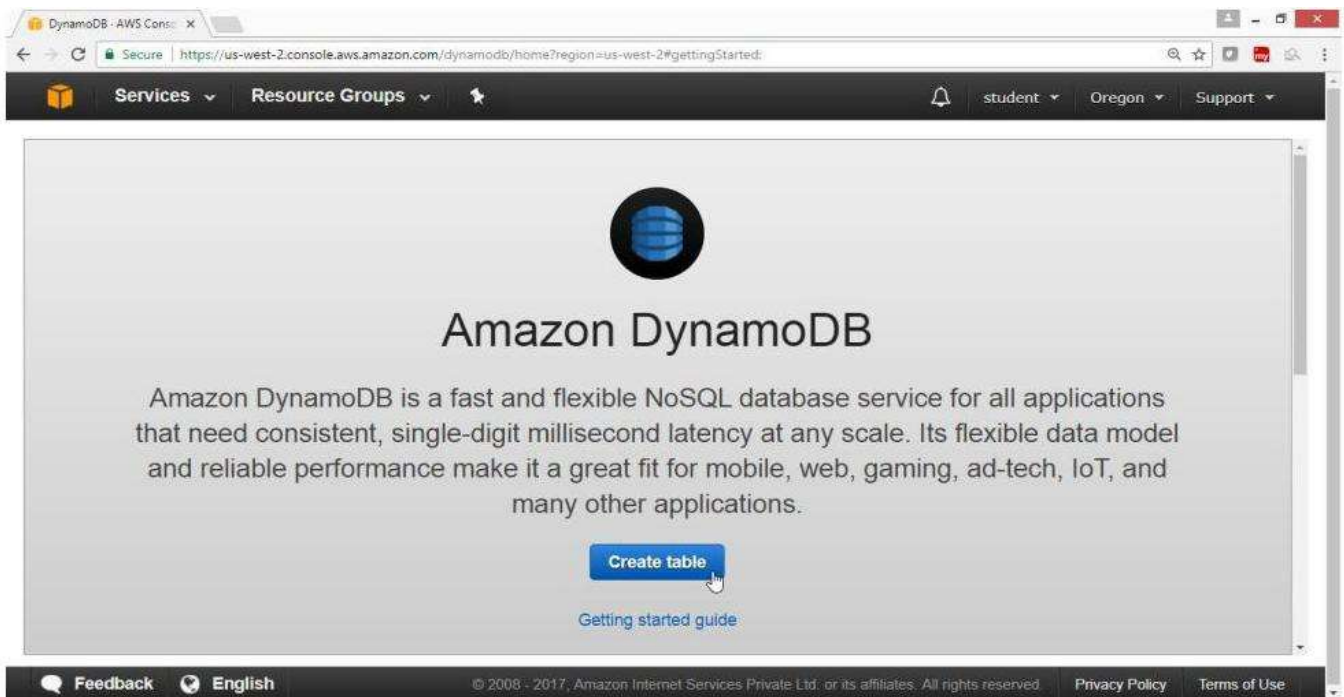


Or



From DynamoDB Dashboard

Click on "Create table" Button

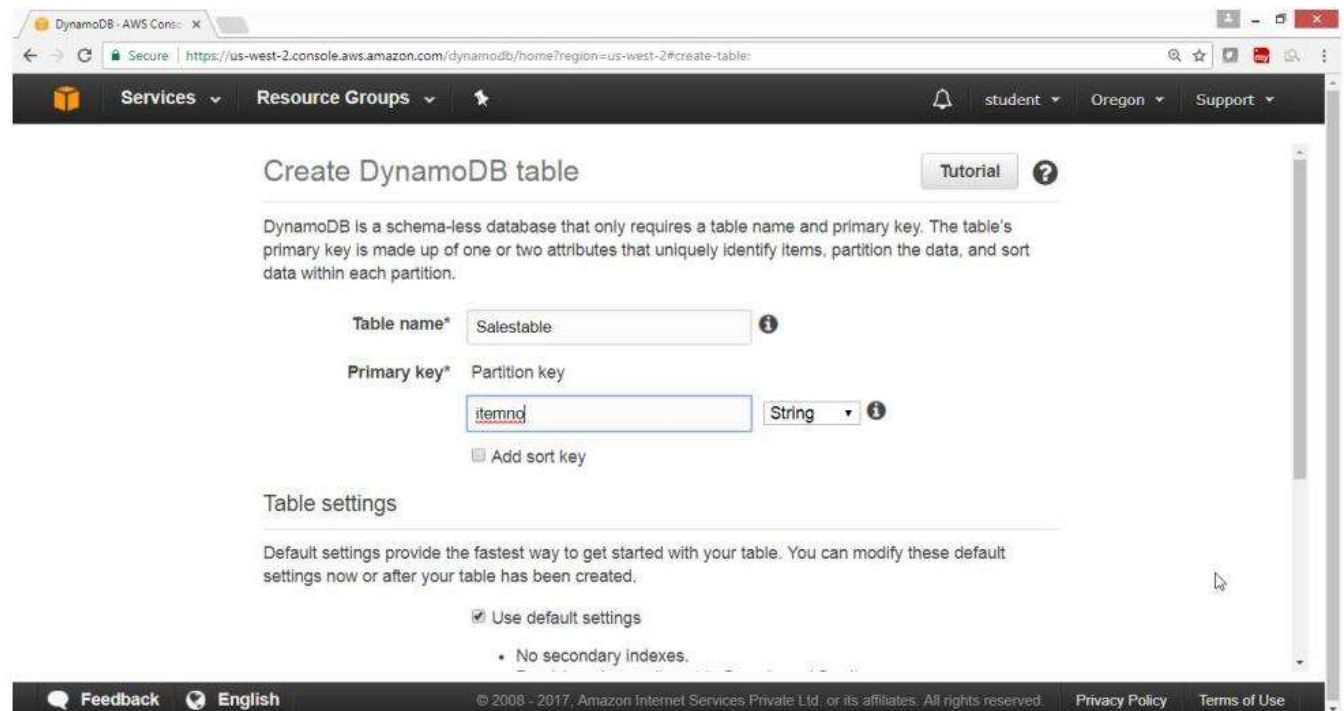


On "Create DynamoDB table" wizard

Provide following value

Table name* -> **Salestable**

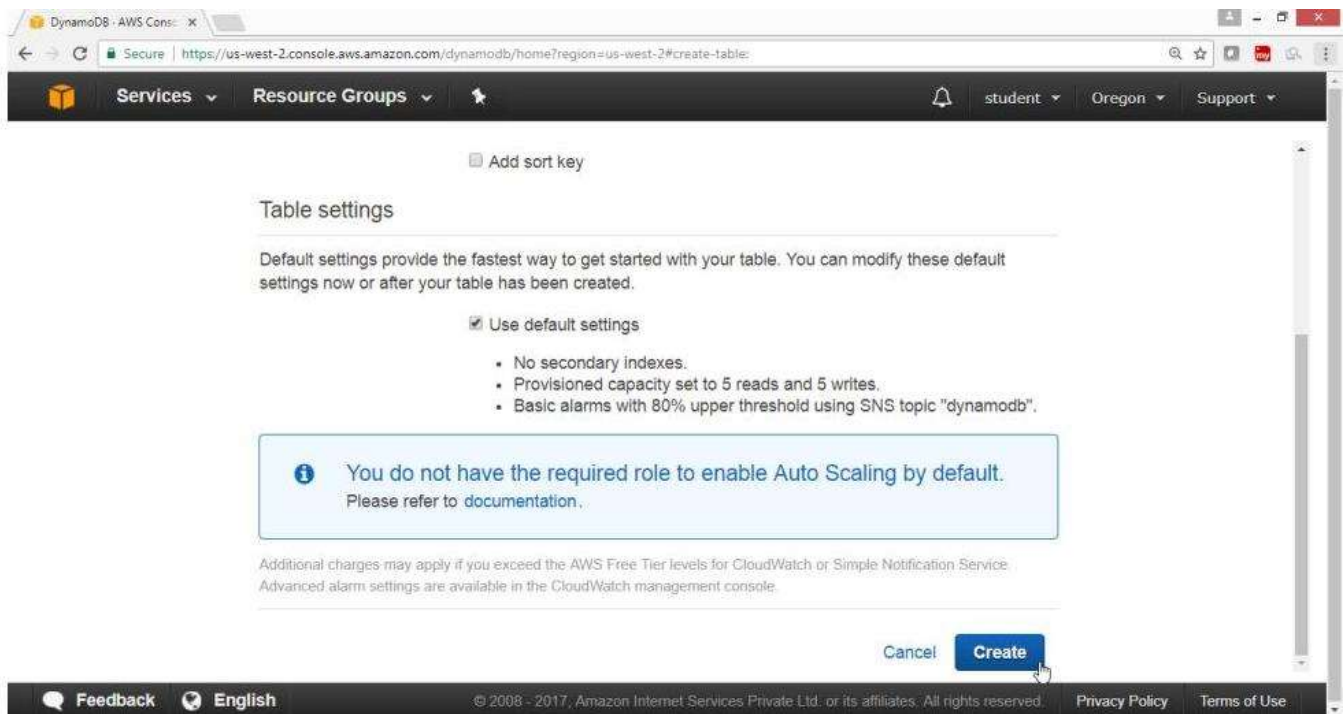
Partition Key -> **itemno, Select String**



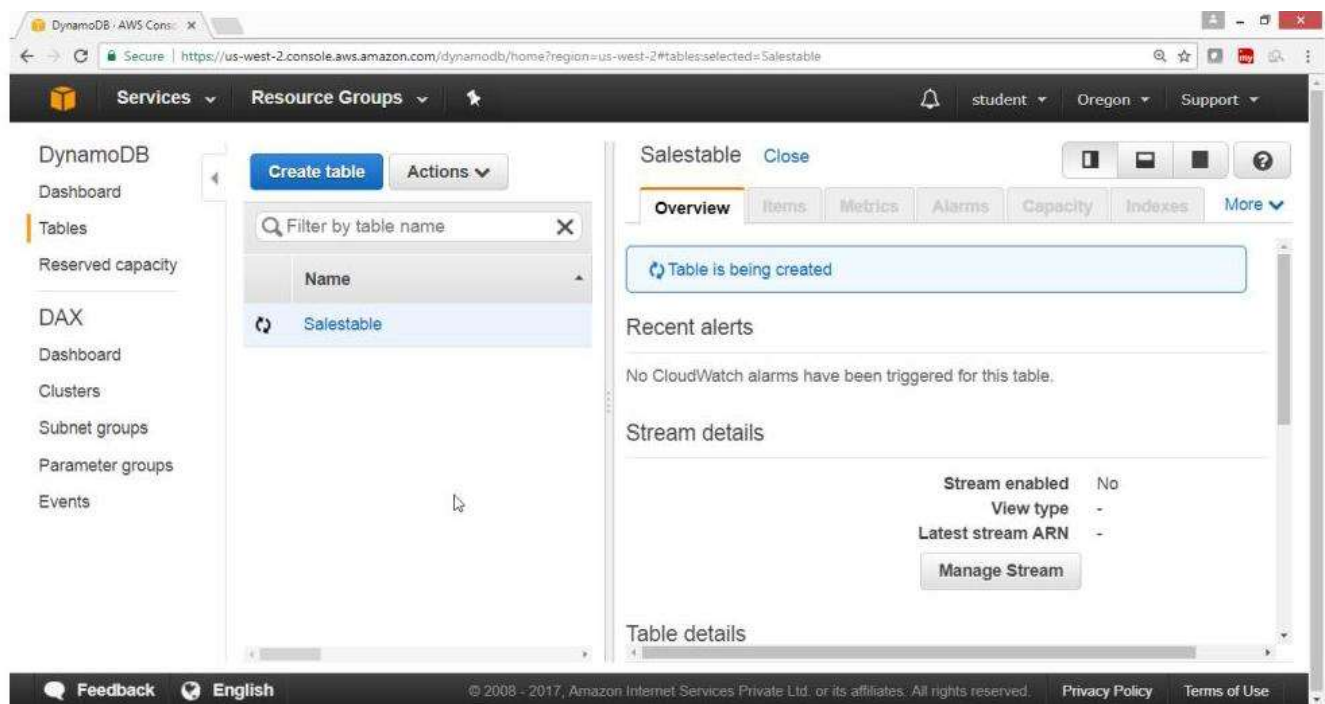
Under Table settings

Select "Use Default Settings" checkbox

Click on "Create" button

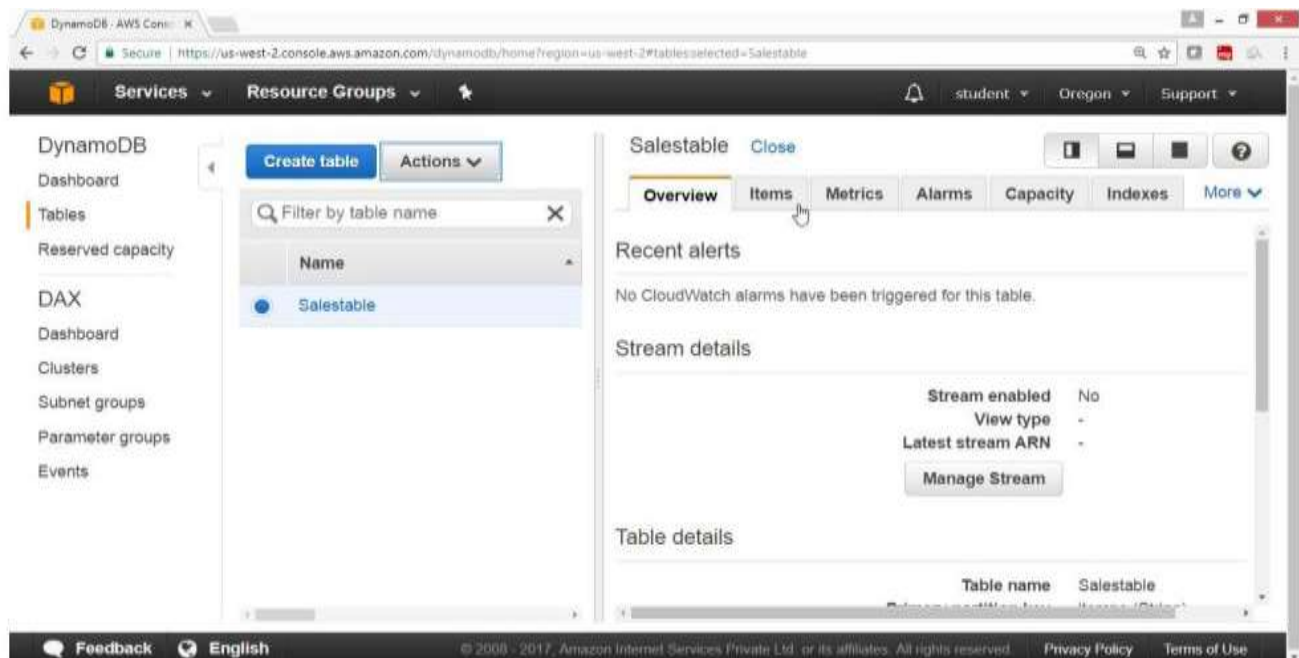


Creating



Verification

Salestable is created



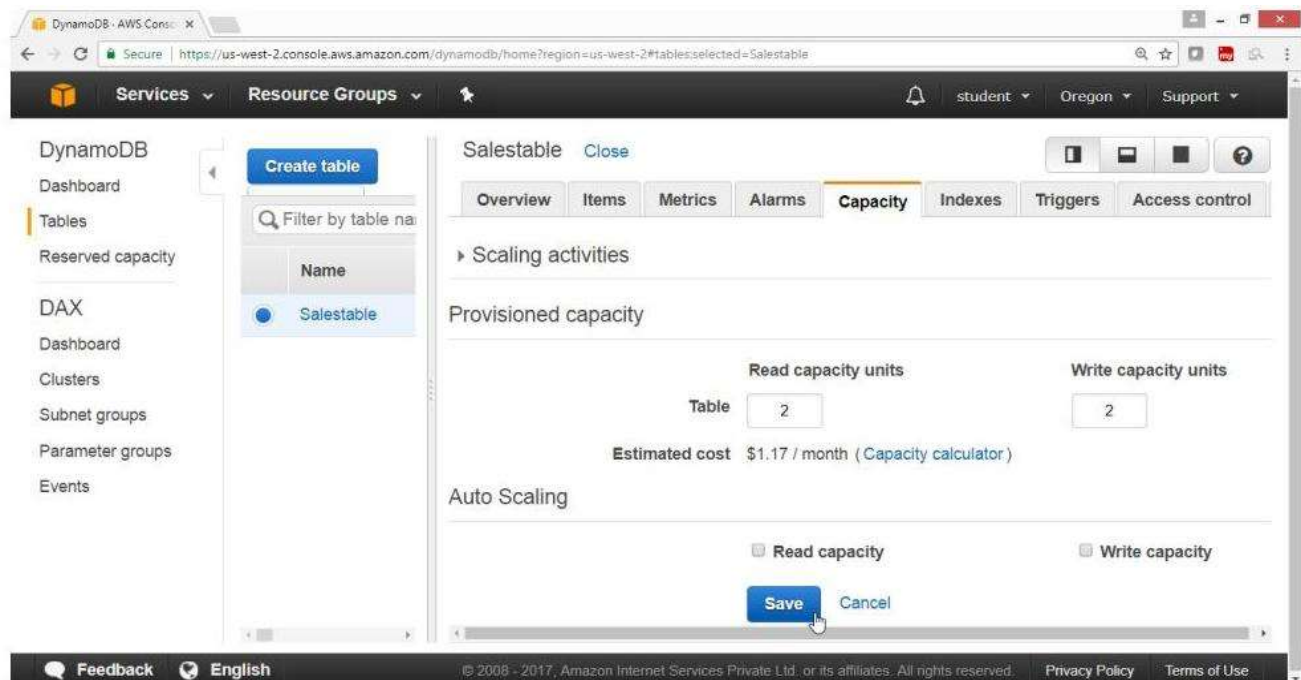
Select Capacity

Under "Provisioned Capacity"

Provide the following values

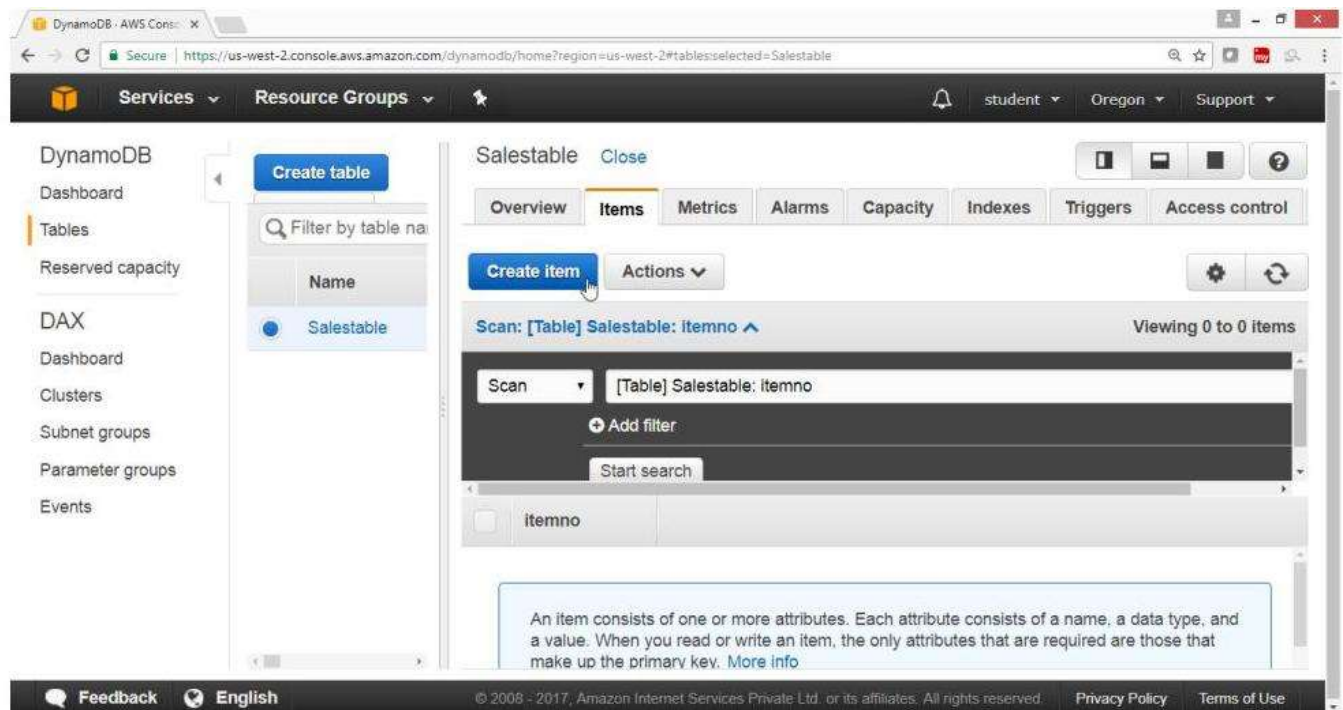
- Read Capacity ->2
- Write Capacity Units ->2

Click on "Save" button



Select item

Click on **Create item**



To add, append, insert values in the table

Open DynamoDB Dashboard, select Tables

Select the tables from the tables list

Check status, by clicking on

- Overview
- Items
- Metrics
- Alarms
- Capacity
- Indexes
- Triggers
- Access Control

Select Items, add tables field

Click on "Create Items"

On "Create Items" page

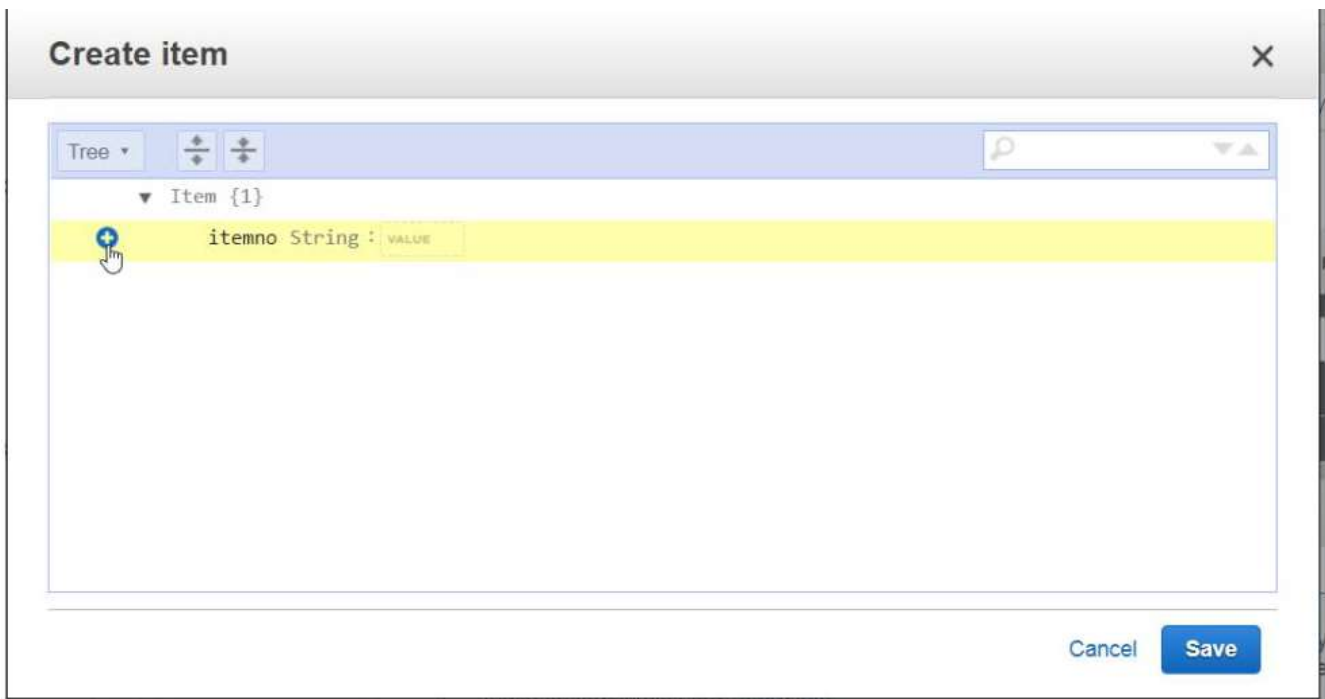
Click on Tree

Click on plus radio button

provide

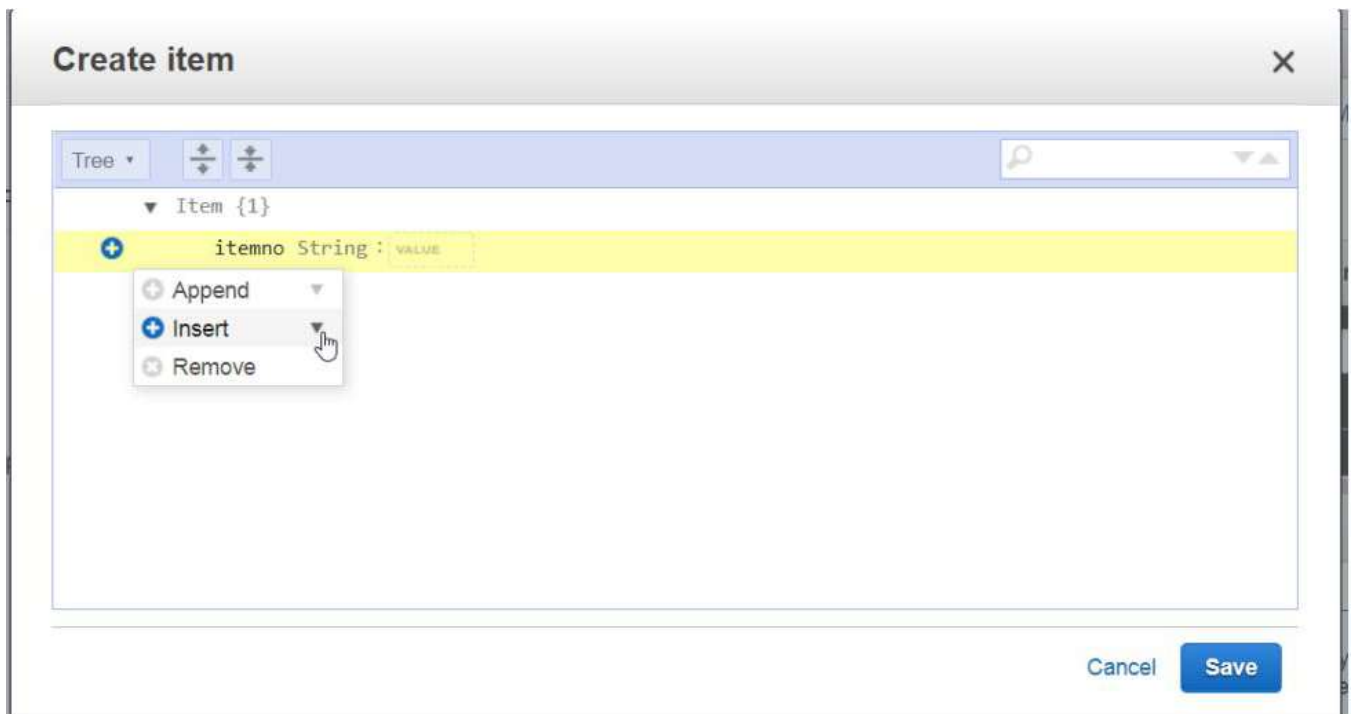
itemnostring 1

Click on plus radio button

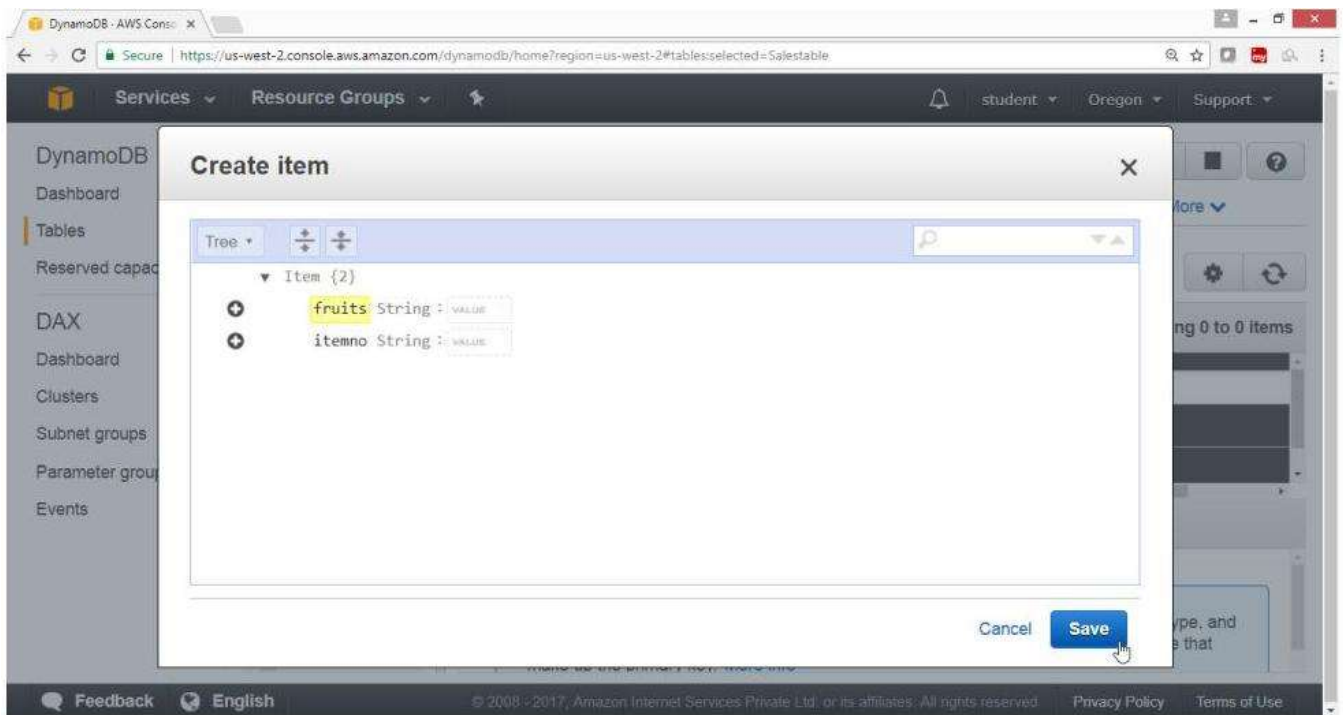


Select insert, select string

Item Name String Fruits



Verify Output

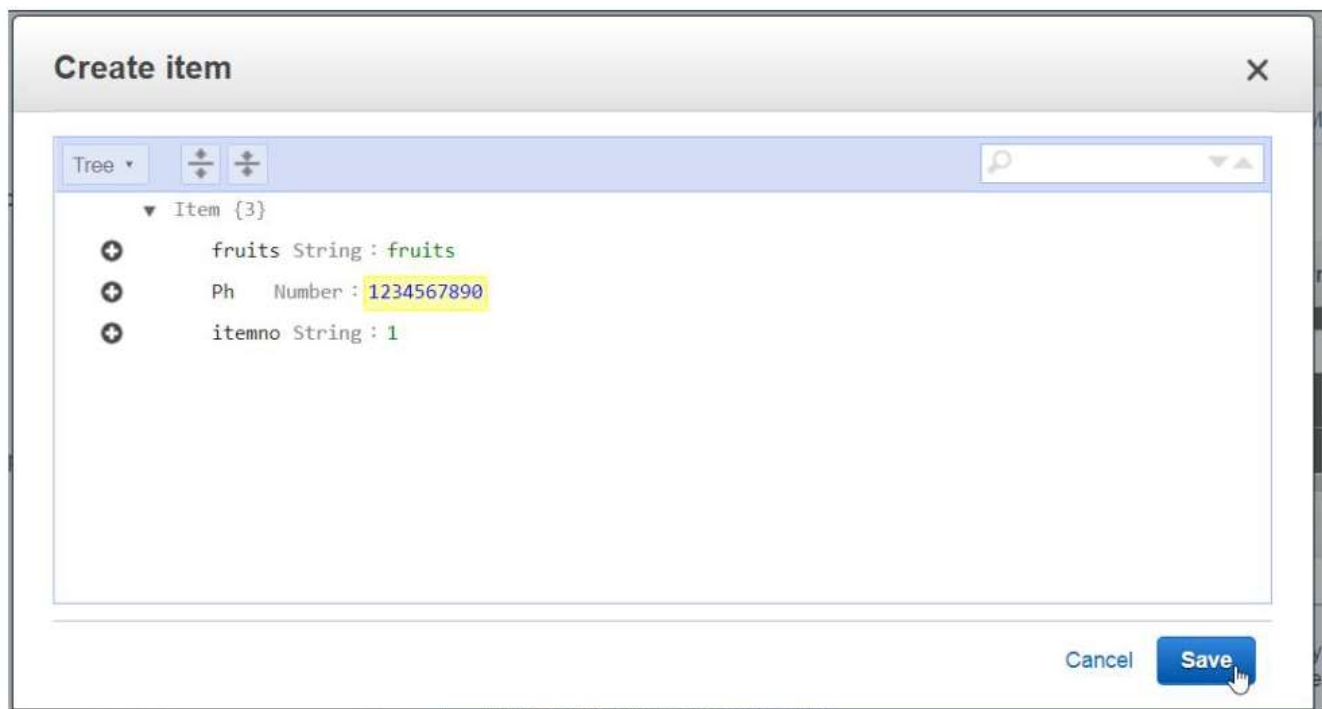


Click on plus radio button

select insert, select number

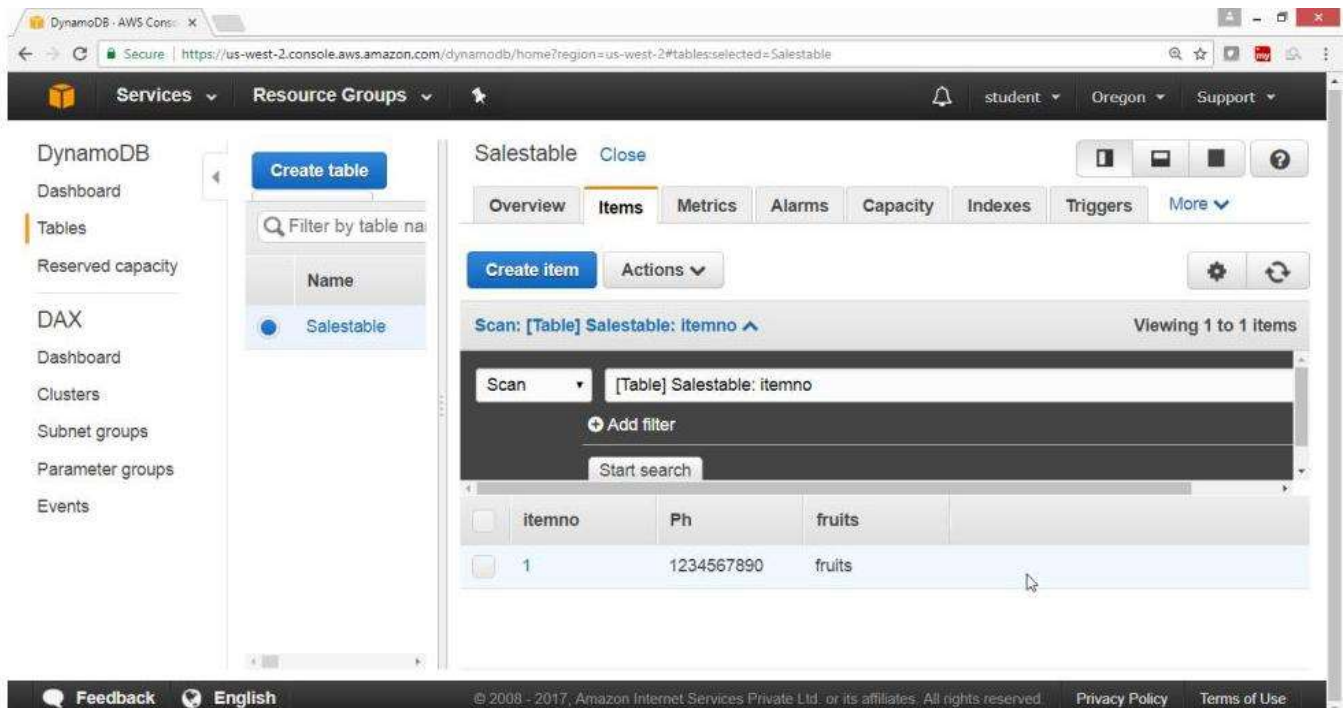
ph->123456789

Click on "Save"



To view all entered data

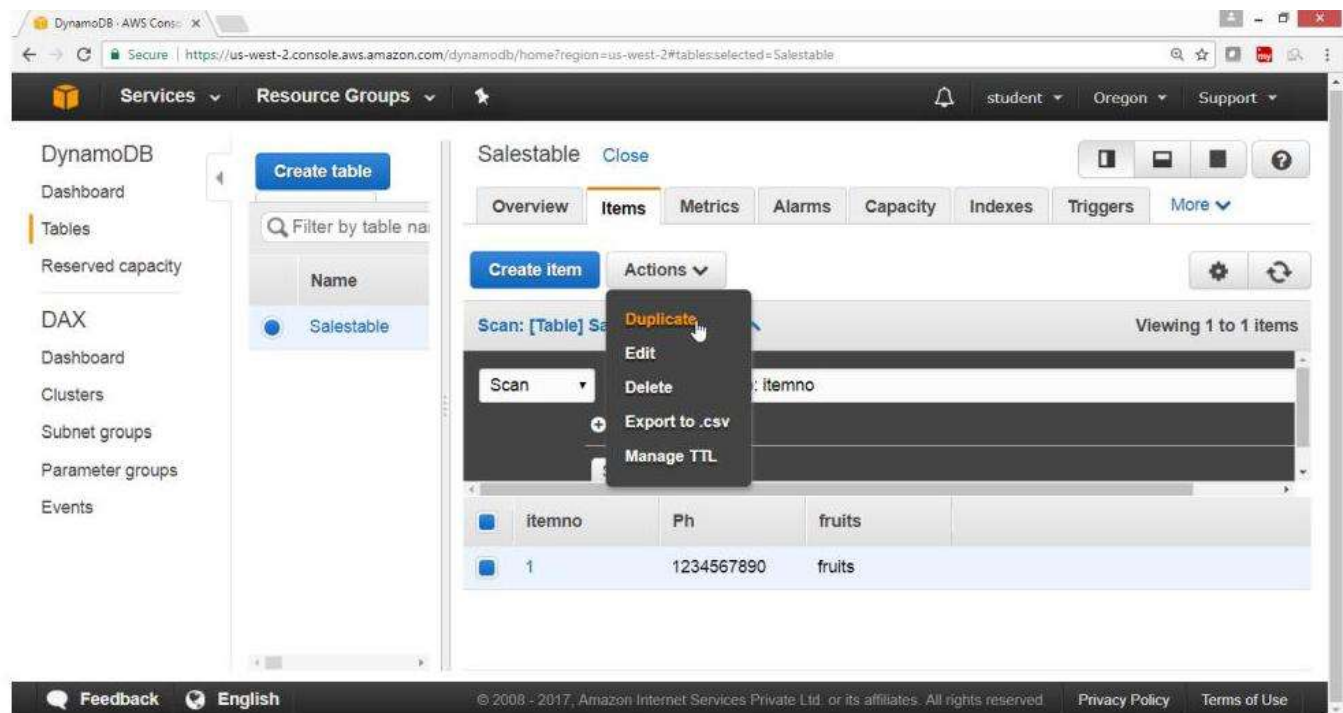
Select Scan, click "Start Search"



To add values in the created fields

Select the Table row, click "Actions" button

Select "Duplicate"



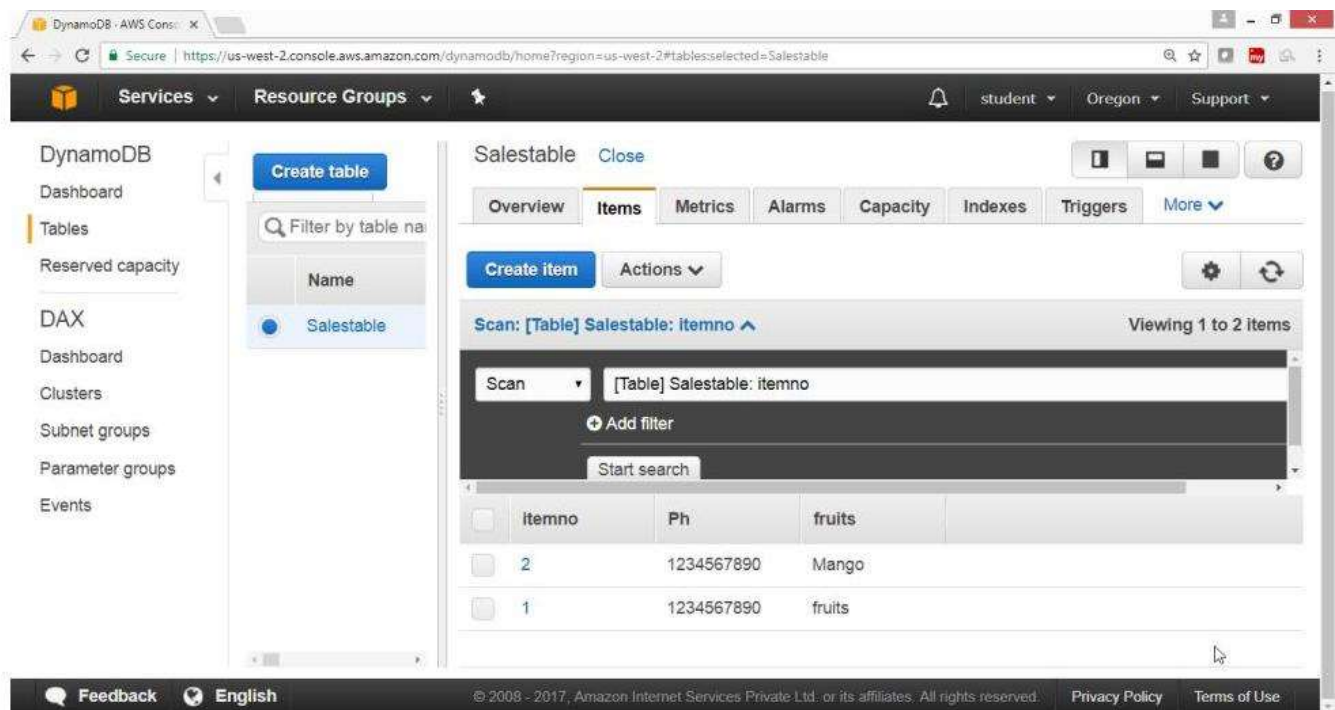
Now modify the values of the field

New row will be created

Click on "Save"



Verify



To Delete the table permanently for DynamoDB

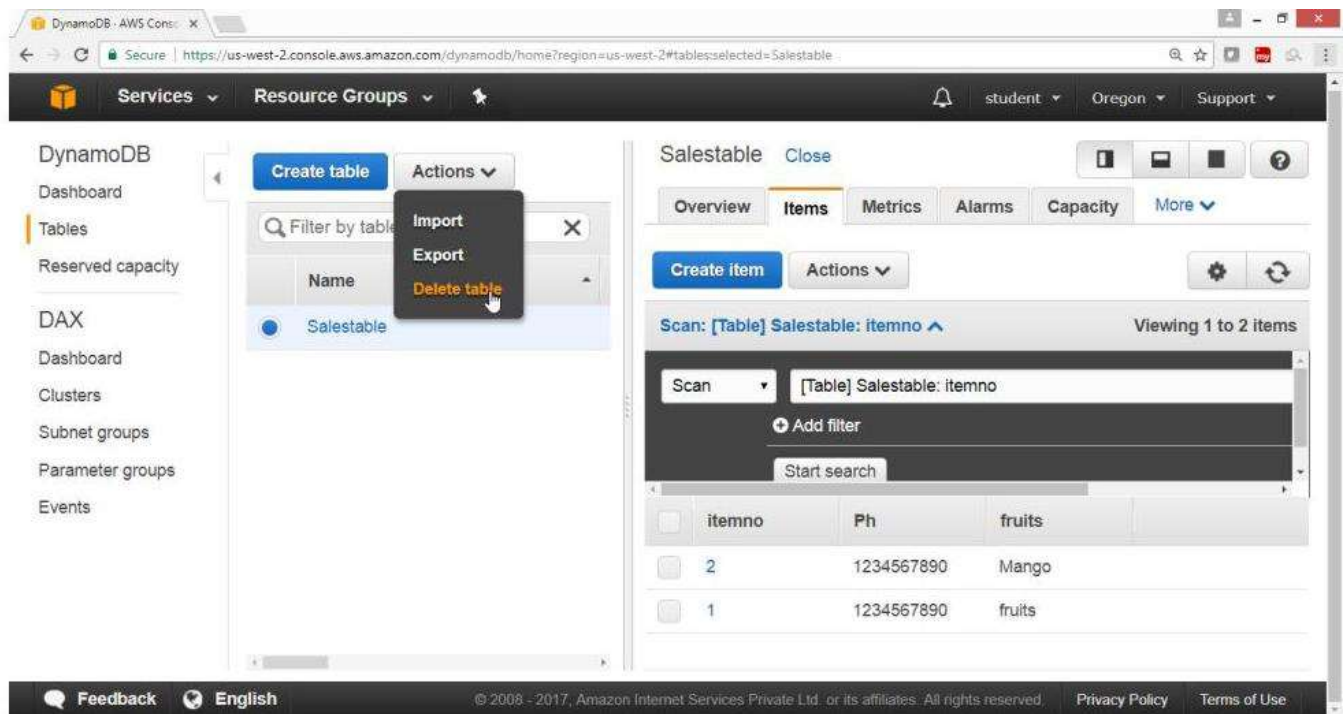
From the AWS Console

- Select Services Database
- Choose **DynamoDB**

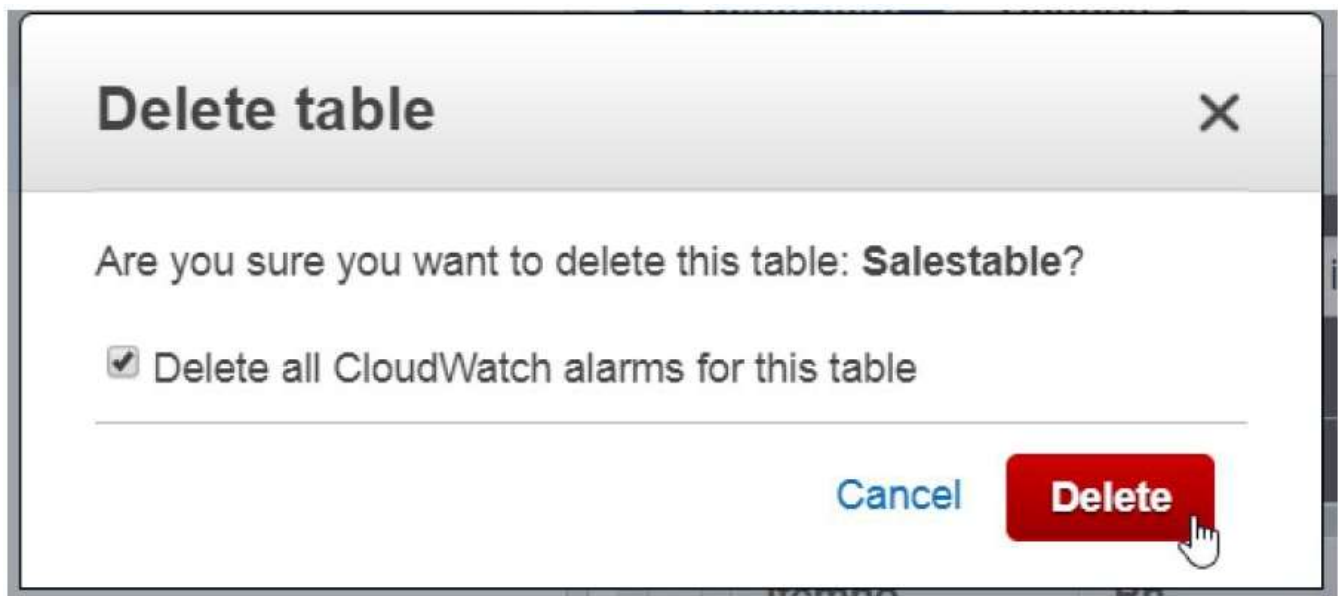
Under Tables, select the table for the list

Click on **Actions** button

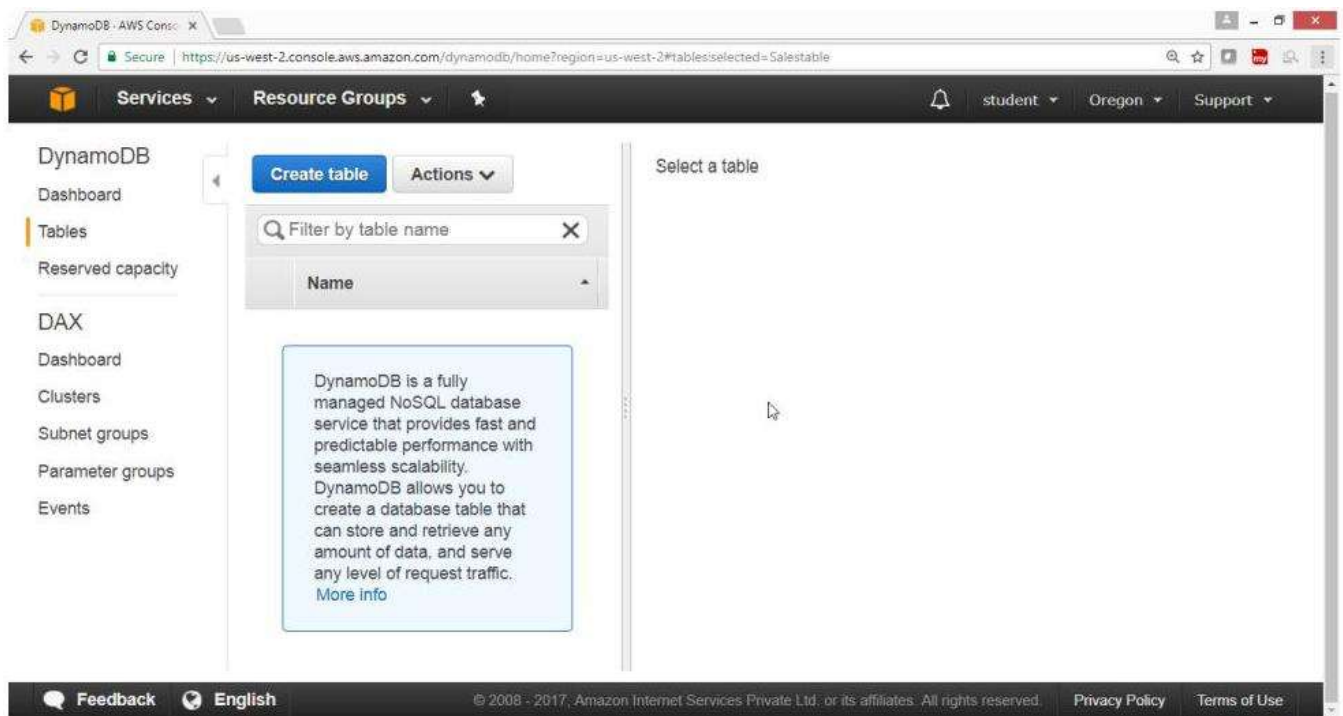
Select "**Delete Table**"



Click on "Delete" button



Verify Table is deleted



What is DynamoDB?

Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. You can use Amazon DynamoDB to create a database table that can store and retrieve any amount of data and serve any level of request traffic. Amazon DynamoDB automatically spreads the data and traffic for the table over a sufficient number of servers to handle the request capacity specified by the customer and the amount of data stored, while maintaining consistent and fast performance.

What are the main benefits of using Amazon DynamoDB?

Amazon DynamoDB is a highly scalable NoSQL database that has very fast performance. Some of the main benefits of using Amazon DynamoDB are as follows:-

Administration: In Amazon DynamoDB, we do not have to spend effort on administration of database. There are no servers to provision or manage. We just create our tables and start using them.

Scalability: DynamoDB provides the option to specify the capacity that we need for a table. Rest of the scalability is done under the hood by DynamoDB.

Fast Performance: Even at a very high scale, DynamoDB delivers very fast performance with low latency. It will use SSD and partitioning behind the scenes to achieve the throughput that a user specifies.

Access Control: We can integrate DynamoDB with IAM to create fine-grained access control. This can keep our data secure in DynamoDB.

Flexible: DynamoDB supports both document and key-value data structures. So it helps in providing flexibility of selecting the right architecture for our application.

Event Driven: We can also make use of AWS Lambda with DynamoDB to perform any event driven programming. This option is very useful for ETL tasks.

What is the basic Data Model in Amazon DynamoDB?

The basic Data Model in Amazon DynamoDB consists of following components:

Table: In DynamoDB, a Table is collection of data items. It is similar to a table in a Relational Database. There can be infinite number of items in a Table. There has to be one Primary key in a Table.

Item: An Item in DynamoDB is made up of a primary key or composite key and a variable number of attributes. The number of attributes in an Item is not bounded by a limit. But total size of an Item can be maximum 400 kilobytes.

Attribute: In DynamoDB, we can associate an Attribute with an Item. We can set a name as well as one or more values in an Attribute. Total size of data in an Attribute is maximum 400 kilobytes.

What are the different APIs available in Amazon DynamoDB?

Amazon DynamoDB supports both document as well as key based NoSQL databases. Due to this APIs in DynamoDB are generic enough to serve both the types.

Some of the main APIs available in DynamoDB are as follows: -

CreateTable, UpdateTable, DeleteTable, DescribeTable, ListTables, PutItem, GetItem, BatchWriteItem, BatchGetItem, UpdateItem, DeleteItem, Query & Scan

When should be use Amazon DynamoDB vs. Amazon S3?

Amazon DynamoDB is used for storing structured data. The data in DynamoDB is also indexed by a primary key for fast access. Reads and writes in DynamoDB have very low latency due to the use SSD. Amazon S3 is mainly used for storing unstructured binary large objects-based data. It does not have a fast index like DynamoDB.

So, we should use Amazon S3 for storing objects with infrequent access requirements. Another consideration is size of the data. In DynamoDB the size of an item can be maximum 400 kilobytes. Whereas Amazon S3 supports size as large as 5 terabytes for an object. Therefore, DynamoDB is more suitable for storing small objects with frequent access and S3 is ideal for storing very large objects with infrequent access.



Redshift

What is Redshift?

Redshift is a fast, fully managed, petabyte-scale data warehouse service that makes it simple and cost-effective to efficiently analyze all your data using your existing business intelligence tools.



Security, Identity & Compliance

AWS Identity & Access Management Manage User Access and Encryption Keys	Amazon Cloud Directory Create Flexible Cloud Native Directories	Amazon Cognito Identity Management for your Apps
Amazon GuardDuty Managed Threat Detection Service	Amazon Inspector Analyze Application Security	Amazon Macie Discover, Classify, and Protect your Data
AWS Certificate Manager Provision, Manage and Deploy SSL/TLS Certificates	AWS CloudHSM Hardware-based Key Storage for Regulatory Compliance	Amazon Directory Service Host and manage Active Directory
AWS Firewall Manager Central Management of Firewall Rules	AWS Key Management Service Managed Creation and Control of Encryption Keys	AWS Organizations Policy-based Management for Multiple AWS Accounts
AWS Secrets Manager Rotate, Manage and Retrieve Secrets	AWS Single Sign-on Cloud Single Sign-on (SSO) Service	AWS Shield DDoS Protection
AWS WAF Filter Malicious Web Traffic		



Security, Identity &
Compliance

Identity Access & Management

IAM Highlights

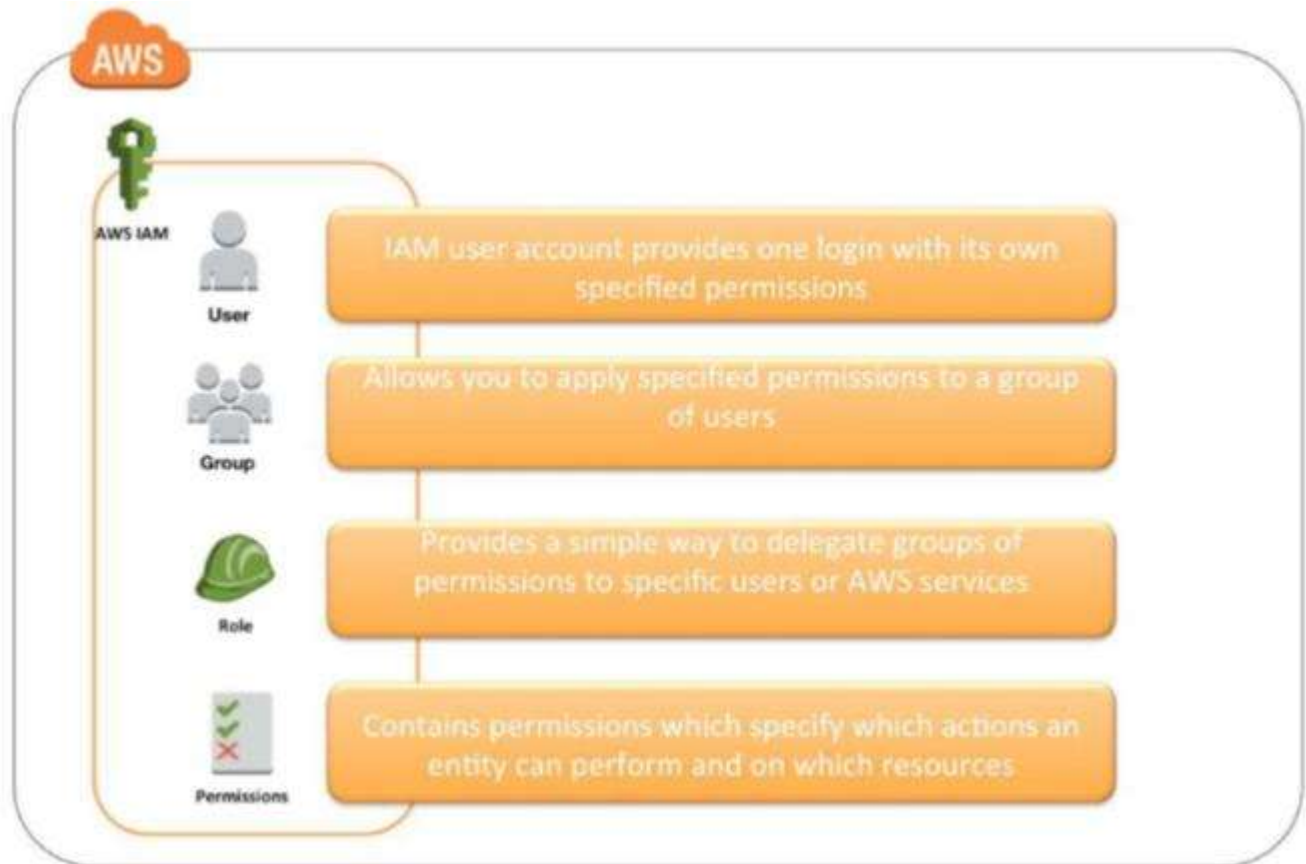
- IAM consists of the following: -
 - Users
 - Groups (A way to group our users and apply policies to them collectively)
 - Roles
 - Policy Documents (using JSON)
- IAM is Universal. It does not apply to regions at this time.
- The "**root account**" is simply the account created when first setup your AWS account. It has complete Admin access.
- New Users have No Permissions when first created
- New Users are assigned "**Access Key ID & Secret Access Keys**" when first create
- These are not the same as a password, and you cannot use the Access key ID & Secret Access Key to Login in to the console. You can use this to access AWS via the APIs and Command Line however
- Always setup Multifactor Authentication on your root account
- You can create and customize your own password rotation policies
- Roles are more secure than storing your access key and secret access key on individual EC2 instances
- Roles are easier to manage
- Roles can be assigned to an EC2 instance AFTER it has been provisioned using both the command line and AWS Console
- Roles are universal, you can use them in any region

Share the IAM Configuration Step by Step?

To Configure and use AWS IAM Service

Topology

AWS IAM Entities



Pre-requisites

User should have AWS root account

To Configure IAM with following task

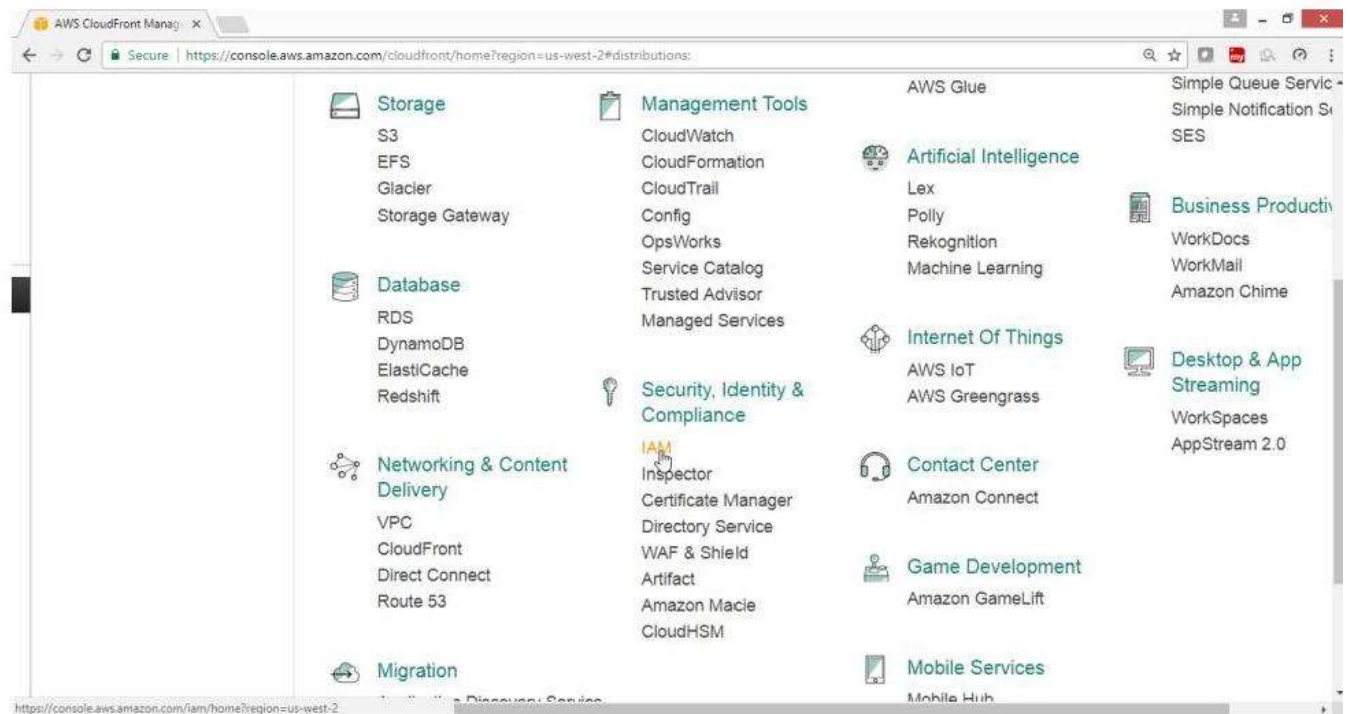
- Create IAM users, assign password, and change password policy
- Create IAM groups
- Add users to a group
- Add policies to Groups and Users
- Create your own policies
- Users Login to Sign-in page
- Deleting users and groups

1)To create user, assign password, change password policy

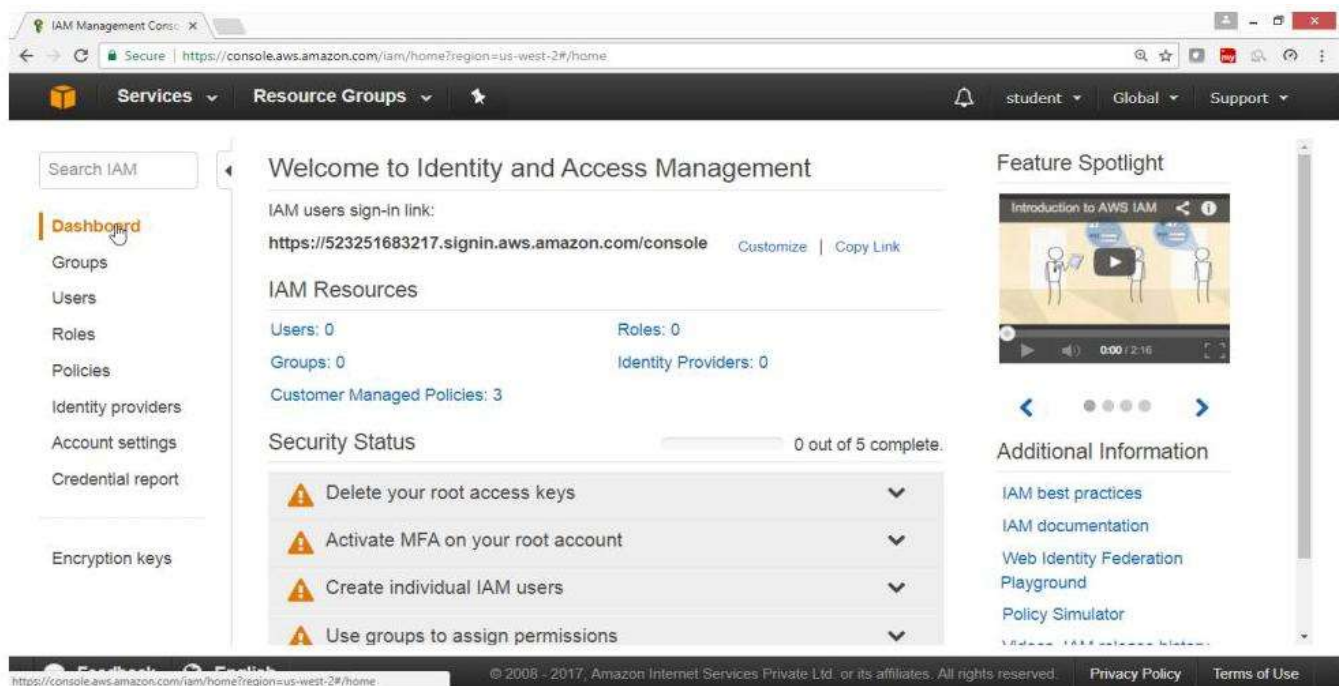
Open AWS console

Select Security, Identity & compliance

Click on **IAM service**



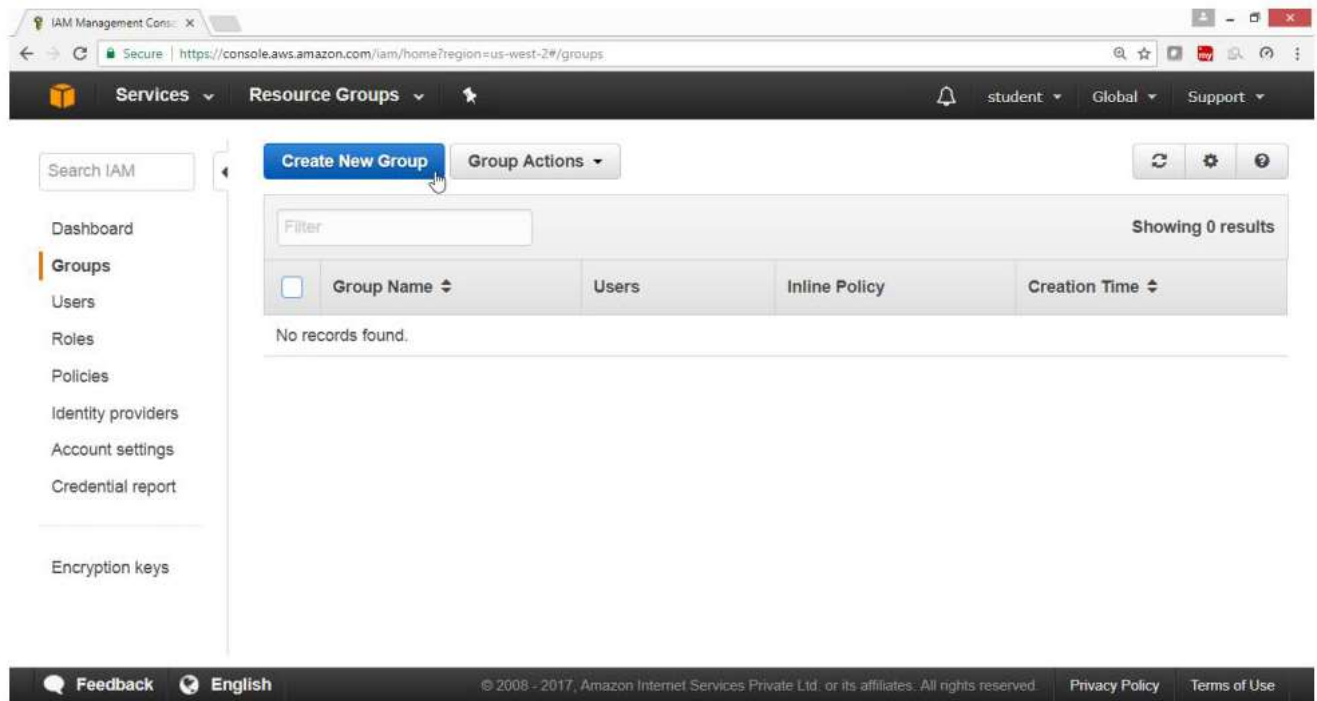
IAM Dashboard panel available



2)To Manage Groups and applying policies

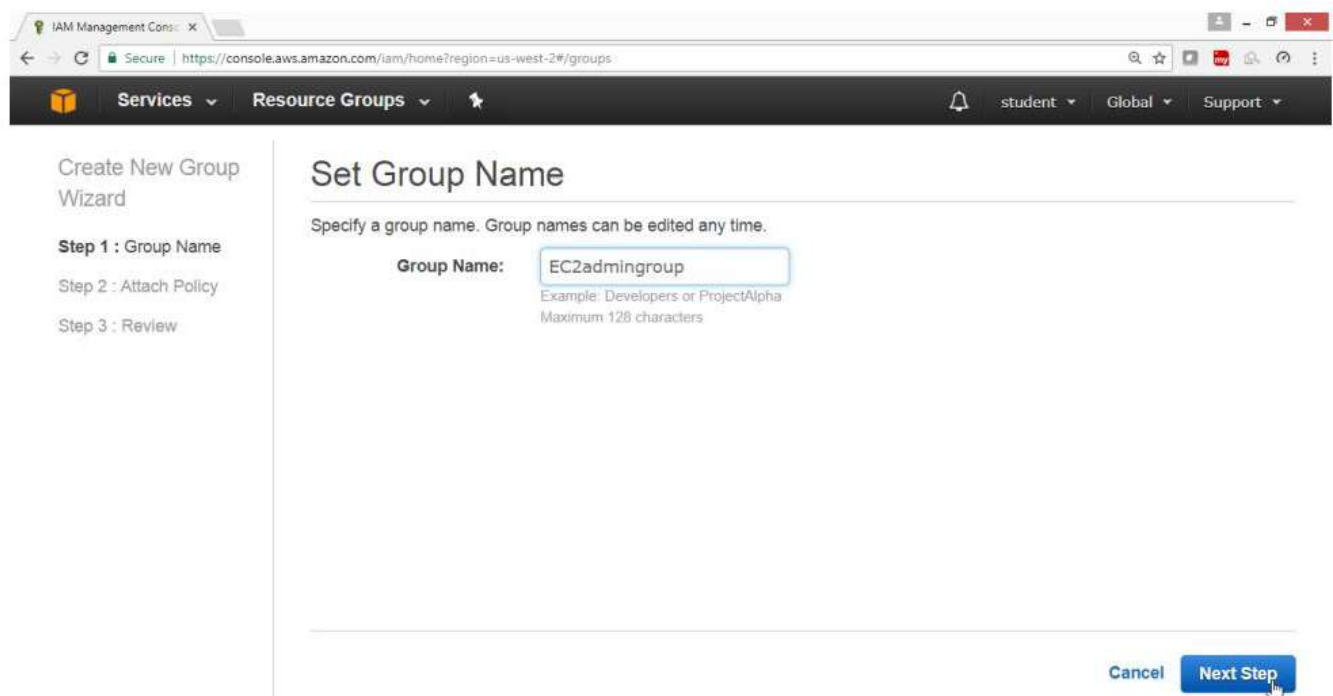
From IAM Dashboard, select "**Groups**"

Click on "**Create New Group**" Button



Give Group Name -> **EC2admingroup**

Click on "**Next Step**" Button



In Filter Type -> **EC2f**

Select check box for "**AmazonEC2FullAccess**"

Click On "**Next Step**" Button

IAM Management Console

Services Resource Groups

Create New Group Wizard

Step 1: Group Name

Step 2: Attach Policy

Step 3: Review

Attach Policy

Select one or more policies to attach. Each group can have up to 10 policies attached.

Filter: Policy Type Showing 2 results

	Policy Name	Attached Entities	Creation Time	Edited Time
<input checked="" type="checkbox"/>	AmazonEC2FullAccess	0	2015-02-07 00:10 UTC...	2015-02-07 00:10 ...
<input type="checkbox"/>	AmazonEC2FullAccess...	0	2017-06-17 16:33 UTC...	2017-06-17 16:33 ...

Cancel Previous Next Step

Click on "Create Group"

IAM Management Console

Services Resource Groups

Create New Group Wizard

Step 1: Group Name

Step 2: Attach Policy

Step 3: Review

Review

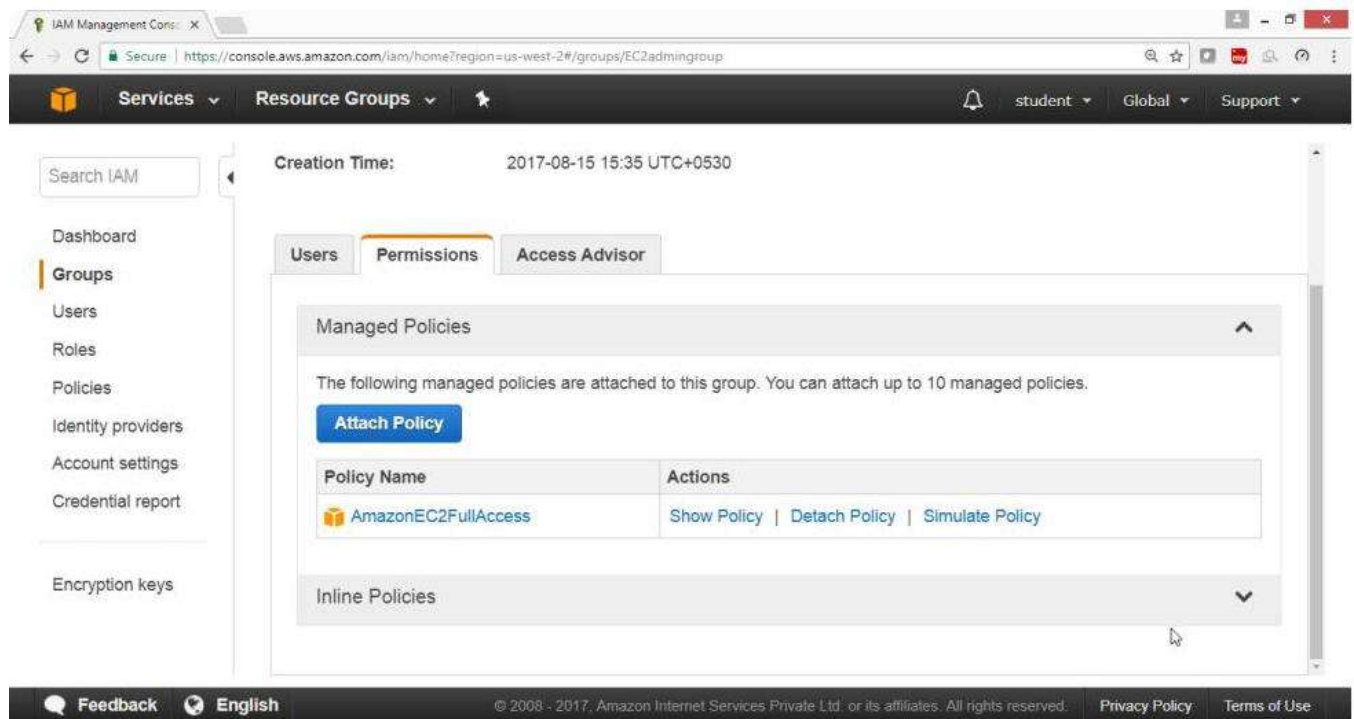
Review the following information, then click **Create Group** to proceed.

Group Name	EC2admingroup	Edit Group Name
Policies	arn:aws:iam::aws:policy/AmazonEC2FullAccess	Edit Policies

Cancel Previous Create Group

Verify

Group EC2admingrp got created with AmazonEC2FullAccess Policy



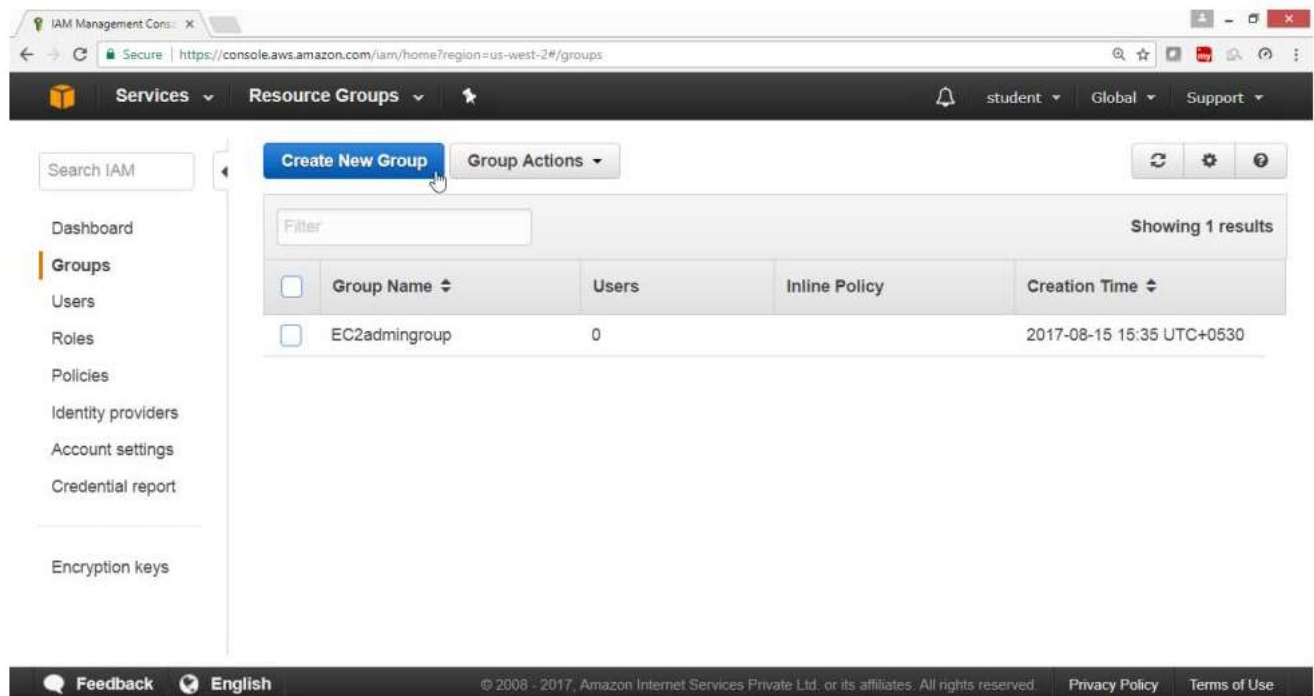
The screenshot shows the AWS IAM console interface. The left sidebar contains navigation links: Search IAM, Dashboard, Groups (selected), Users, Roles, Policies, Identity providers, Account settings, Credential report, and Encryption keys. The main content area is titled 'Creation Time: 2017-08-15 15:35 UTC+0530'. Below this, there are tabs for 'Users', 'Permissions' (selected), and 'Access Advisor'. The 'Permissions' tab shows a section for 'Managed Policies' with the text: 'The following managed policies are attached to this group. You can attach up to 10 managed policies.' Below this text is a blue 'Attach Policy' button. A table lists the attached policies:

Policy Name	Actions
AmazonEC2FullAccess	Show Policy Detach Policy Simulate Policy

Below the table is a section for 'Inline Policies' with a downward arrow. The bottom of the console shows a footer with 'Feedback', 'English', copyright information, and links to 'Privacy Policy' and 'Terms of Use'.

Now again create Another Group

Click on "Create Group" Button

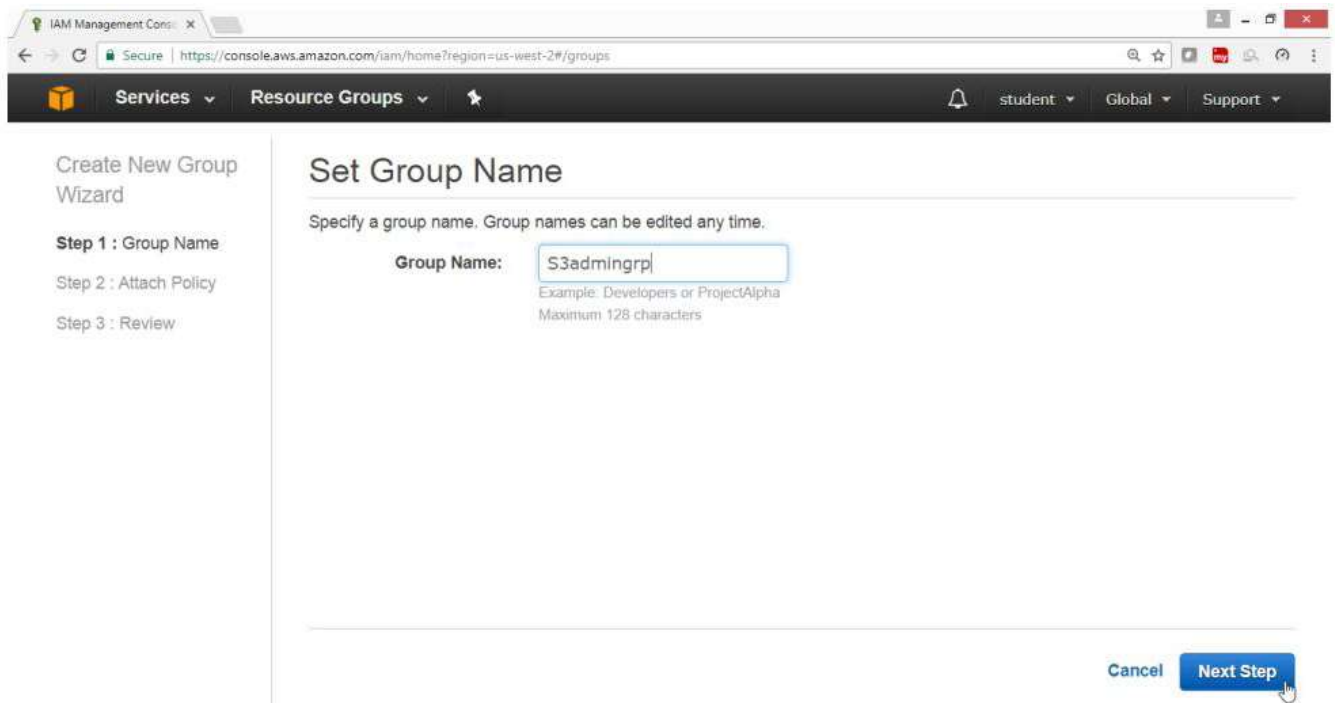


The screenshot shows the AWS IAM console interface. The left sidebar contains navigation links: Search IAM, Dashboard, Groups (selected), Users, Roles, Policies, Identity providers, Account settings, Credential report, and Encryption keys. The main content area has a 'Create New Group' button highlighted with a mouse cursor. Below this button is a 'Group Actions' dropdown menu. A table below shows the list of groups:

Group Name	Users	Inline Policy	Creation Time
EC2admingrp	0		2017-08-15 15:35 UTC+0530

The bottom of the console shows a footer with 'Feedback', 'English', copyright information, and links to 'Privacy Policy' and 'Terms of Use'.

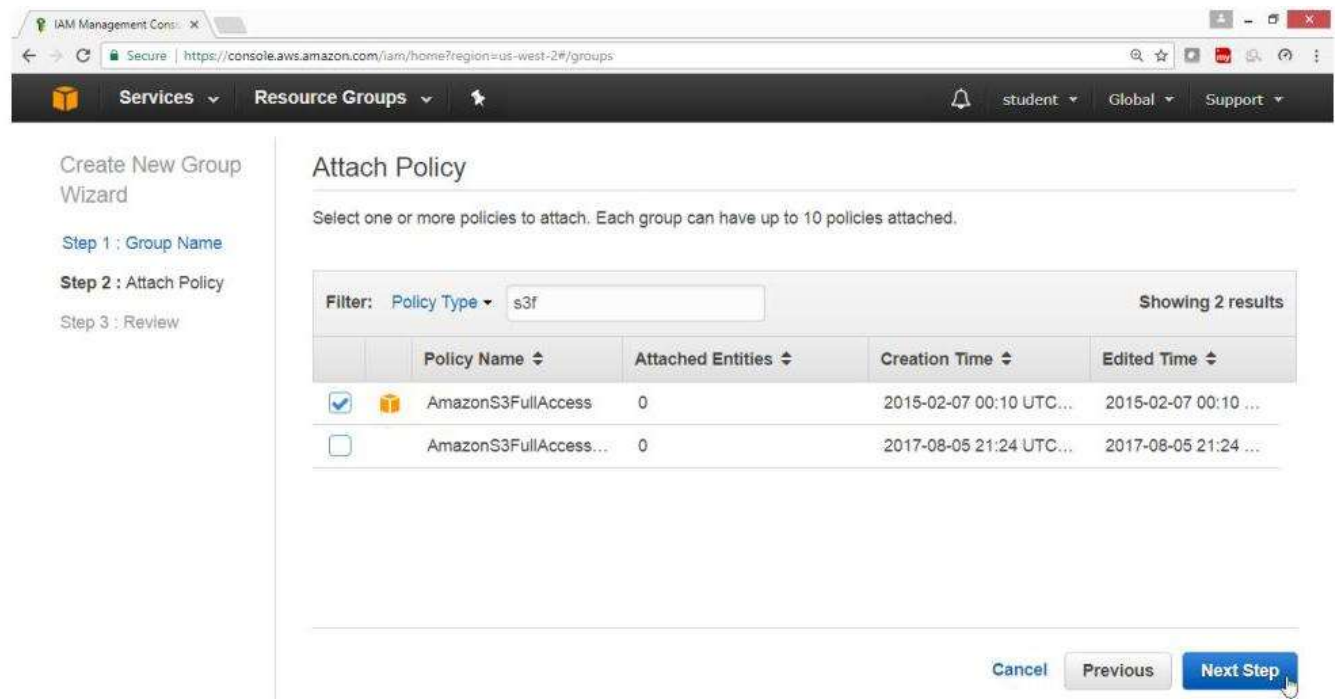
To create a group with S3FullAccess



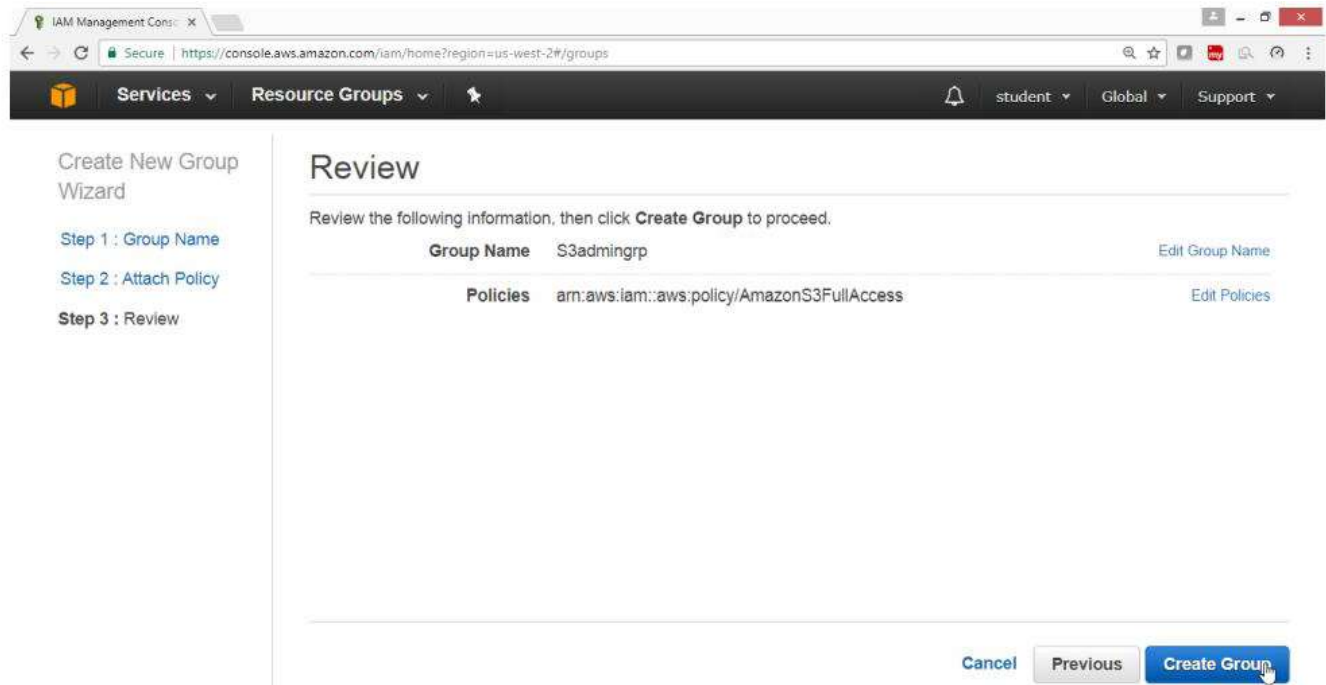
In Filter Type -> **S3f**

Select check box for "**AmazonS3FullAccess**"

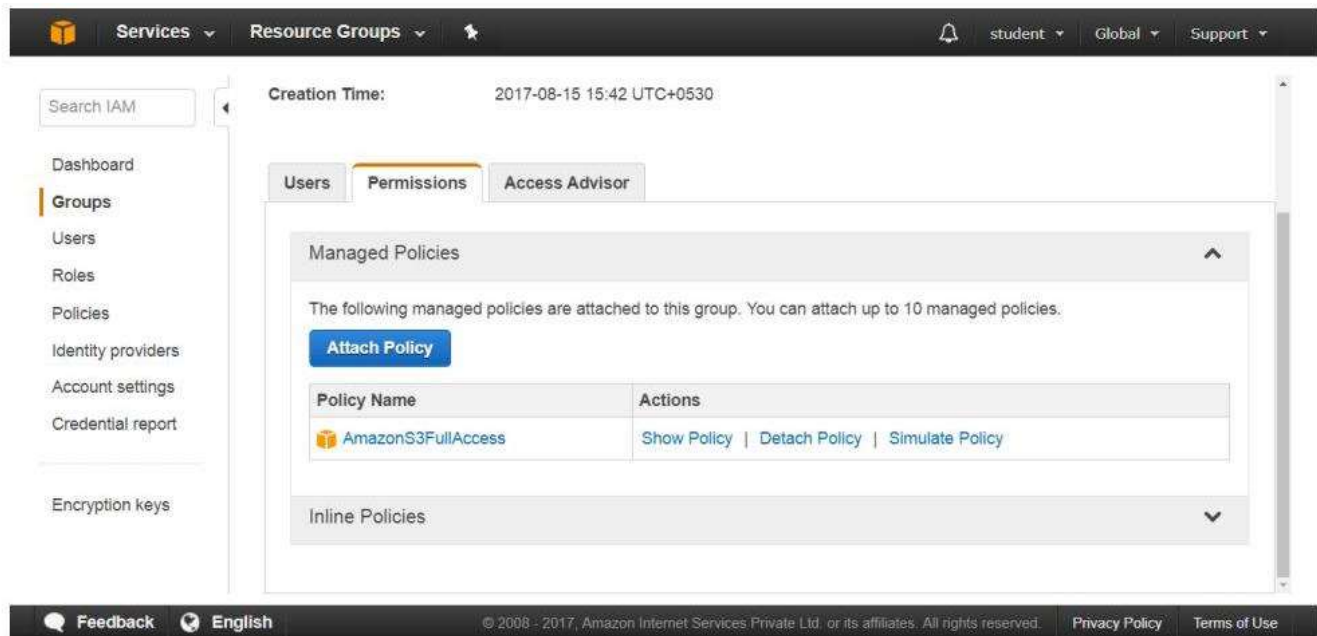
Click On "**Next Step**" Button



Create on "Create Group" Button



Verify EC2admingroup & S3admingr groups got created



Verify S3 policy is attached

Services

Resource Groups

studentGlobalSupport

Search IAM

Dashboard

Groups

Users

Roles

Policies

Identity providers

Account settings

Credential report

Encryption keys

Creation Time: 2017-08-15 15:42 UTC+0530

UsersPermissionsAccess Advisor

Managed Policies

The following managed policies are attached to this group. You can attach up to 10 managed policies.

Attach Policy

Policy Name	Actions
AmazonS3FullAccess	Show Policy Detach Policy Simulate Policy

Inline Policies

FeedbackEnglish

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

Create user tom and join to EC2admingroup

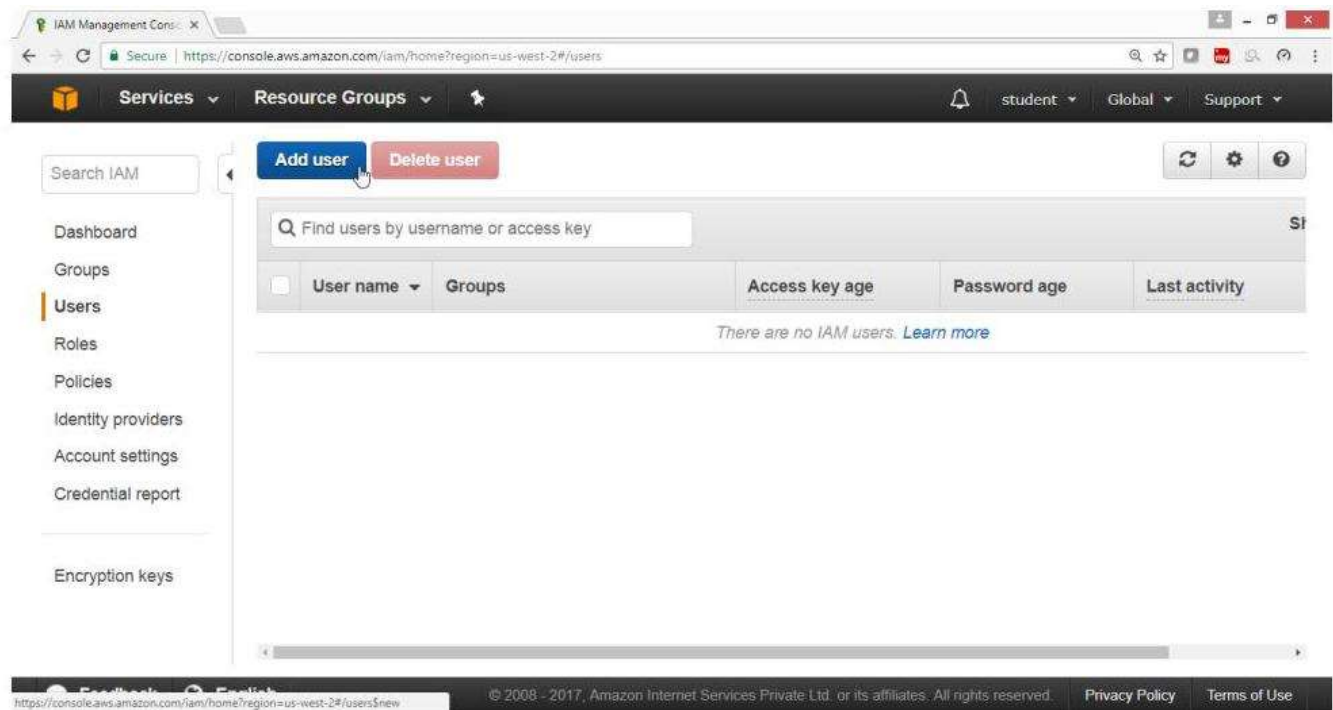
Create user john and join to S3admingroup

Create a user sai add EC2fullaccess and S3fullaccess Policy

From IAM dashboard

Select "Users"

Click on "ADD Users" button



Scenario 1)

Create user tom and join to [EC2admingroup](#)

For User name ->tom

For Access type -> [AWS Management Console Access](#)

Drag Down

The screenshot shows the AWS IAM 'Add user' wizard. At the top, there's a navigation bar with 'Services', 'Resource Groups', and a search icon. On the right, there are links for 'student', 'Global', and 'Support'. Below the navigation bar, the title 'Add user' is followed by a progress indicator with four steps: 1. Details (active), 2. Permissions, 3. Review, and 4. Complete. The main section is titled 'Set user details' and includes a sub-header: 'You can add multiple users at once with the same access type and permissions. [Learn more](#)'. There is a text input field for 'User name*' containing the text 'tom'. Below the input field is a blue link with a plus icon that says 'Add another user'. The next section is titled 'Select AWS access type' and includes a sub-header: 'Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)'. Under this section, there are two radio button options for 'Access type*': 'Programmatic access' (unchecked) and 'AWS Management Console access' (checked). The 'Programmatic access' option has a description: 'Enables an access key ID and secret access key for the AWS API, CLI, SDK, and other development tools.' The 'AWS Management Console access' option has a description: 'Enables a password that allows users to sign-in to the AWS Management Console.' At the bottom of the page, there is a footer bar with 'Feedback', 'English', copyright information '© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.', and links for 'Privacy Policy' and 'Terms of Use'.

Services ▾ Resource Groups ▾

student ▾ Global ▾ Support ▾

Add user

1 Details 2 Permissions 3 Review 4 Complete

Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name* tom

[+ Add another user](#)

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Access type* ☐ Programmatic access
Enables an access key ID and secret access key for the AWS API, CLI, SDK, and other development tools.

☒ AWS Management Console access
Enables a password that allows users to sign-in to the AWS Management Console.

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

For Console Password->*****

Click on Next "**Permissions**" button

user: development1, iuser:

☒ **AWS Management Console access**
Enables a password that allows users to sign-in to the AWS Management Console.

Console password* ☐ Autogenerated password ☒ Custom password

☐ Show password

Require password reset ☐ User must create a new password at next sign-in.
Users automatically get the `IAMUserChangePassword` policy to allow them to change their own password.

* Required

Cancel **Next: Permissions**

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Under Group column

Select **EC2admingroup**

Click on "**Next Review**"

Details

Group	Attached policies
<input checked="" type="checkbox"/> EC2admingroup	AmazonEC2FullAccess
<input type="checkbox"/> S3admingrp	AmazonS3FullAccess

Cancel Previous **Next: Review**

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Verify users detail

Click on "**Create User**" Button

IAM Management Console

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name	tom
AWS access type	AWS Management Console access - with a password
Console password type	Custom
Require password reset	No

Permissions summary

The user shown above will be added to the following groups.

Type	Name
Group	EC2admingroup

[Cancel](#) [Previous](#) [Create user](#)

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Down the .csv file

IAM Management Console

Details Permissions Review **Complete**

Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://523251683217.signin.aws.amazon.com/console>

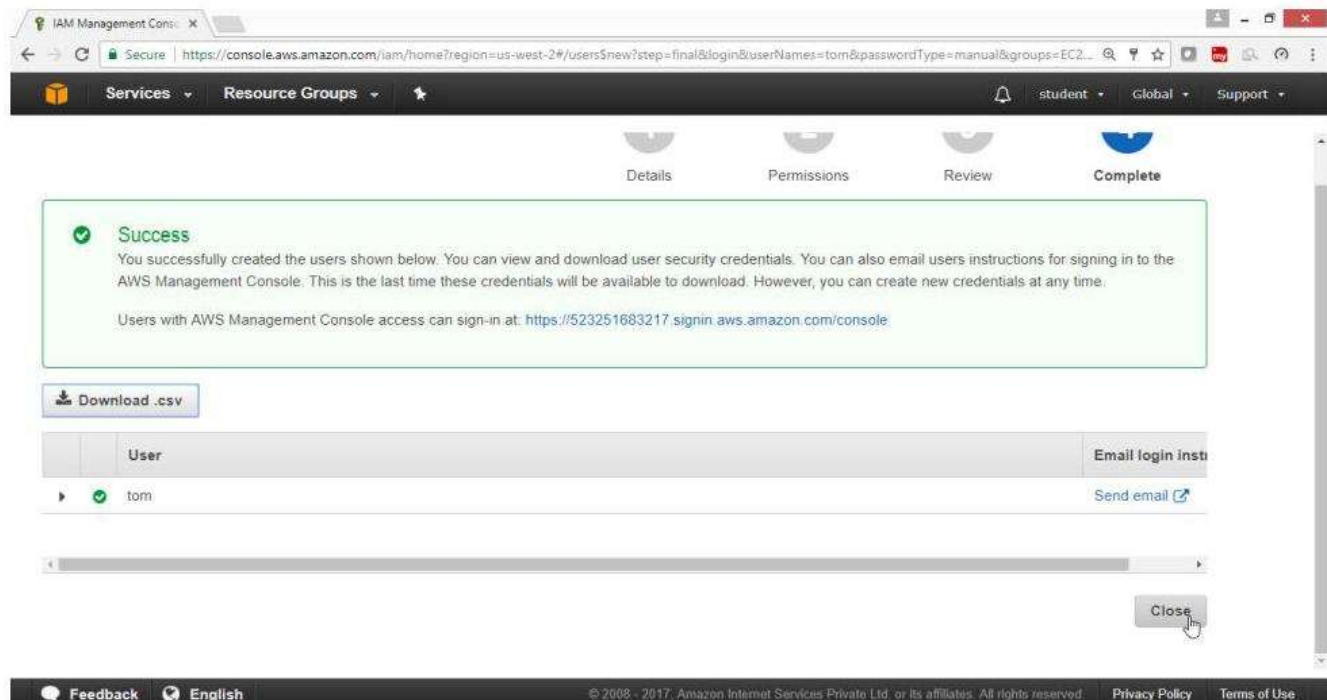
[Download .csv](#)

	User	Email login instructions
▶	tom	Send email

[Close](#)

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Click on close button

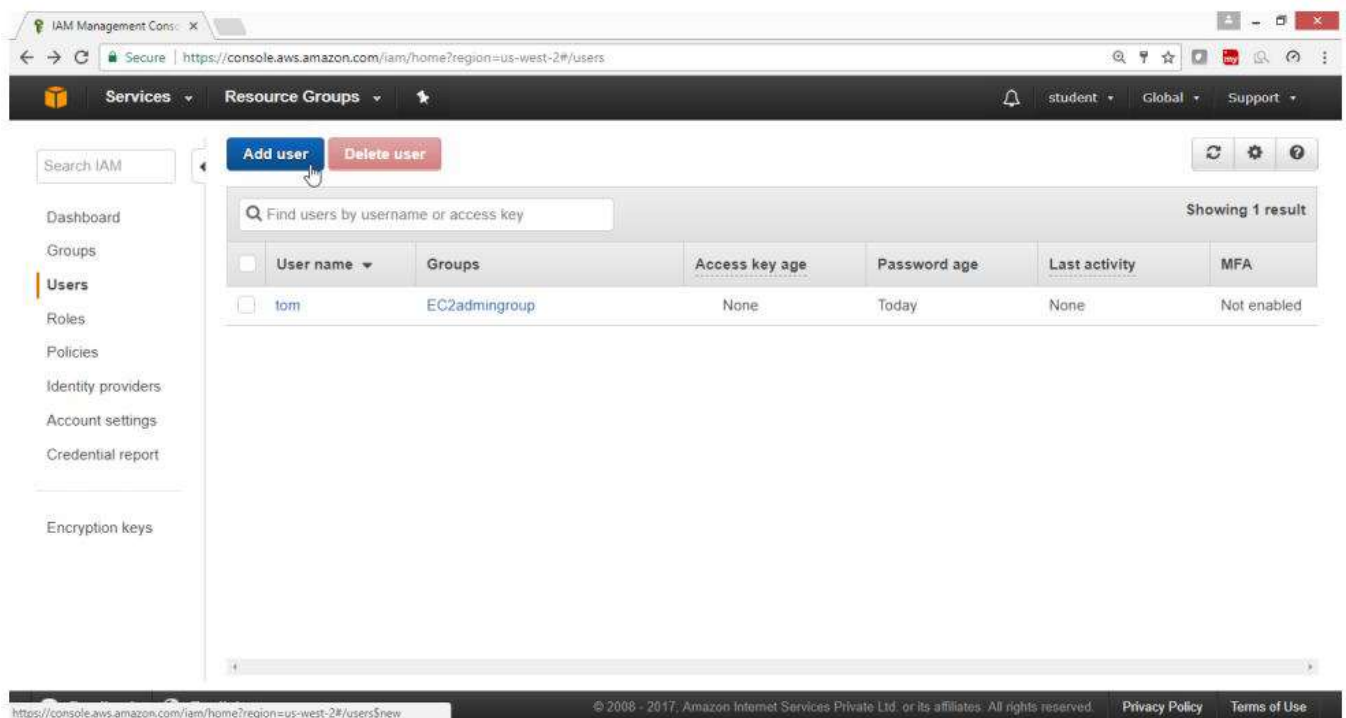


Scenario 2

Create user john and join to "S3admingroup"

Select user

Click on "Add User" Button



For user name -> john

For Access Type -> john

For Console password -> AWS Management Console Access

For Console Password ->*****

Drag Down

IAM Management Console

Services Resource Groups

student Global Support

User name* john

+ Add another user

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Access type* ☐ Programmatic access
Enables an access key ID and secret access key for the AWS API, CLI, SDK, and other development tools.

☒ AWS Management Console access
Enables a password that allows users to sign-in to the AWS Management Console.

Console password* ☐ Autogenerated password

☒ Custom password

.....

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Click on "Next Permission" Button

The screenshot shows the 'Details' step of the 'Create New User' wizard in the AWS IAM console. The 'AWS management console access' checkbox is checked, indicating that the user will be able to sign in to the AWS console. Under 'Console password*', the 'Custom password' radio button is selected, and a password field with masked characters is visible. The 'Require password reset' checkbox is unchecked. At the bottom right, the 'Next: Permissions' button is highlighted with a mouse cursor. The footer includes a 'Feedback' link, 'English' language selection, and copyright information for 2008-2017.

AWS management console access
Enables a password that allows users to sign-in to the AWS Management Console.

Console password*

- ☐ Autogenerated password
- ☒ Custom password

☐ Show password

Require password reset ☐ User must create a new password at next sign-in
Users automatically get the `IAMUserChangePassword` policy to allow them to change their own password.

* Required

Cancel **Next: Permissions**

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Select **S3admingrp**

Click on "Next Review" Button

The screenshot shows the 'Permissions' step of the 'Create New User' wizard. A table lists the groups and their attached policies. The 'S3admingrp' group is selected with a checkmark, and its attached policy is 'AmazonS3FullAccess'. The 'EC2admingrp' group is not selected. At the bottom right, the 'Next: Review' button is highlighted with a mouse cursor. The footer includes a 'Feedback' link, 'English' language selection, and copyright information for 2008-2017.

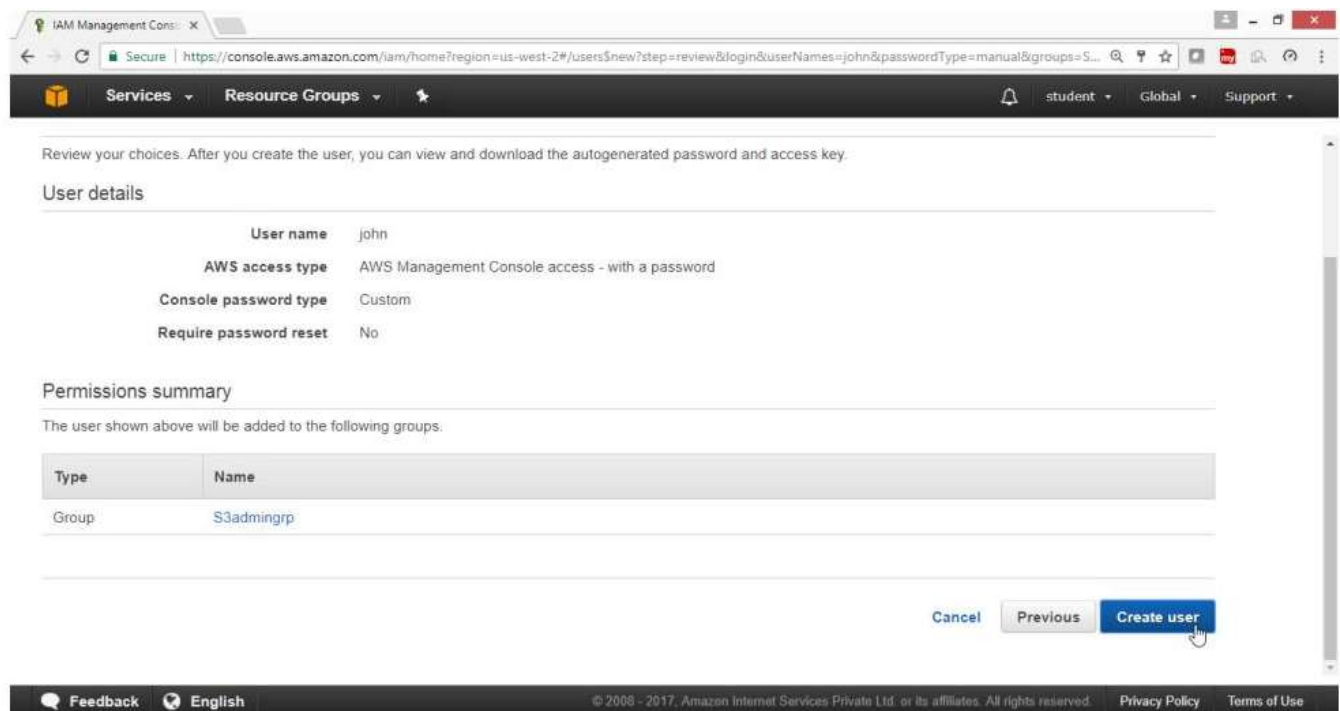
Group	Attached policies
<input type="checkbox"/> EC2admingrp	AmazonEC2FullAccess
<input checked="" type="checkbox"/> S3admingrp	AmazonS3FullAccess

Cancel Previous **Next: Review**

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

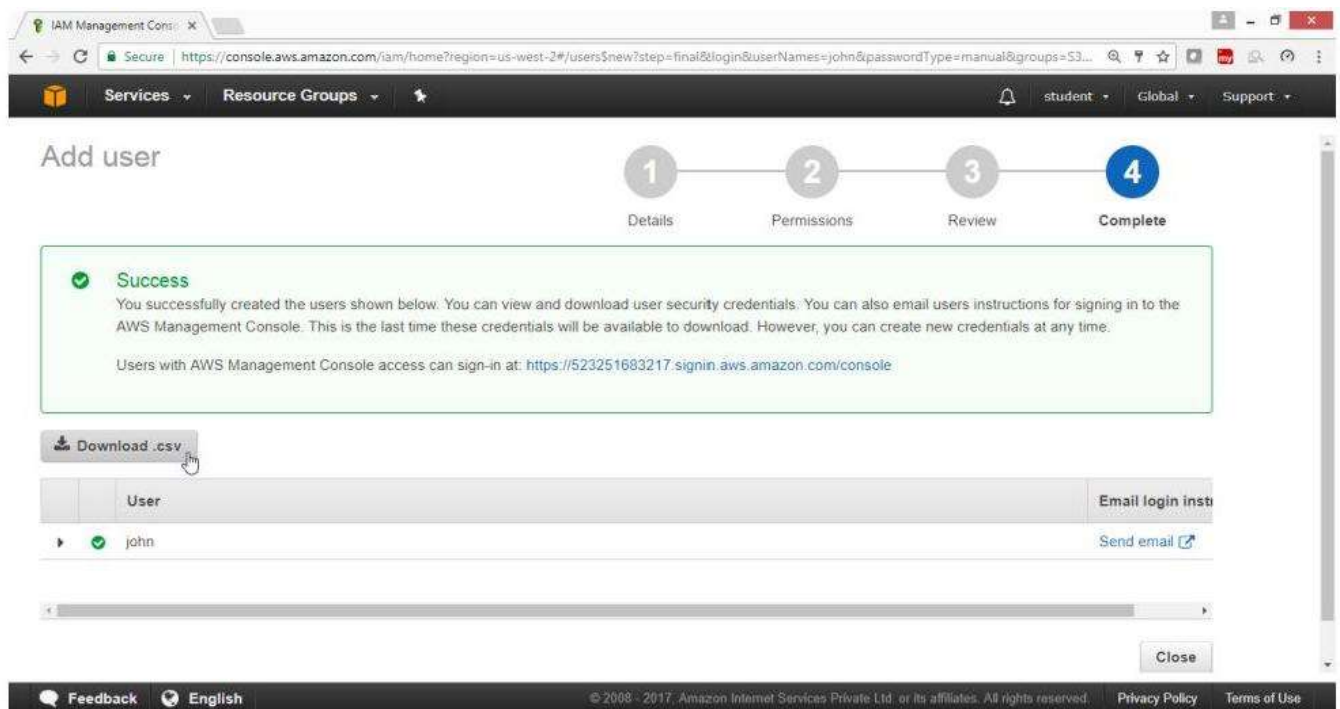
Verify user details

Click on "Create User" Button



Download the [.csv file](#)

Click on [Close](#) button



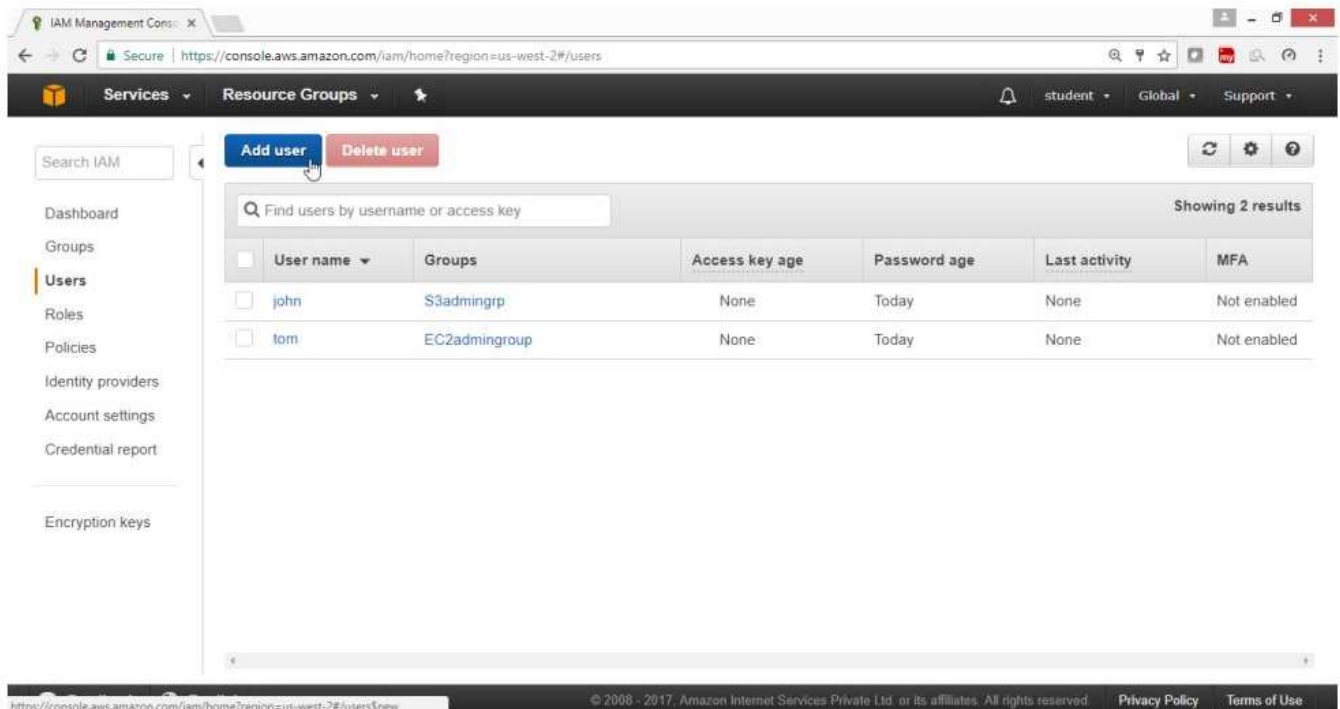
Scenario 3

Add a user individual user sai without joining to any group

Attach EC2FullAccess and [S3FullAccess Policy](#)

Select User

Click on "Add User" Button

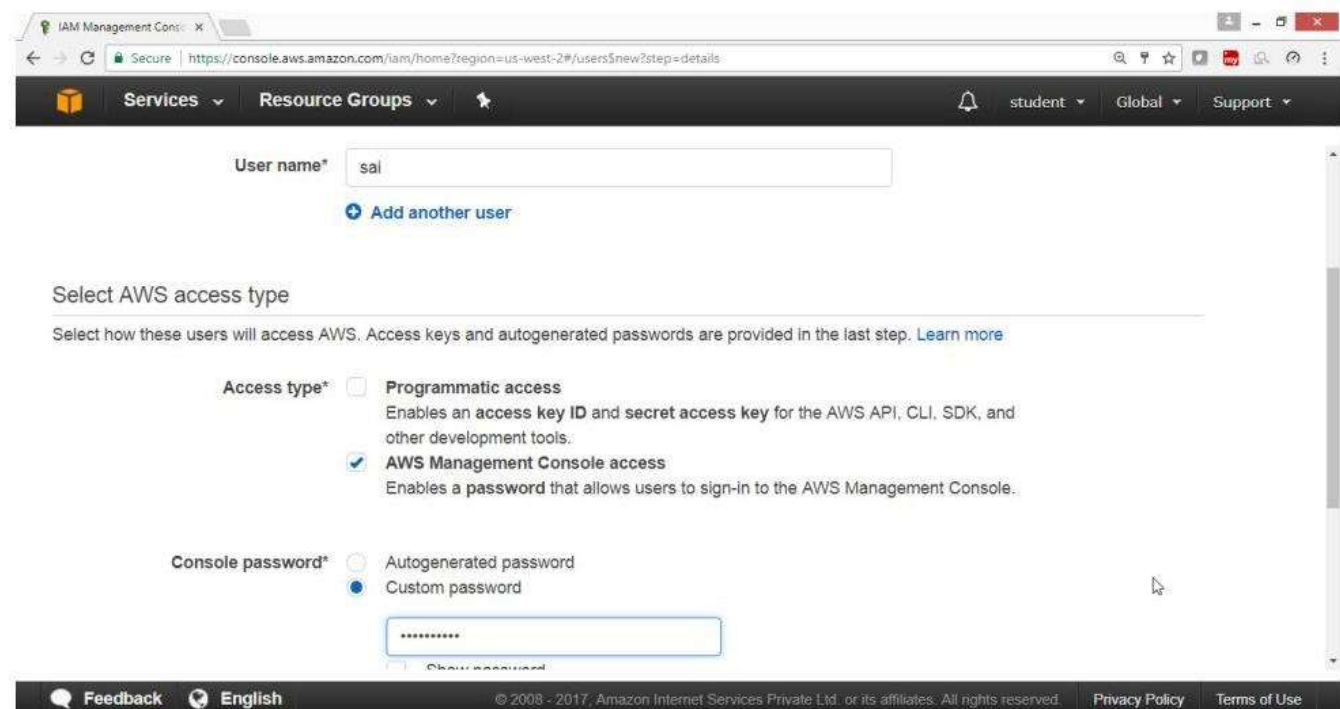


For User Name -> sai

For Access Type -> AWS Management Console Access

For Console Password -> *****

Drag Down



Click on "Next Permission" Button

IAM Management Console

Secure | https://console.aws.amazon.com/iam/home?region=us-west-2#/users\$new?step=details

Services Resource Groups

student Global Support

other development tools.

☒ **AWS Management Console access**
Enables a password that allows users to sign-in to the AWS Management Console.

Console password*

☐ Autogenerated password

☒ Custom password

.....

☐ Show password

Require password reset

☐ User must create a new password at next sign-in
Users automatically get the [IAMUserChangePassword](#) policy to allow them to change their own password.

* Required

Cancel

Next: Permissions

Feedback English

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

Privacy Policy Terms of Use

Click on "Attach Existing Policies Directly" box

1 Details 2 Permissions 3 Review 4 Complete

Set permissions for sai

Add user to group Copy permissions from existing user Attach existing policies directly

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

In Filter type search for **ec2f**

Select **AmazonEC2FullAccess** Check box

existing user directly

Attach one or more existing policies directly to the user or create a new policy. [Learn more](#)

Create policy Refresh

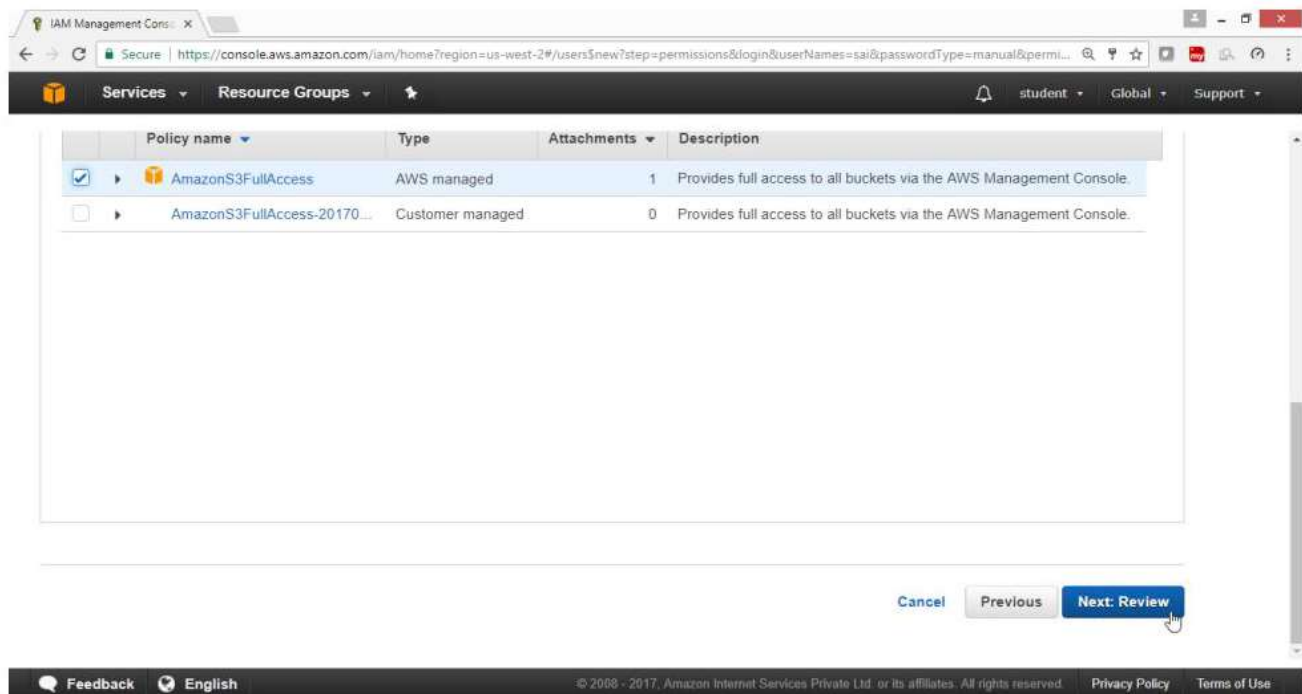
Filter: Policy type ec2f Showing 2 results

	Policy name	Type	Attachments	Description
<input checked="" type="checkbox"/>	AmazonEC2FullAccess	AWS managed	1	Provides full access to Amazon EC2 via the AWS Man...
<input type="checkbox"/>	AmazonEC2FullAcce...	Customer managed	0	Provides full access to Amazon EC2 via the AWS Man...

In Filter type search for s3f

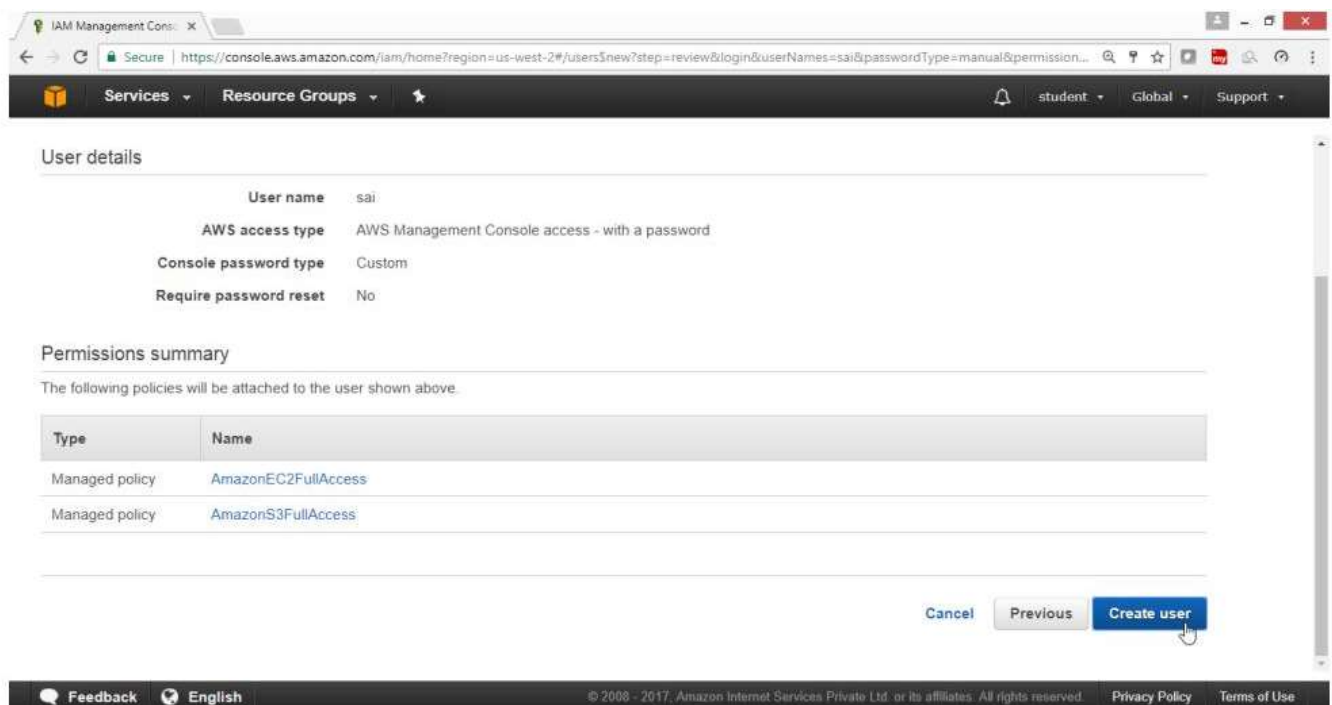
Select AmazonS3FullAccess check box

Click on "Next Review" Button



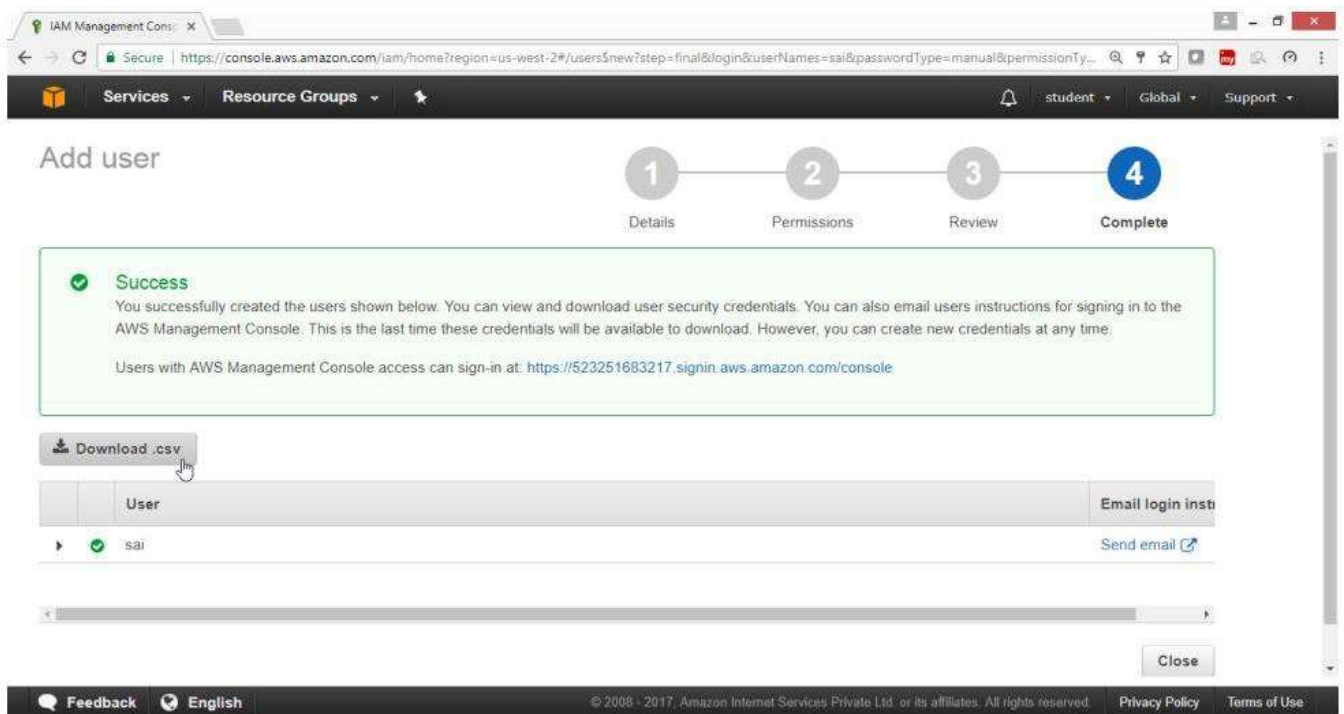
Verify user detail

Click on "Create User" button



Download the .csv file

Click on Close button

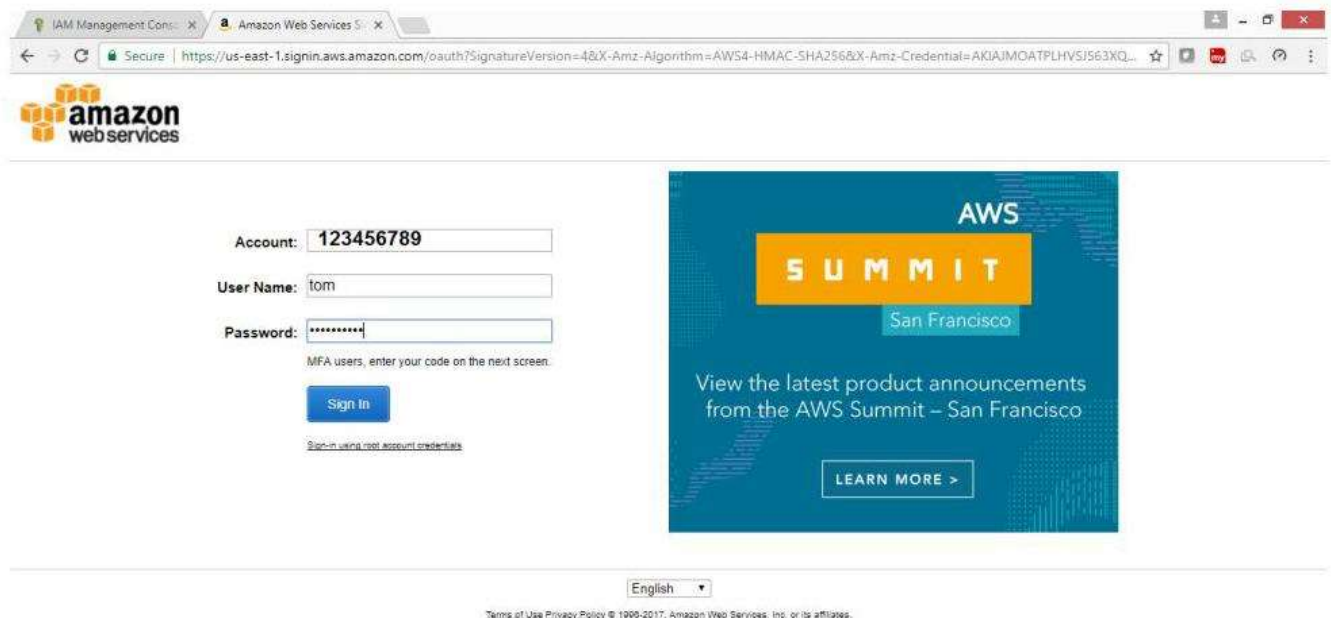


To verify whether user can access particular service

Login as tom user

- Provide the following url in Browser
- <https://123456789.signin.aws.amazon.com/console>

Click on Sign Button



User tom is not having S3 access

Click on S3 Verify the access

IAM Management Console

AWS Management Console

Secure | https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2#

Services

Resource Groups

tom @ 123456789

Oregon

History

Console Home

S3

IAM

CloudFront

VPC

EC2

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Compute

EC2

EC2 Container Service

Lightsail

Elastic Beanstalk

Lambda

Batch

Developer Tools

CodeStar

CodeCommit

CodeBuild

CodeDeploy

CodePipeline

X-Ray

Analytics

Athena

EMR

CloudSearch

Elasticsearch Service

Kinesis

Data Pipeline

QuickSight

AWS Glue

Storage

S3

EFS

Glacier

Management Tools

CloudWatch

CloudFormation

CloudTrail

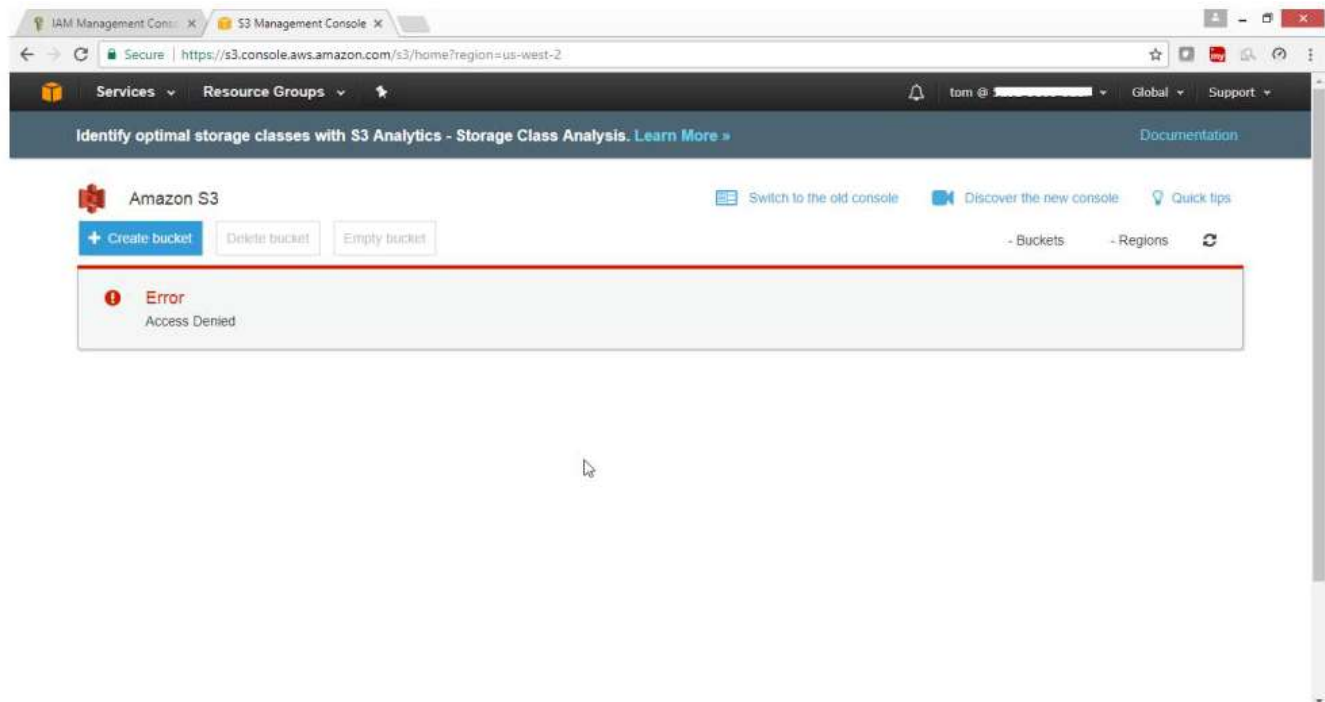
Artificial Intelligence

Lex

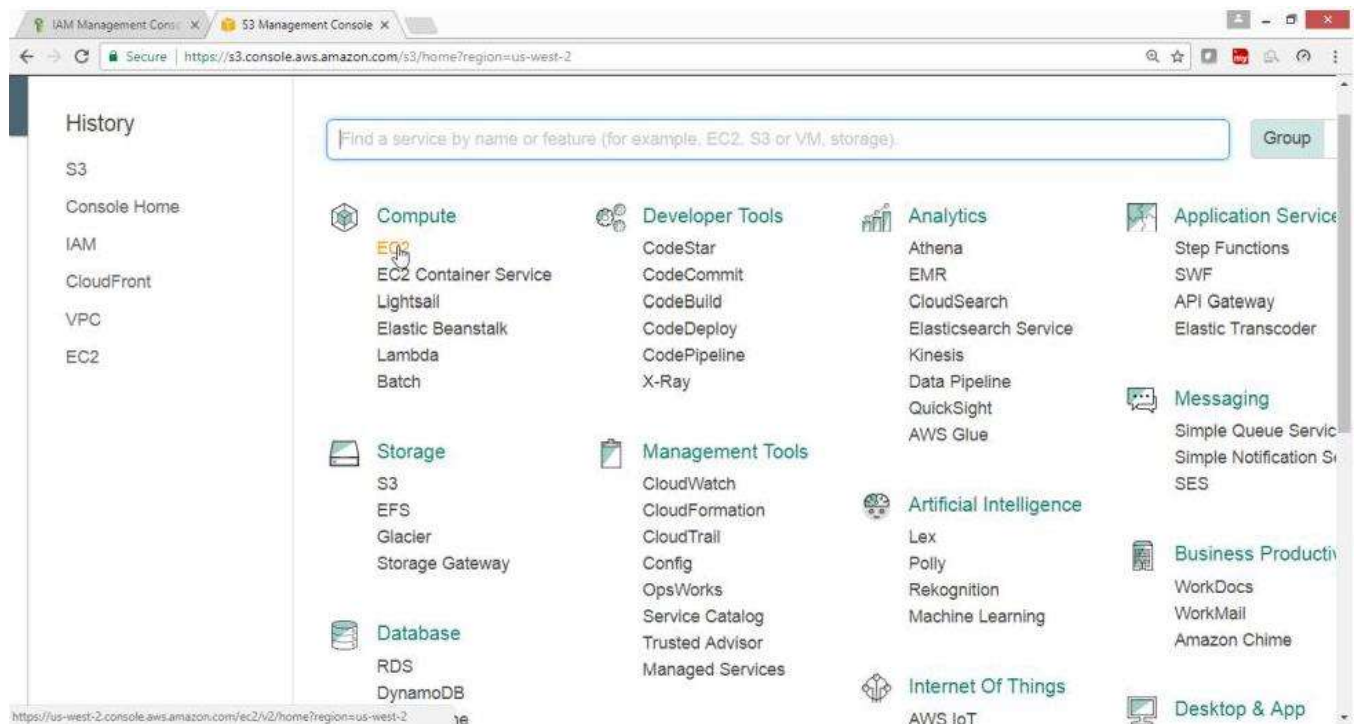
https://s3.console.aws.amazon.com/s3/home?region=us-west-2

Verification

Error Access Denied



Now Select EC2 Service



Verification

User tom can access EC2 service

IAM Management Console

Amazon Web Services

EC2 Management Console

Securehttps://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#

Services

Resource Groups

tom @

Oregon

Support

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Spot Requests

Reserved Instances

Scheduled Instances

Dedicated Hosts

IMAGES

AMIs

Resources

You are using the following Amazon EC2 resources in the US West (Oregon) region:

1 Running Instances

0 Elastic IPs

0 Dedicated Hosts

1 Snapshots

1 Volumes

0 Load Balancers

3 Key Pairs

6 Security Groups

0 Placement Groups

Create Instance

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

Account Attributes

Supported Platforms

VPC

Default VPC

vpc-89c341ee

Resource ID length management

Additional Information

[Getting Started Guide](#)

[Documentation](#)

Feedback

English

Privacy Policy

Terms of Use

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

https://docs.aws.amazon.com/console/ec2/EC2_GetStarted.html

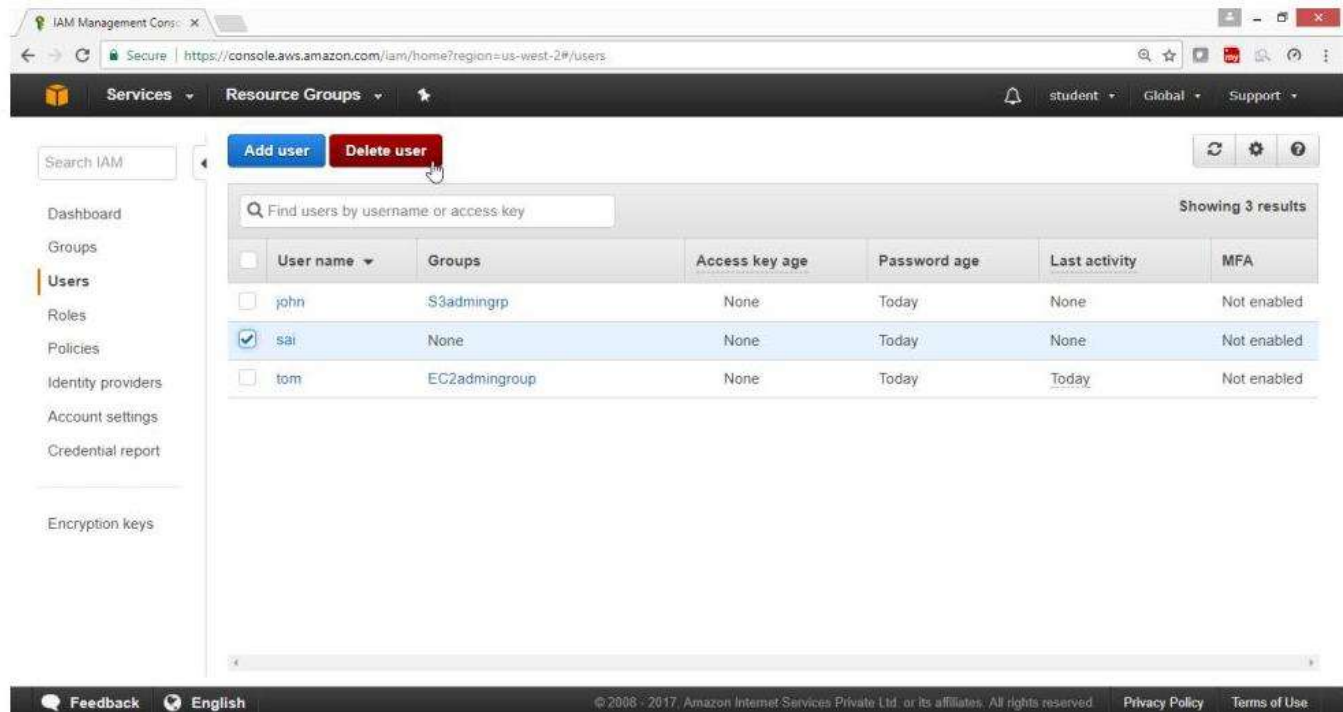
Similarly check for user john

To Delete users and groups

From IAM dashboard, select "Users"

Select the users, drop down "Action" Button

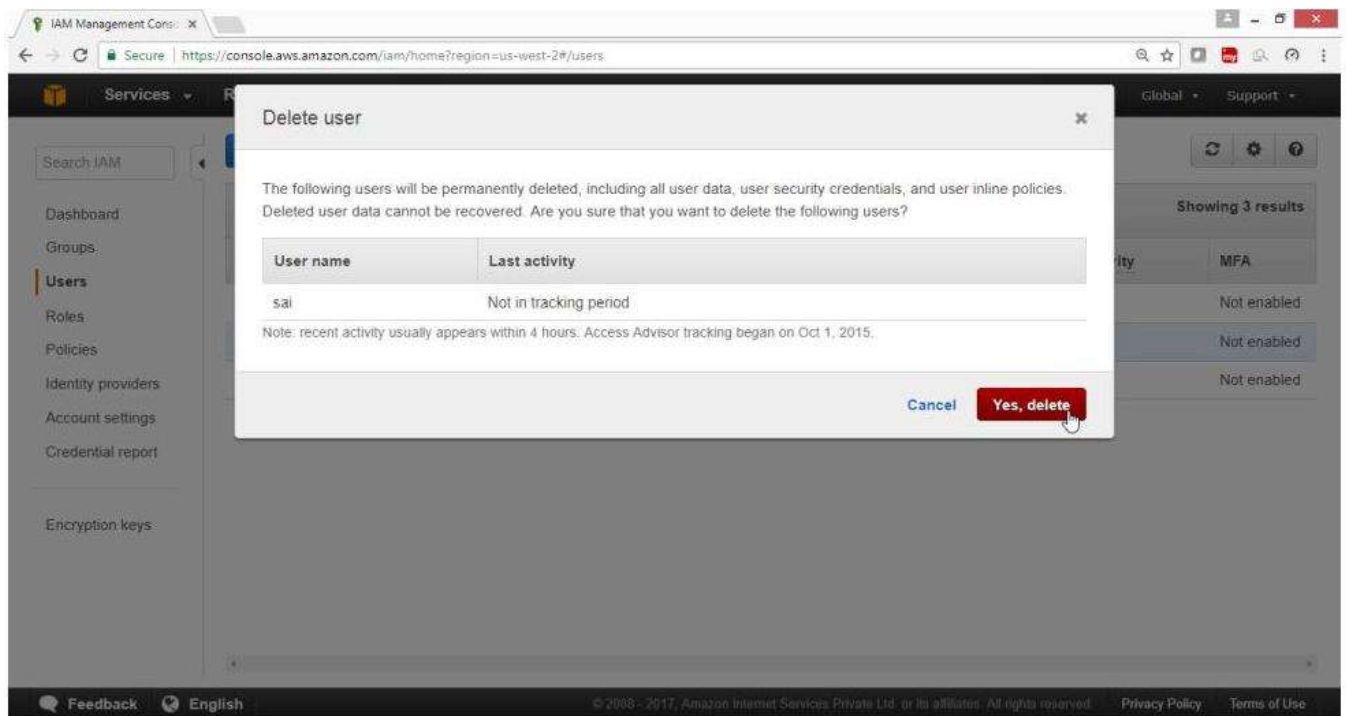
Click on "Delete Users" button



The screenshot shows the AWS IAM Management Console interface. The left sidebar contains navigation links: Dashboard, Groups, Users (selected), Roles, Policies, Identity providers, Account settings, Credential report, and Encryption keys. The main content area displays the 'Users' page with a search bar and two buttons: 'Add user' and 'Delete user'. Below the buttons is a table of users. The table has columns: User name, Groups, Access key age, Password age, Last activity, and MFA. Three users are listed: john, sai, and tom. User 'sai' is selected, indicated by a blue highlight and a checked checkbox. The 'Delete user' button is highlighted with a red box and a mouse cursor.

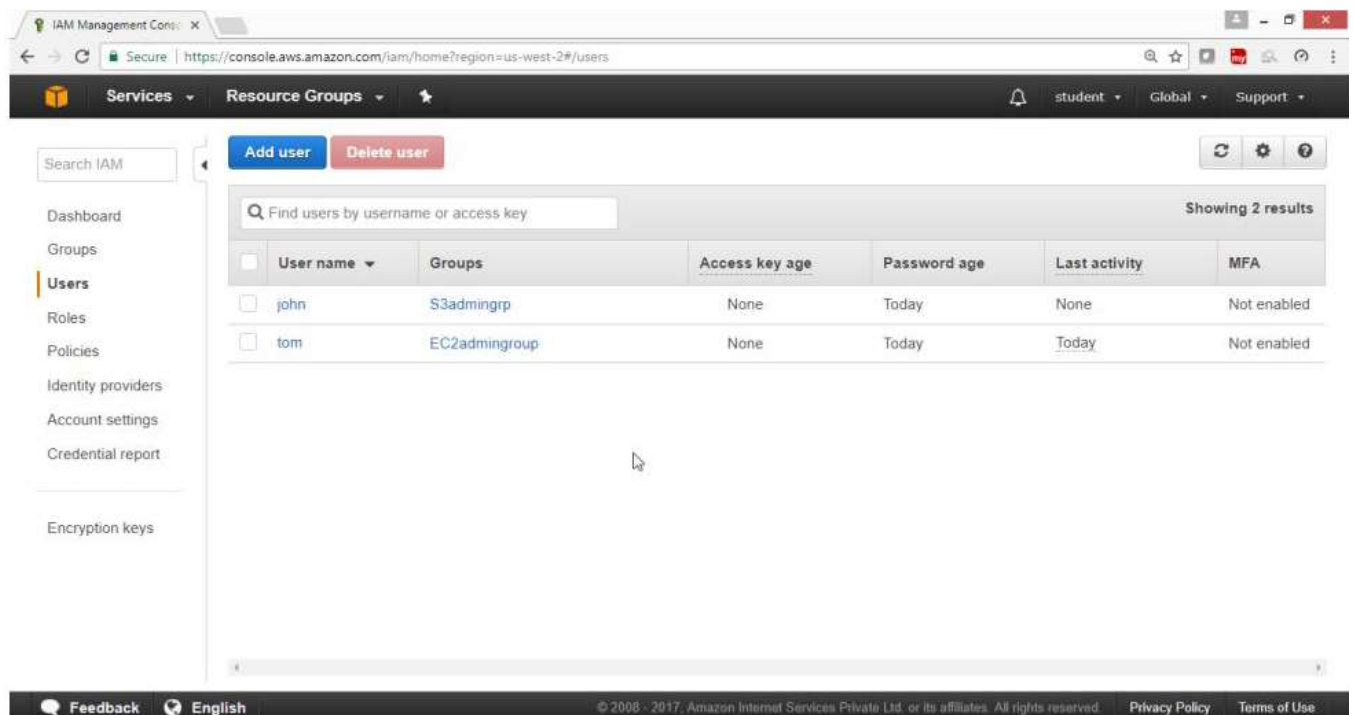
	User name	Groups	Access key age	Password age	Last activity	MFA
<input type="checkbox"/>	john	S3admingrp	None	Today	None	Not enabled
<input checked="" type="checkbox"/>	sai	None	None	Today	None	Not enabled
<input type="checkbox"/>	tom	EC2admingroup	None	Today	Today	Not enabled

Click on “Yes, Delete” Button



Verification

User sai is deleted



To Deleting Groups

From IAM Dashboard

Select the "Groups"

Drop Down "Group Action" Button

Select "Delete Group"

The screenshot shows the AWS IAM Management Console interface. The left sidebar contains navigation links: Search IAM, Dashboard, Groups (selected), Users, Roles, Policies, Identity providers, Account settings, Credential report, and Encryption keys. The main content area is titled 'Resource Groups' and shows a table of groups. A dropdown menu is open under the 'Group Actions' button, with 'Delete Group' highlighted. The table lists two groups: 'EC2admingroup' and 'S3admingrp'. The 'S3admingrp' group is selected with a checkbox and has a count of 1.

Search IAM

Dashboard

Groups

Users

Roles

Policies

Identity providers

Account settings

Credential report

Encryption keys

Create New Group

Group Actions

Filter

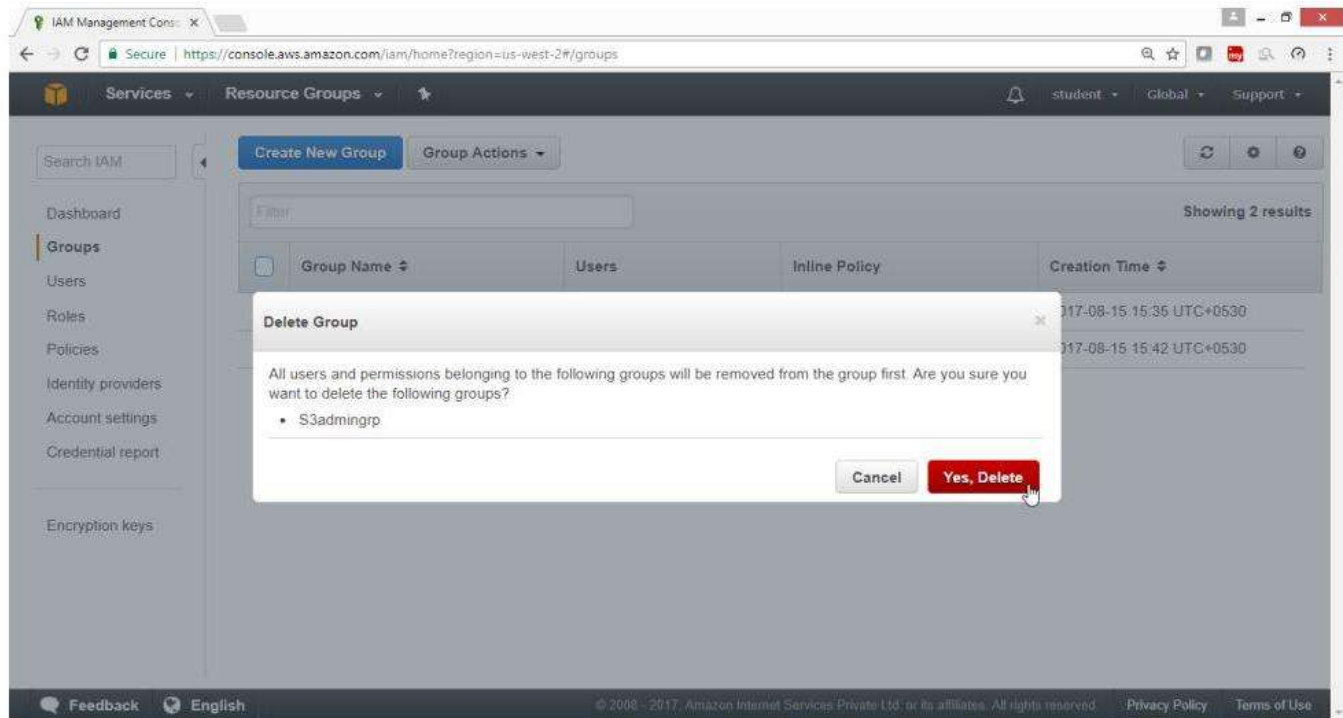
Showing 2 results

<input type="checkbox"/>	Group Name	Inline Policy	Creation Time
<input type="checkbox"/>	EC2admingroup		2017-08-15 15:35 UTC+0530
<input checked="" type="checkbox"/>	S3admingrp	1	2017-08-15 15:42 UTC+0530

Feedback English

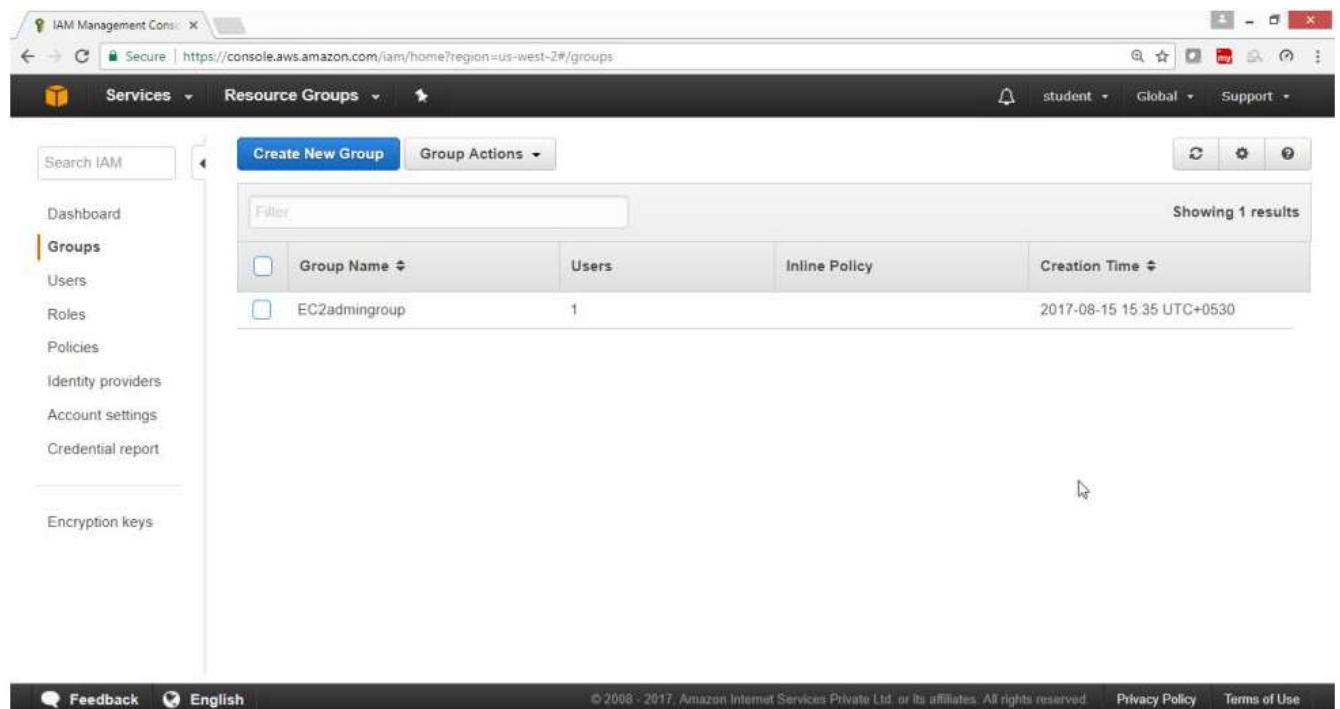
© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Click "Yes, Delete" Button



Verification

Group is deleted



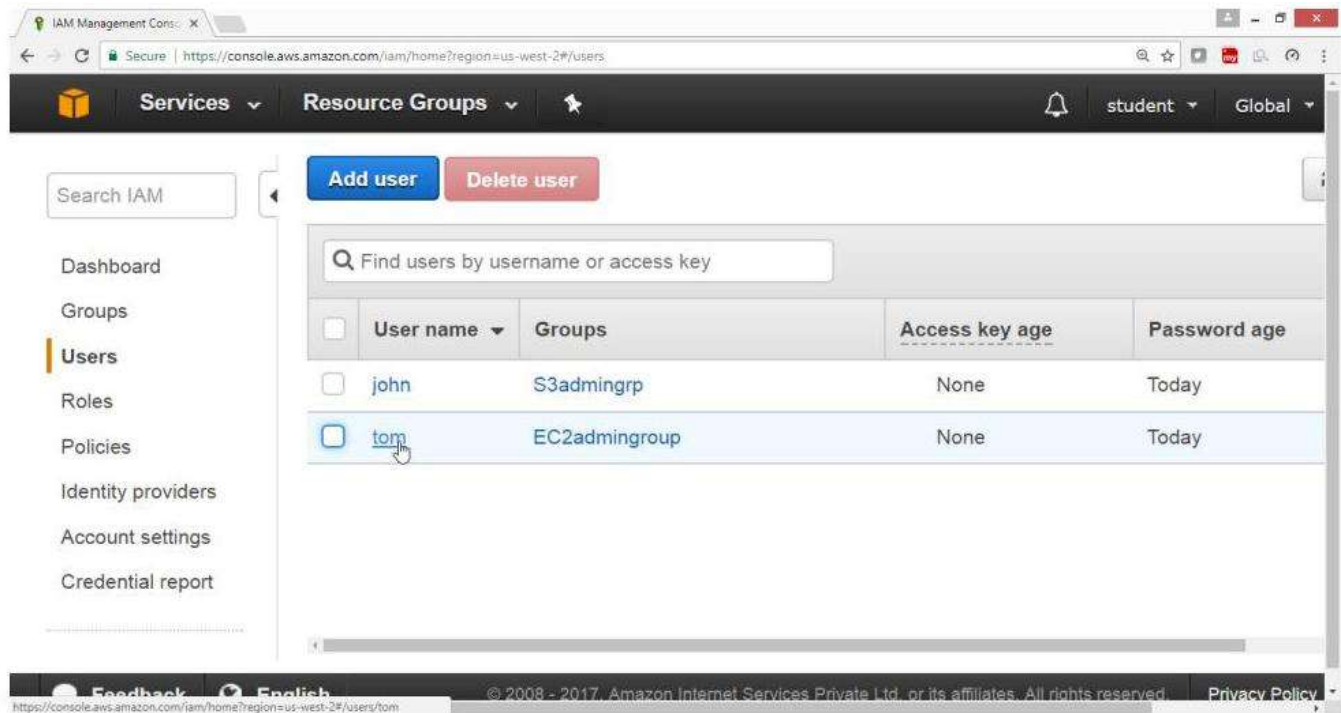
The screenshot shows the AWS IAM Management Console interface. The left sidebar contains navigation links: Dashboard, Groups (selected), Users, Roles, Policies, Identity providers, Account settings, Credential report, and Encryption keys. The main content area displays the 'Groups' page with a 'Create New Group' button and a 'Group Actions' dropdown. A table lists the groups, showing one result: 'EC2admin' with 1 user and a creation time of 2017-08-15 15:35 UTC+0530. The bottom footer includes 'Feedback', 'English', and copyright information.

Group Name	Users	Inline Policy	Creation Time
EC2admin	1		2017-08-15 15:35 UTC+0530

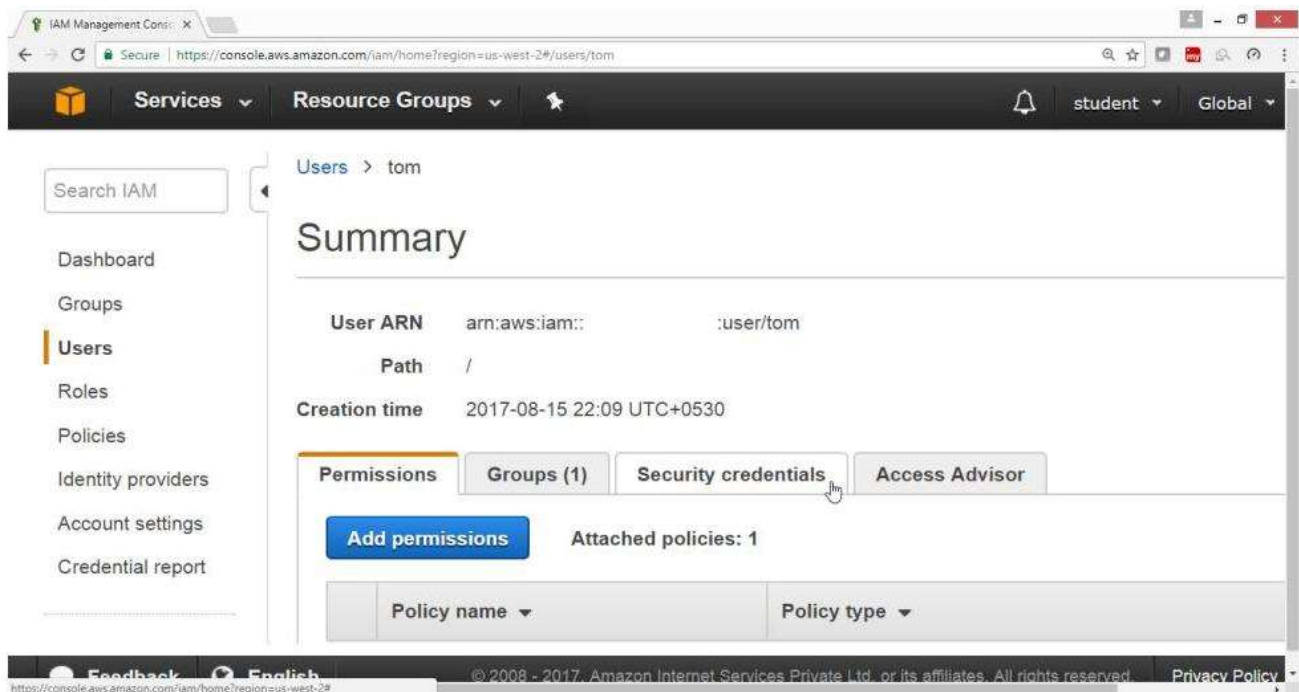
To create Multifactor Authentication

Install Google Authenticator in your Android Mobile

- On the "IAM Dashboard" Panel
- Click on Users
- Click on the user tom



Click on Security Credentials



Click on pen sign for "Assigned MFS Device"

Manage MFA Device

Select the type of MFA device to activate:

☒ A virtual MFA device

☐ A hardware MFA device

For more information about supported MFA devices, see [AWS Multi-Factor Authentication](#).

Cancel

Next Step

Select ->"A Virtual MFS Device"

Click on "Next Step" Button

Manage MFA Device

To activate a virtual MFA device, you must first install an AWS MFA-compatible application on the user's smartphone, PC, or other device. You can find a list of AWS MFA-compatible applications [here](#). After the application is installed, click Next Step to configure the virtual MFA.

☐ Don't show me this dialog box again.

Cancel

Previous

Next Step

Bar code will be created

Scan this bar code from your mobile Google Authenticator application


Now type 6 digital bar code in Authentication Code 1

Once the bar code changes

Retype 6-digit bar code in Authentication Code 2

Manage MFA Device

If your virtual MFA application supports scanning QR codes, scan the following image with your smartphone's camera.



Show secret key for manual configuration

After the application is configured, enter two consecutive authentication codes in the boxes below and click Activate Virtual MFA.

Authentication Code 1

232323

Authentication Code 2

455454

Cancel

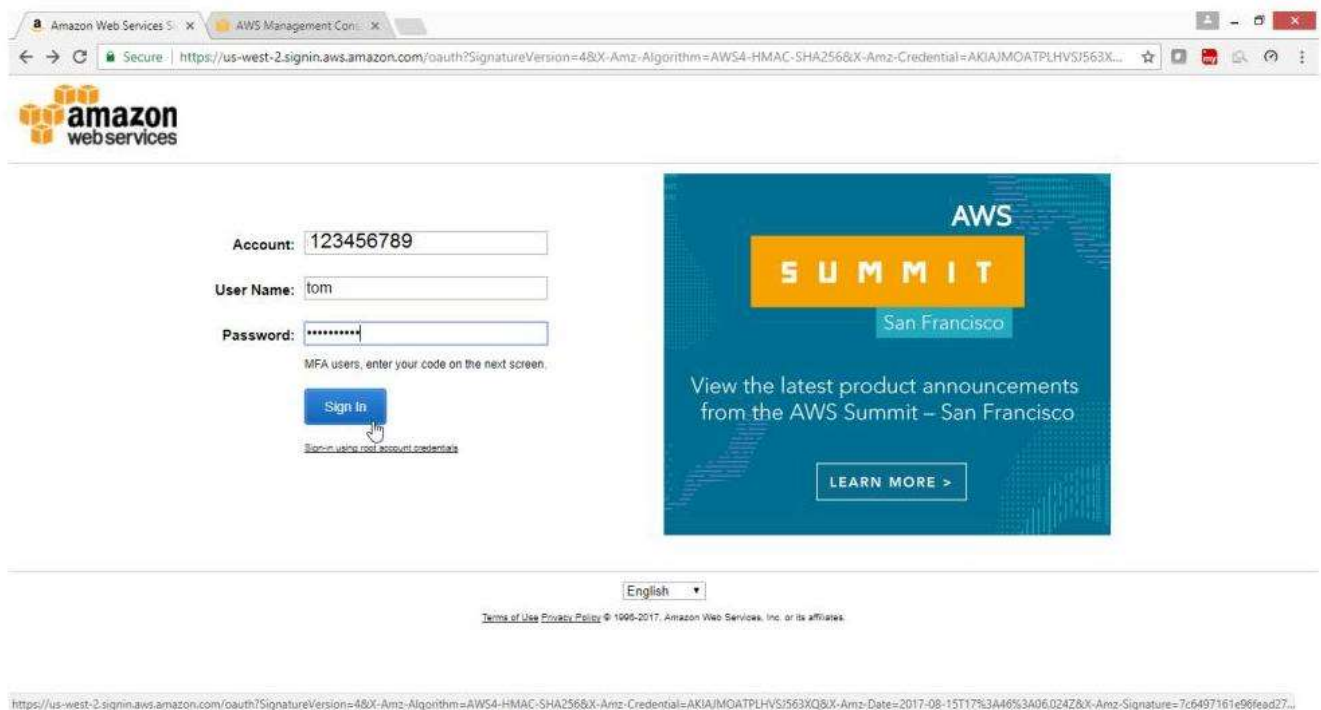
Previous

Activate Virtual MFA

Click on finish



Now login as tom user



Once the user types the MFA 6-digit coder

Click on Submit



Multi-factor Authentication

Please enter an MFA code to complete sign-in.

MFA Code:

English

[Terms of Use](#) [Privacy Policy](#) © 1995-2017, Amazon Web Services, Inc. or its affiliates.

https://us-west-2.signin.aws.amazon.com/oauth?SignatureVersion=4&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAJMOATPLHVSJ563XQ&X-Amz-Date=2017-08-15T17%3A46%3A06.024Z&X-Amz-Signature=7c6497161e96ead27...

Verify user had successfully logged in

The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with 'Services', 'Resource Groups', and a user profile 'tom @'. Below this, the 'Amazon Web Services' header is followed by a grid of service categories: Compute (EC2, EC2 Container Service, Lightsail, Elastic Beanstalk, Lambda, Batch), Developer Tools (CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, X-Ray), Internet of Things (AWS IoT, AWS Greengrass), Contact Center (Amazon Connect), Game Development (Amazon GameLift), Mobile Services (Mobile Hub), Storage (S3), and Management Tools (CloudWatch). On the right, the 'Resource Groups' section explains that a resource group is a collection of resources sharing tags, and includes a 'Create a Group' button. Below that, 'Additional Resources' links to 'Getting Started' documentation and the 'AWS Console Mobile App'.

What is IAM? What is IAM service?

- IAM stands for **Identity and Access Management**
- IAM is a web services that enable you to manage users and group permissions in AWS

- It is targeted at organizations with multiple users or systems that use AWS products such as Amazon Elastic Compute Cloud, Amazon Relational Database Service, and the AWS Management Console
- AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources for your users. You use IAM to control who can use your AWS resources (authentication) and what resources they can use and in what ways (authorization).

What does IAM gives you?

- Centralized control of your AWS account, Granular Permissions & Multifactor Authentication
- Identity Federation (Including Active Directory, Facebook, LinkedIn etc.,)
- Provide temporary access for users | devices and services where necessary
- Allow you to set up your own password rotation policy
- Integrates with many different AWS services
- Supports PCI DSS Compliance

What are the important components of IAM?

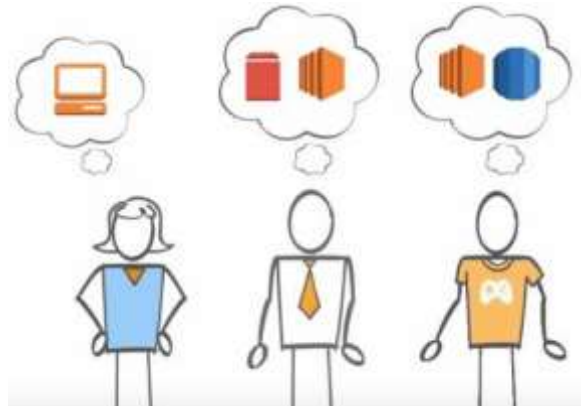
The important components of IAM are as follows:

- **IAM User:** An IAM User is a person or service that will interact with AWS. User can sign into AWS Management Console for performing tasks in AWS. (End Users)
- **IAM Group:** An IAM Group is a collection of IAM users under one set of permissions. We can specify permission to an IAM Group. This helps in managing large number of IAM users. We can simply add or remove an IAM User to an IAM Group to manage the permissions.
- **IAM Role:** An IAM Role is an identity to which we give permissions. A Role does not have any credentials (password or access keys). We can temporarily give an IAM Role to an IAM User to perform certain tasks in AWS.
- **IAM Permission:** In IAM we can create two types of Permissions. Identity based and Resource based. We can create a Permission to access or perform an action on an AWS Resource and assign it to a User, Role or Group. We can also create Permissions on resources like S3 bucket, Glacier vault etc and specify who has access to the resource.
- **IAM Policy:** An IAM Policy is a document in which we list permissions to specify Actions, Resources and Effects. This document is in JSON format. We can attach a Policy to an IAM User or Group.

Why we go for IAM?

- To avoid a security and logistical headache
- When you create an AWS account, it has permissions to do anything and everything with all the resources

- IAM Allows you to limit access as needed and gives you the peace of mind that approved people are accessing the right resources in the desired manner
- IAM will allow us to create multiple users with individual security credentials and permissions, with this IAM, each user is allowed to do only what they need to do



- Each user in the AWS account must have a unique set of credentials to access the console



How IAM works?

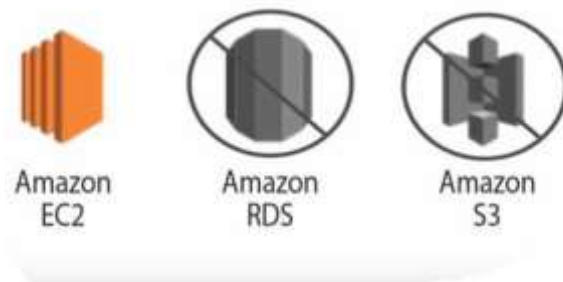
- Different types of users have different set of permissions



- Administrators need to access all AWS resource



- Developers need only access on Amazon Elastic Compute Cloud (EC2)



- We can use IAM to create a unique user for each employee and define their permissions

Adele
(Administrator)



Bob
(Systems Operations)



Dave
(Developer)



What is a Group?

- A group is a collection of IAM users
- After you set permissions on a group, those permissions are set to all users in the group
 - Even if we create user, we need to use groups to set permissions.
 - We need to manage access for number of groups instead of managing access for every individual user.



- We can able to,
 - Create a Group
 - Review the Group
 - Attach policy
 - Change the Group name
 - Delete a Group
 - Adding User to the Group

What is Multi-Factor Authentication?

AWS **Multi-Factor Authentication** (MFA) is a simple best practice that adds an extra layer of protection on top of user name and password. With MFA enabled, when a user signs into an AWS website, they will be prompted for their user name and password, as well as for an authentication code from their AWS MFA device. Taken together, these multiple factors provide increased security for your AWS.

- MFA provides additional security by requiring users to use a password and an authentication code from an external device.
- MFA is especially recommended for the AWS root accounts and accounts with administrator permissions since they have access to all your AWS resources.

Two types:

1. Hardware MFA device [Like your RSA token]
2. Virtual MFA device

Notes:

- It's not region specific.
- The Created Roles, Users, policies, groups etc are Universal, thus can be used across the regions.

How will you manage multiple users and their access rights with Amazon IAM?

AWS Identity and Access Management (IAM) is a web service in AWS cloud. It provides us APIs to create multiple Users and manage their permissions on AWS resources.

A user in AWS is an identity with unique security credentials that can be used to access AWS Services and Resources. With IAM we do not need to share passwords or access keys. IAM makes it easy to enable or disable a User's access as per the configuration.

We can implement best practices of security like least privilege, granting unique credentials to every User within AWS account etc. by using IAM.

Other Services

What is AWS Certificate Manager?



AWS Certificate Manager (ACM) handles the complexity of provisioning, deploying, and managing certificates provided by ACM (ACM Certificates) for your AWS-based websites and applications. You use ACM to request and manage the certificate and then use other AWS services to provision the ACM Certificate for your website or application. As shown by the following illustration, ACM Certificates are currently available for use with only Elastic Load Balancing and Amazon CloudFront. You cannot use ACM Certificates outside of AWS.

Like any other cloud computing environment, in AWS security is very important. We use AWS Certificate Manager (ACM) to handle the administration of security certificates (ACM Certificates) provided by AWS. ACM can be used to provision, deploy and manage the certificates in cloud environment. It can be used to provision as certificate on AWS based website. AWS certificates have a limitation that they can not be used outside AWS

What is the AWS Key Management Service?

The AWS Key Management Service (AWS KMS) is a managed service that makes it easy for you to create and control the encryption keys used to encrypt your data.

What is AWS WAF? What are the potential benefits of using WAF?

AWS WAF is a web application firewall that lets you monitor the HTTP and HTTPS requests that are forwarded to Amazon CloudFront and lets you control access to your content. Based on conditions that you specify, such as the IP addresses that requests originate from or the values of query strings, CloudFront responds to requests either with the requested content or with an HTTP 403 status code (Forbidden). You can also configure CloudFront to return a custom error page when a request is blocked.

Benefits of using WAF:

- Additional protection against web attacks using conditions that you specify. You can define conditions by using characteristics of web requests such as the IP address that the requests originate from, the values in headers, strings that appear in the requests, and the presence of malicious SQL code in the request, which is known as SQL injection.

- Rules that you can reuse for multiple web applications
- Real-time metrics and sampled web requests
- Automated administration using the AWS WAF API



Networking & Content Delivery

Amazon VPC Isolated Cloud Resources	Amazon CloudFront Global Content Delivery Network	Amazon Route 53 Scalable Domain Name System
Amazon API Gateway Build, Deploy, and Manage APIs	AWS Direct Connect Dedicated Network Connection to AWS	AWS Load Balancing High Scale Load Balancing

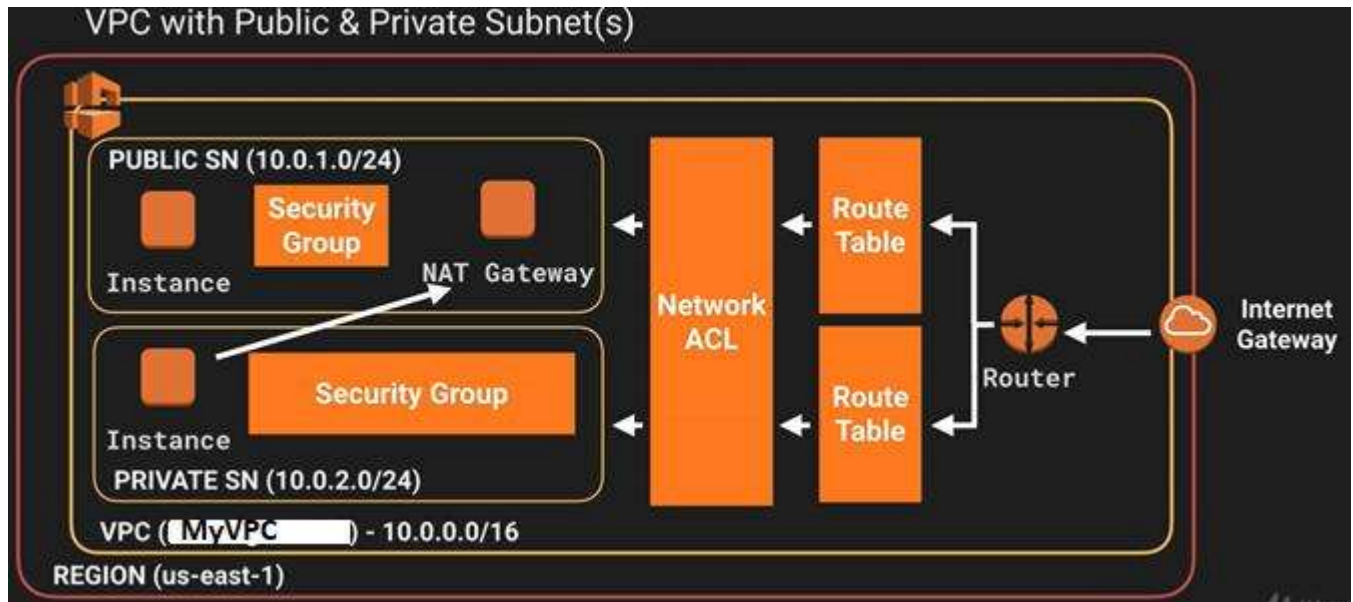


Networking & Content
Delivery

Amazon VPC

Share the VPC Configuration Step by Step

A virtual private cloud (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS cloud. We can launch your AWS resources, such as Amazon EC2 instances, into your VPC. We can configure your VPC; we can select its IP address range, create subnets, and configure route tables, network gateways, and security settings.



1. Create VPC

Log in to the AWS console.

Navigate to Services->VPC->Your VPCs.

Click —[Create VPC](#).

When you create a VPC, you specify a set of IP addresses in the form of a Classless Inter-Domain Routing (CIDR) block (for example, 10.0.0.0/16). For more information about CIDR notation and what "/16" means, see [Classless Inter-Domain Routing](#). (CIDR)

You can assign a single CIDR block to a VPC. The allowed block size is between a /28 netmask and /16 netmask. In other words, the VPC can contain from 16 to 65,536 IP addresses.

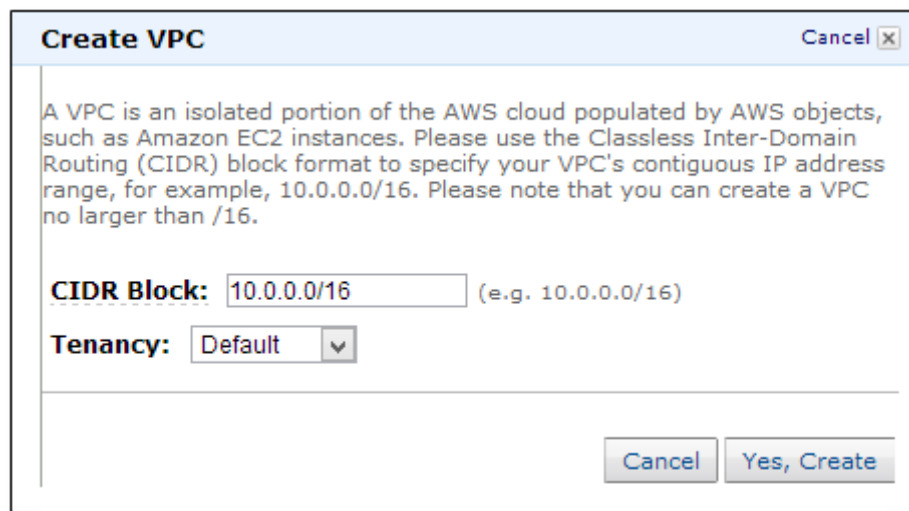
You cannot change a VPC's size after creating it. If your VPC is too small for your needs, you'll need to terminate all of the instances in the VPC, delete it, and then create a new, larger VPC.

To create your VPC, go to the Create VPC dialog box, specify the following VPC details and then click —
Yes, [Create](#)".

CIDR Block: Specify the CIDR block for your VPC. I prefer 10.0.0.0/16.

Tenancy: Default tenancy: This is for running instances on shared hardware and is free of charge.

Dedicated Tenancy: This is for running your instances on single-tenant hardware. A \$2 fee applies for each hour in which any dedicated instance is running in a region.



2. Create Subnets

In the navigation pane click on —[Subnets](#).

Click —[Create Subnet](#).

Before we create a subnet, let's understand the best practices for creating them.

You should create subnets across multiple availability zones, with each subnet residing within a single zone.

Creating subnets in and launching instances across multiple availability zones will ensure a high-availability environment.

When creating separate subnets for ELB, EC2 and RDS instances, each tier should have at least 2 subnets across availability zones.

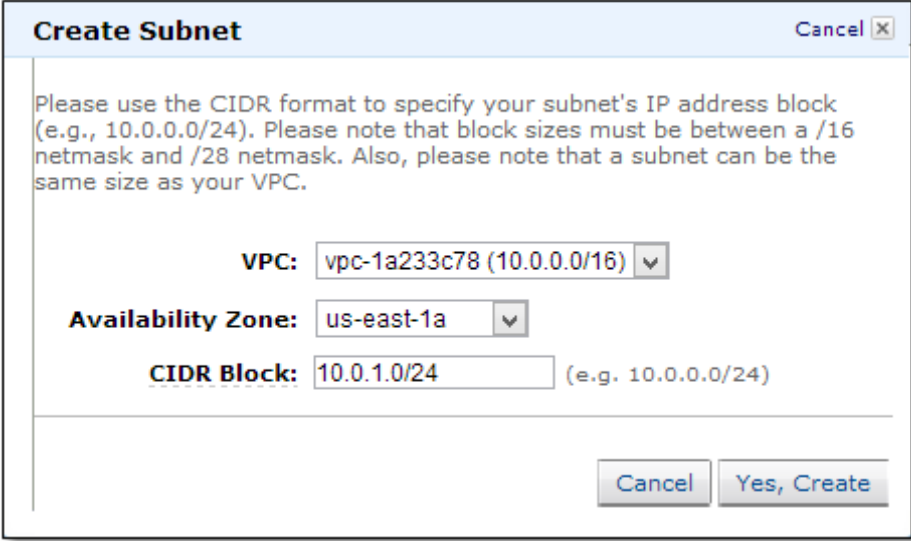
For this example, we created subnets using zones us-east1b and us-east-1d. These subnets are called —**private subnets** because the instances we launch are not accessible from the Internet. In other words, these instances don't have a public IP unless you assign an EIP.

App Tier: 10.0.1.0/24(zone-b), 10.0.2.0/24(zone-d)

ELB: 10.0.51.0/24(zone-b), 10.0.52.0/24(zone-d)

Database (RDS): 10.0.11.0/24(zone-b), 10.0.12.0/24(zone-d)

Always choose the same availability zones for all tiers. For example, if you choose two zones for high availability and use us-east-1a and us-east1b, then maintain those same 1a and 1b zones for all tiers. This will minimize data transfer charges because data transfers between instances within the same availability zone are free.



The screenshot shows the 'Create Subnet' dialog box. At the top, there's a title bar with 'Create Subnet' and a 'Cancel' button with a close icon. Below the title bar, there's instructional text: 'Please use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Please note that block sizes must be between a /16 netmask and /28 netmask. Also, please note that a subnet can be the same size as your VPC.' Below this text, there are three fields: 'VPC:' with a dropdown menu showing 'vpc-1a233c78 (10.0.0.0/16)', 'Availability Zone:' with a dropdown menu showing 'us-east-1a', and 'CIDR Block:' with a text input field containing '10.0.1.0/24' and a hint '(e.g. 10.0.0.0/24)'. At the bottom right, there are two buttons: 'Cancel' and 'Yes, Create'.

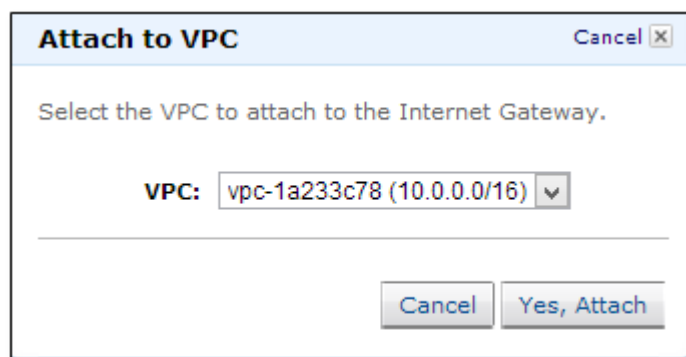
3. Create an Internet Gateway

By default, instances that are launched into a VPC can't communicate with the Internet. However, you can enable Internet access by attaching an Internet gateway to the VPC.

Go to Internet Gateways in the navigation pane and click —[Create Internet Gateway](#).



Now attach the gateway to a VPC by right clicking on —[VPC](#) and selecting —[Attach to VPC](#).



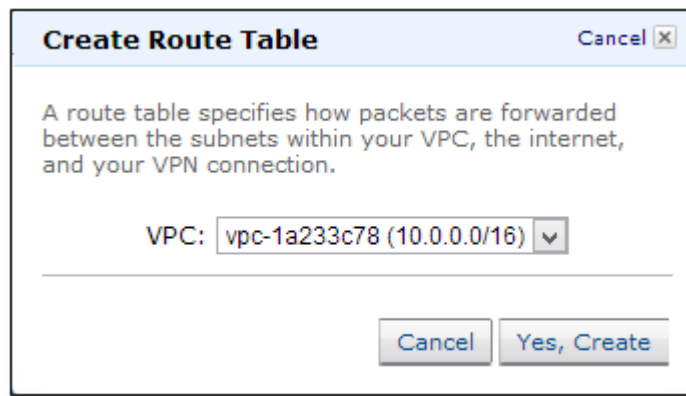
4. Create Route Tables

A route table contains a set of rules, called routes, that determine where network traffic is directed.

Each subnet in your VPC must be associated with a route table that will control that subnet's routing. You can associate multiple subnets with a single route table; however, you can only associate a subnet with one route table.

Creating a VPC automatically creates a main route table which, by default, enables the instances in your VPC to communicate with one other.

Go to Route Tables in the navigation pane and click on —[Create Route Table](#).



As a best practice create separate route tables for each tier. This will provide more control in maintaining the security of each subnet.

Now associate the subnets to the route tables.

Click on one route table and go to the Associations tab.

Select the subnet and click —Associate.



Associate each tier's subnets separately to the dedicated route table.

Create 3 new route tables:

1. **ELB Route table**—Associate 10.0.51.0/24 and 10.0.52.0/24.
2. **APP route table**—Associate 10.0.1.0/24 and 10.0.2.0/24.
3. **RDS route table**—Associate 10.0.11.0/24 and 10.0.12.0/24.

Do not associate any subnets with the main route table.

Now navigate to the main route table to add a route to allow Internet traffic to the VPC.

Go to Routes and specify the following values:

Destination: 0.0.0.0/0

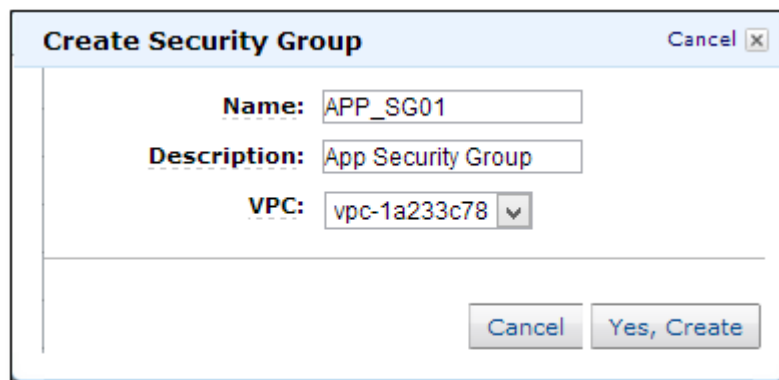
Target: Select —Internet Gateway— from the dropdown menu.



5. Create Security Groups

This process is similar to creating an SG (Security Group) in classic EC2.

Create separate security groups for ELB, APP, DB (RDS) and NAT instances.



1. APP_SG01
2. NAT_SG01
3. ELB_SG01
4. DB_SG01

Allow Inbound rules for ELB, DB and APP to suit your needs.

6. Create NAT instance

Instances launched into a private subnet in a VPC cannot communicate with the Internet unless you assign a public IP or EIP to the instance. However, assigning a public IP to an instance will allow everyone to initiate inbound Internet traffic.

Using a Network Address Translation (NAT) instance in your VPC enables instances in the private subnet to initiate outbound Internet traffic.

Create a subnet with netmask 10.0.0.0/24 for NAT instance. [Refer to section #2 of this post]. We call this subnet a —public subnet|| and the others —private subnets||. While, technically, there is no difference between public or private subnet, for clarity we call publicly accessible instances public subnets.

Associate this subnet to the main route table. You can also create separate route tables to associate to the subnet. If you do create a separate route table, don't forget to add a route that will allow Internet traffic into the subnet. [Refer to section #4 of this post].

Now navigate to [Services->EC2->Launch Instance](#)

In the Launch Wizard select —[Community AMIs](#) and search for —[ami-vpc-nat](#). — Select the first AMI from the results list to launch the instance into the VPC created in section #1. Choose the subnet 10.0.0.0/24 and then check the —Assign public IP|| box. You can also assign an EIP, if needed. On the Configure Security Group page, choose —Select an existing security group|| and select the NAT_SG security group that you created earlier.



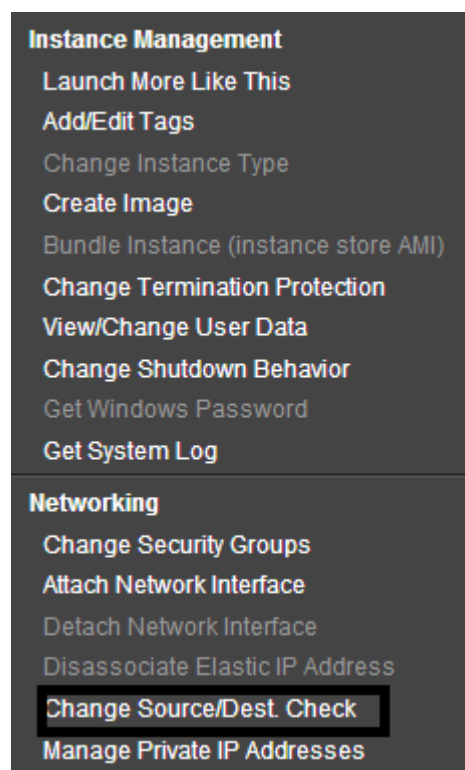
The screenshot shows the AWS EC2 Launch Wizard configuration page. It includes the following fields and options:

- Number of instances:** A text input field containing the value "1".
- Purchasing option:** A section with a checkbox labeled "Request Spot Instances" which is currently unchecked.
- Network:** A dropdown menu showing "vpc-1a233c78 (10.0.0.0/16)". To the right of the dropdown is a button with a plus icon and the text "Create new VPC".
- Subnet:** A dropdown menu showing "subnet-cc7d01e4(10.0.0.0/24) | us-east-1b". Below the dropdown, it says "251 IP Addresses available". To the right of the dropdown is a button with a plus icon and the text "Create new subnet".
- Public IP:** A section with a checkbox labeled "Automatically assign a public IP address to your instances" which is checked.

For this example, we created a micro server.

Choose a NAT instance type based on your intended workload. If your application only occasionally needs to connect to the Internet and doesn't require high network bandwidth, then a micro instance will suffice. If your application talks to the Internet continuously and requires better bandwidth, then start with m1.medium instances. You may need to upgrade the NAT instance to m1.large because network I/O varies between instance types.

Now, deselect the —Source/Destination check box, right click on the NAT instance, select —Change Source/Dest. Check, and click on —Disable.



The NAT instance must be able to send and receive traffic from sources or destinations other than itself, so you'll need to deselect the —source/destination check boxes.

Now navigate to Security Groups to add rules for inbound traffic.

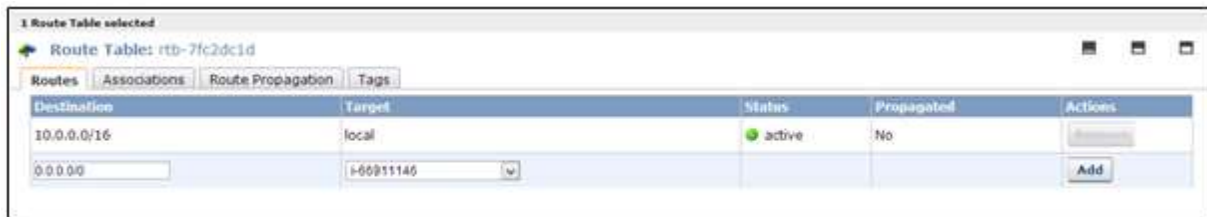
Go to the Inbound tab for NAT_SG01. These rules will allow app servers to talk to the NAT instance on the 80 and 443 ports.

1. Select —HTTP from the Create a new rule list. In the Source box, specify the IP address range of your private subnet (App server subnets) and then click —Add Rule.
2. Select —HTTPS from the Create a new rule list. In the Source box, specify the IP address range of your private subnet, and then click —Add Rule.

Click —[Apply Rule Changes](#).

Now navigate to Route Tables and select the private subnets 10.0.1.0/24 and 10.0.2.0/24.

On the Routes tab, specify 0.0.0.0/0 in the Destination box, specify the instance ID of the NAT instance in the Target box, and then click —[Add](#).



If you don't need an additional instance for NAT, you can minimize cost by assigning a public IP to the instance that needs Internet access. That will allow the instance to access the Internet directly.

7. Create App Servers

Now go to Services->EC2 ->[Launch Instance](#).

On the Configure Instance Details page, from the Network list choose the VPC that you created previously and select your app server subnet (10.0.1.0/24, 10.0.2.0/24) from the Subnet list.

Optional: Select the —[Public IP](#) check box to request that your app instance receive a public IP address. This is required when you don't have a NAT instance, but your instance requires Internet access.

On the Configure Security Group page, select the option —[Select an existing security group](#) and then select the APP_SG01 security group that you created previously. Click —[Review and Launch](#).

Now log in to the server and check to see whether or not you can access the Internet.

```
$ ping google.com
```

You now might ask, —[How can I access from my desktop an instance that was created in a private subnet and has no assigned public IP?](#) The answer is that you can't. To do so, you'll need a bastion box in the public subnet. You can use a NAT instance as a bastion server (also known as a jump box).

Log in to the bastion (NAT) server first. You can access any instance from this server that was created in a private subnet.

8. Create RDS

Navigate to Services->RDS

Go to Subnet Groups in the navigation pane and click —Create DB Subnet Group||.

Select the VPC ID from the drop-down menu.

Select —Availability Zone and choose the Subnet IDs of 10.0.11.0/24 and 10.0.12.0/24. Then click —Add||

Click —Yes, Create to create the subnet group.

Create DB Subnet Group

To create a new Subnet Group give it a name, description, and select an existing VPC below. Once you select an existing VPC, you will be able to add subnets related to that VPC.

Name:

Description:

VPC ID:

Add Subnet(s) to this Subnet Group. You may add subnets one at a time below or add all the subnets related to this VPC. You may make additions/edits after this group is created.

Availability Zone:	Subnet ID:	Availability Zone	Subnet ID	CIDR Block	Action
<input type="text" value="us-east-1d"/>	<input type="text" value="subnet-2c90cd6a"/>	us-east-1d	subnet-2c90cd6a	10.0.12.0/24	Remove
<input type="button" value="Add"/>		us-east-1b	subnet-057d012d	10.0.11.0/24	Remove

Creating an Options Group and a Parameters Group is similar to doing so in classic EC2.

Launch an RDS instance within the subnet group created above.

In the Additional Config window, select the VPC and DB Subnet Groups created previously.

Additional Config

Provide the optional additional configuration details below.

Database Name: (e.g. mydb)

Note: if no database name is specified then no initial MySQL database will be created on the DB Instance.

Database Port:

Choose a VPC:

DB Subnet Group:

Publicly Accessible: ☐ Yes ☒ No

To make sure that your RDS instance is launched in subnets 10.0.11.0/24 and 10.0.12.0/24, [select the —mydbsubgroup01](#) subnet group. All other steps for creating an RDS are as usual.

9. Create ELB

Now it's time to create the load balancer. The load balancer will be the frontend and will be accessible from the Internet, which means that the ELB will be launched in public subnets 10.0.51.0/24 and 10.0.52.0/24.

At this point the two subnets can't access the Internet. To make them public subnets, update the route table that these subnets are associated to.

Navigate to Services->VPC->Route Tables

Select the ELB route table.

On the Routes tab, specify 0.0.0.0/0 in the Destination box, select the Internet gateway in the Target box, and then click [—Add](#).

Navigate to Services-> EC2-> [Load Balancers](#)

Click [—Create Load Balancer](#).

In the Launch Wizard, select [—Create LB inside](#) as your VPC ID.

Do not select [—Create an internal load balancer](#).

Click [—Continue](#)

In Add EC2 Instances select the subnets where you want the load balanced instances to be. Select 10.0.51.0/24 and 10.0.52.0/24.

Create a New Load Balancer

Cancel

DEFINE LOAD BALANCER

CONFIGURE HEALTH CHECK

ADD EC2 INSTANCES

REVIEW

You will need to select a Subnet for each Availability Zone where you wish to have load balanced instances. A Virtual Network Interface will be placed inside the Subnet and allow traffic to be routed into that Availability Zone. Only one subnet per Availability Zone may be selected.

VPC: vpc-1a233c78

Available Subnets

	Subnet ID	Subnet CIDR	Availability Zones
	subnet-ca037fe2	10.0.1.0/24	us-east-1b
	subnet-2c90cd6a	10.0.12.0/24	us-east-1d
	subnet-cc7d01e4	10.0.0.0/24	us-east-1b
	subnet-057d012d	10.0.11.0/24	us-east-1b

Selected Subnets*

	Subnet ID	Subnet CIDR	Availability Zones
	subnet-067d012e	10.0.51.0/24	us-east-1b
	subnet-2093ce66	10.0.52.0/24	us-east-1d

Back

Continue

* Required field

In the next window select **Choose from your existing security group** and then select the ELB_SG01 security group that you created previously. Click —[Continue](#).

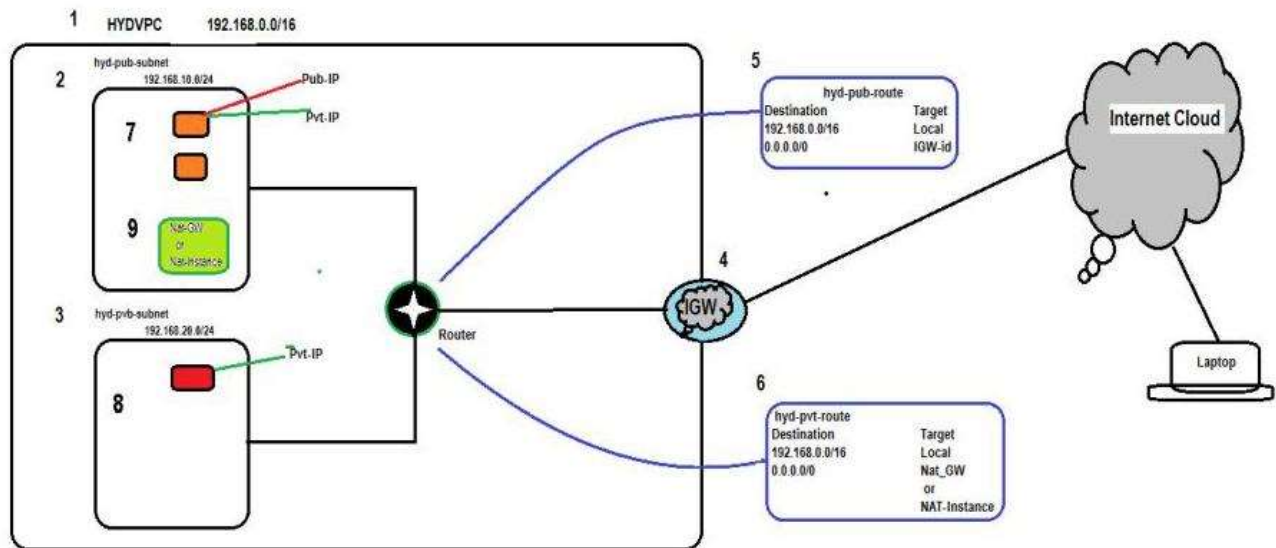
In the next window select the App servers. Click —[Continue](#).

Review the details and click —[Create](#).

Make sure that you've enabled the APP_SG01 inbound ports (80/443) to ELB_SG01 so that the ELB can route traffic to backend app servers. Also make sure that ELB_SG01 HTTP and HTTPS ports are publicly accessible (0.0.0.0/0).

Share the VPC Configuration with Public Subnet and Private Subnet Step by Step
[To configure Amazon Virtual Private Cloud with public and private subnet](#)

[Topology](#)



Pre-requisites

User should have AWS account, or IAM user with VPCFullAccess Policy

Task

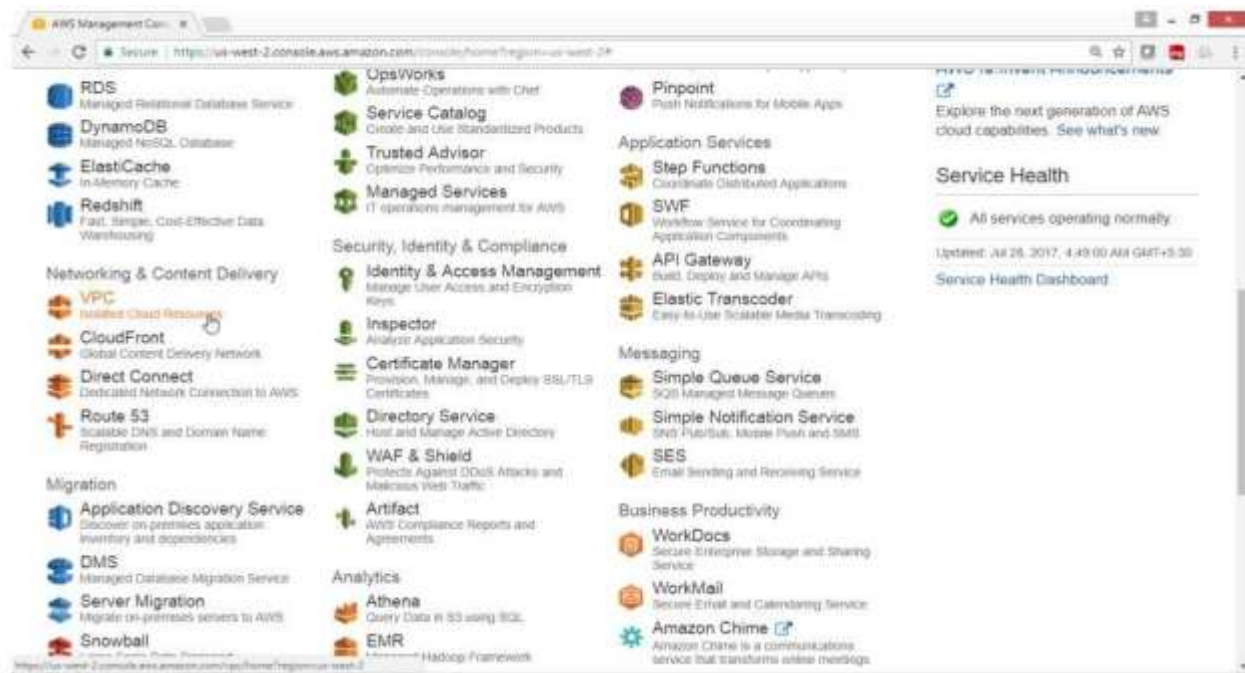
1. Create your own VPC
2. Create Public Subnet
3. Create Private Subnet
4. Create Internet Gateway
5. Attach Internet Gateway to your VPC
6. Create Public Routing Table, associate subnet and add routing rules
7. Create Private Routing Table, associate subnet and add routing rules
8. Launch an instance in Public network
9. Launch an instance in Private network
10. Create NAT Gateway
11. Connect to public instance and check internet connectivity
12. Connect to private instance and check internet connectivity

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

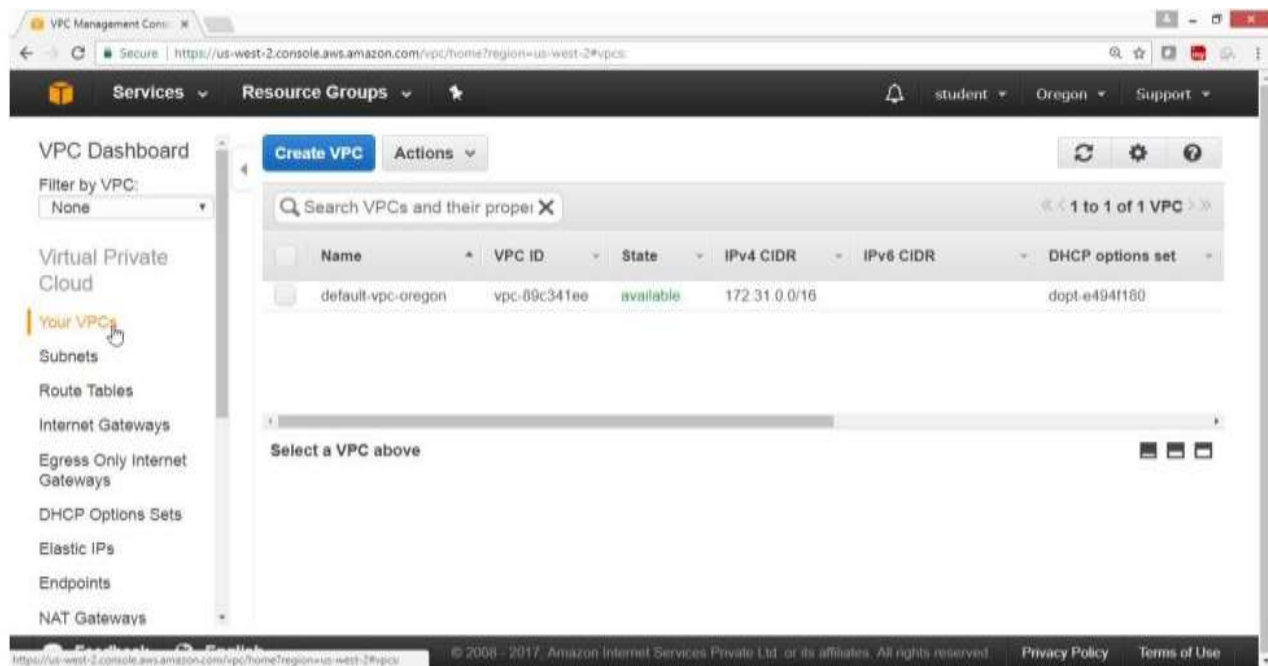
1) To create your own VPC

Open AWS console

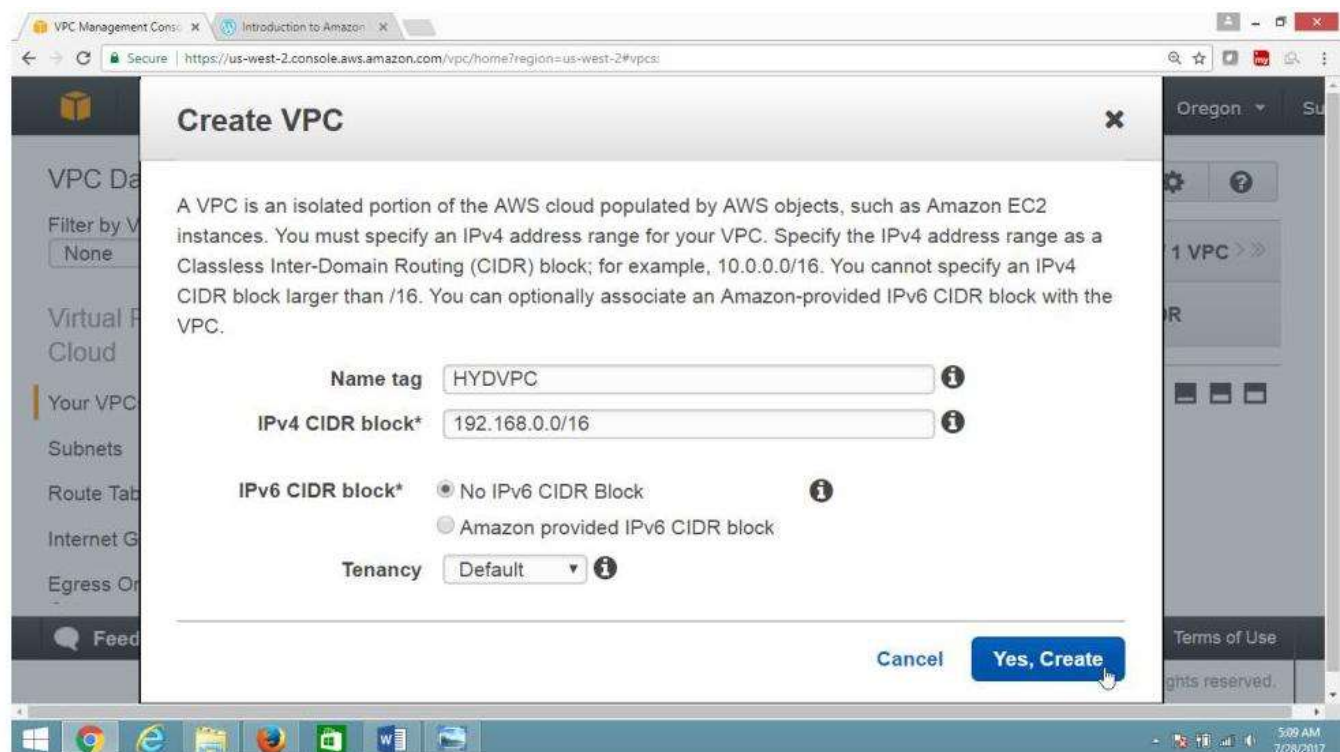
- Click on "Services"
- Select "Networking and Content Delivery"
- Click on VPC



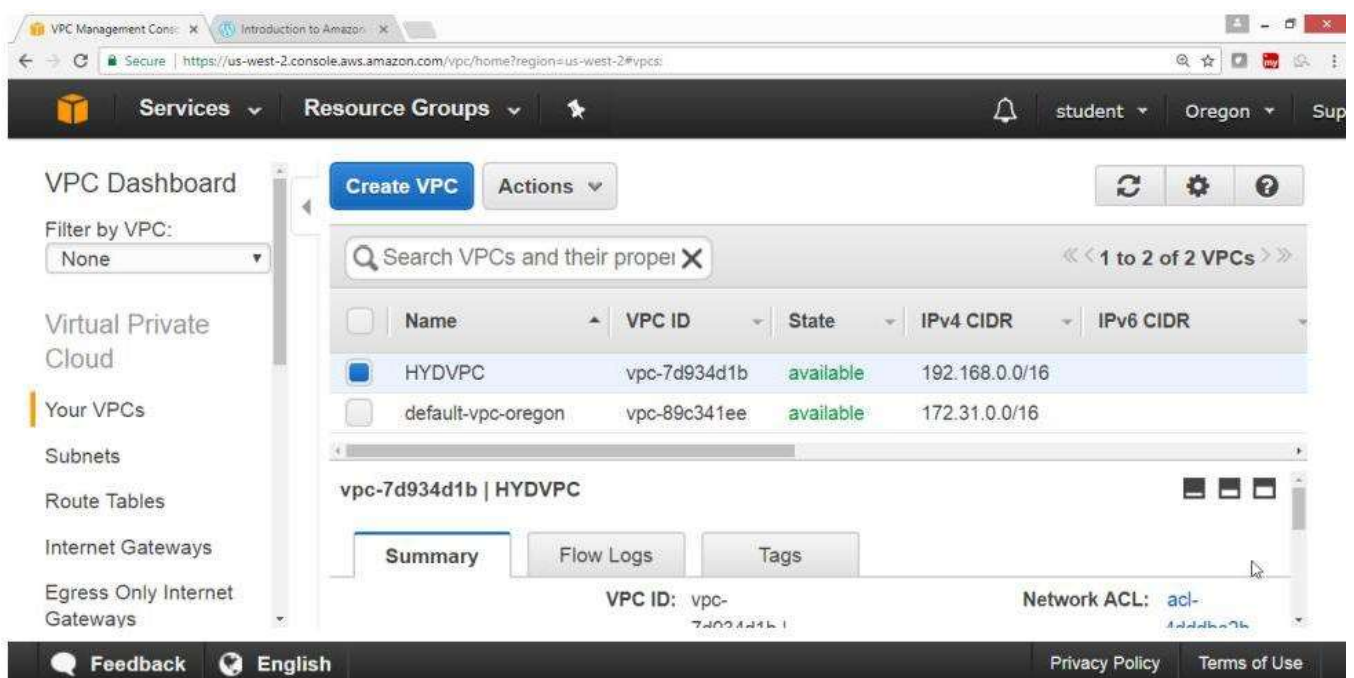
On "VPC Dashboard" Panel
 Click on "Your VPC"
 Click on "Create VPC" Button



On "Create VPC", page
 For Name tag->HYDVPC
 For IPV4 CIDR Block -> 192.168.0.0/16
 Leave remaining field as default
 Click on "Yes Create" Button



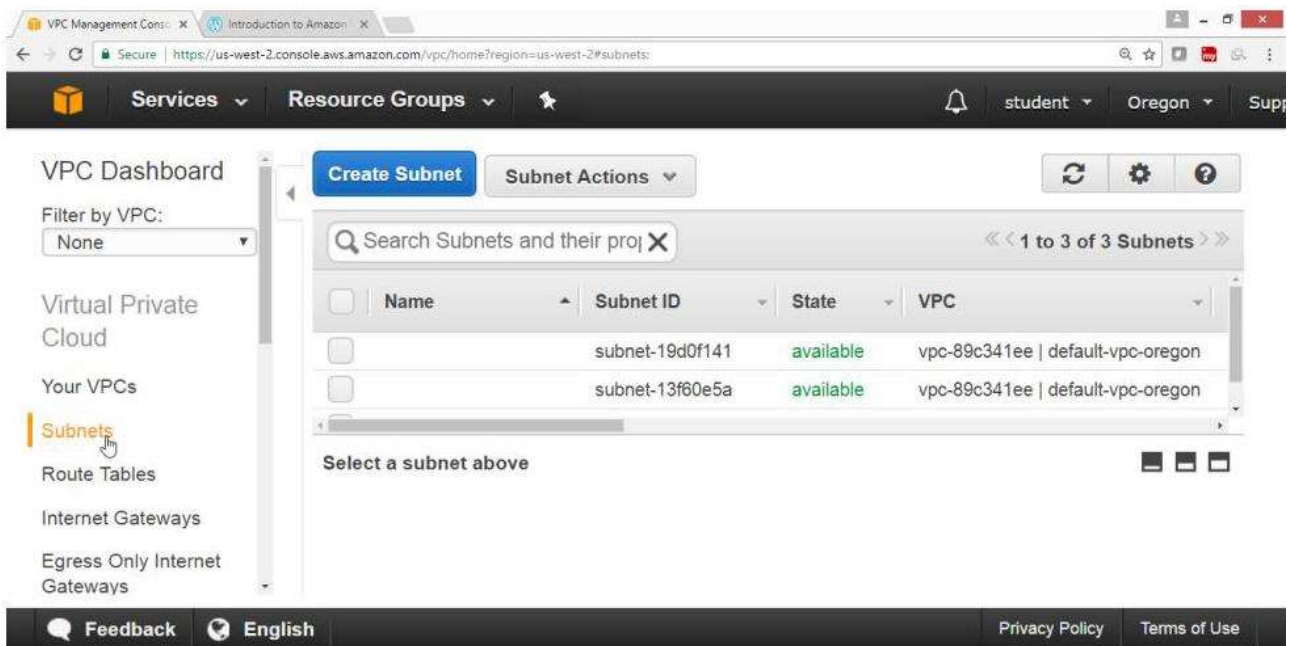
Verify HYDVPC is created



2) To create public subnet

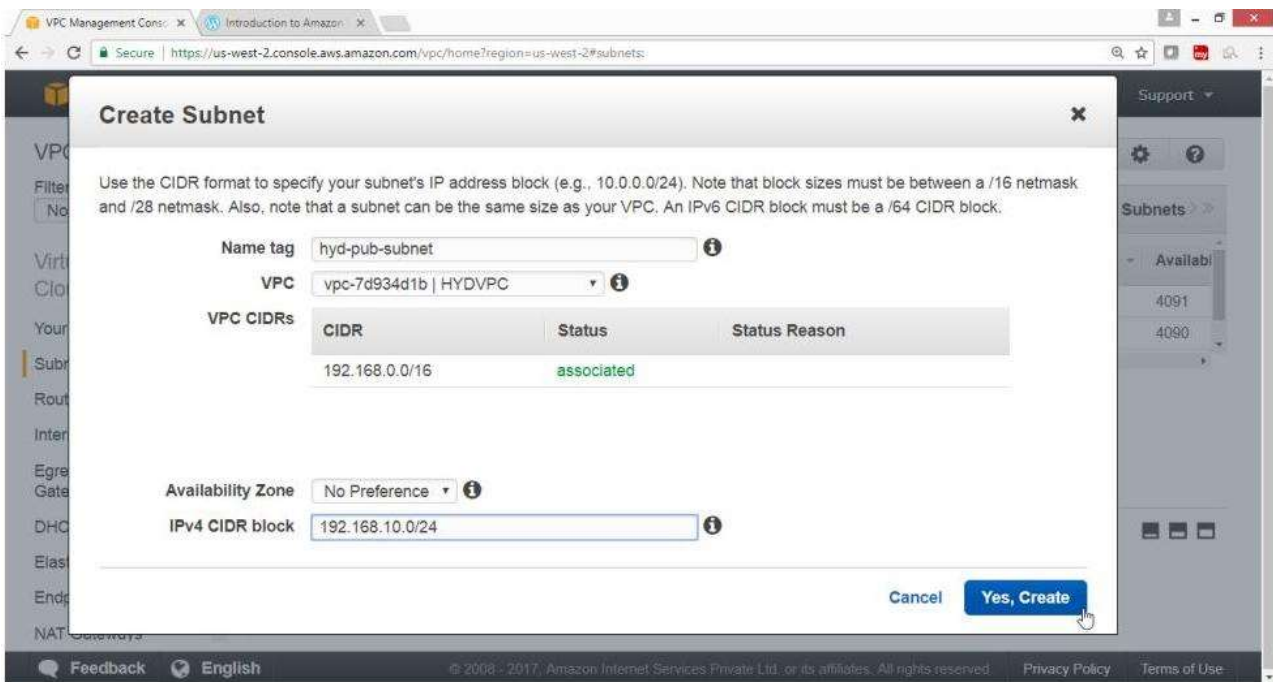
Click on Subnet

Click on Create Subnet button

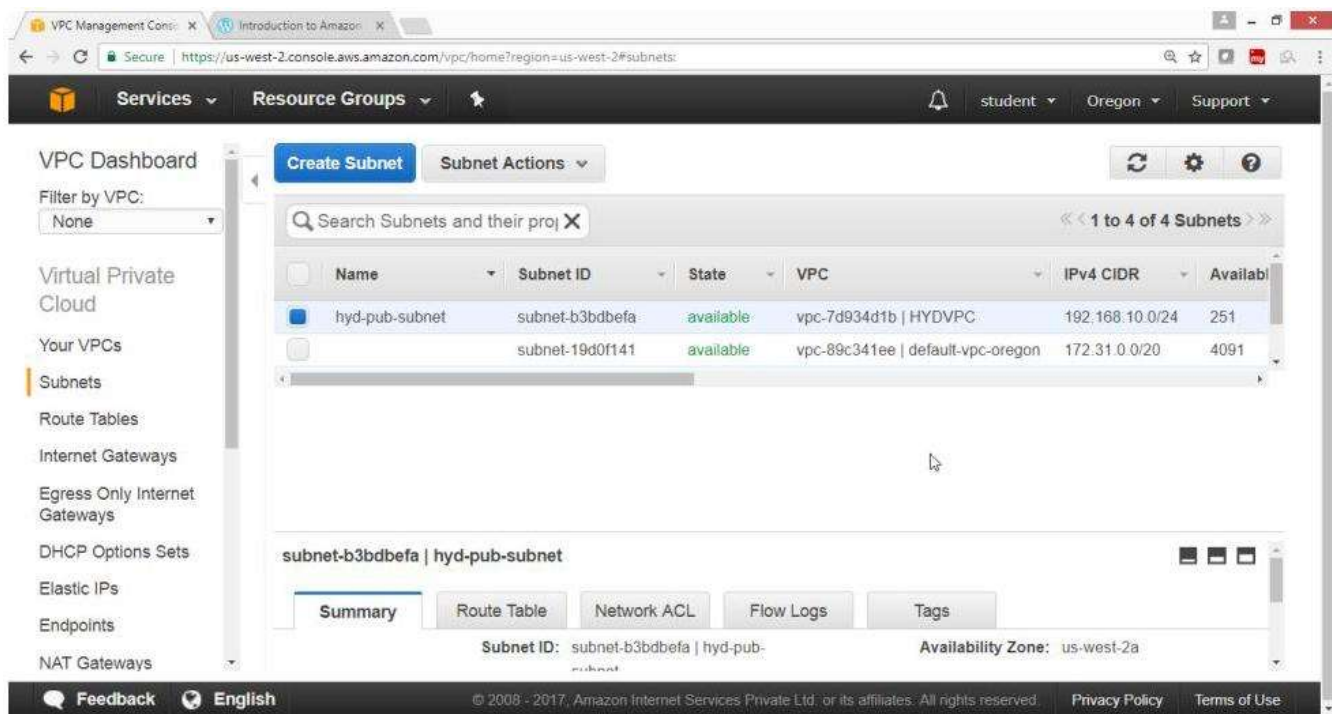


On Create Subnet, page

- For Name tag -> hyd-pub-subnet
- For VPC -> HYDVPC
- For IPV4 CIDR Block -> 192.168.10.0/24
- Click on "Yes Create" Button



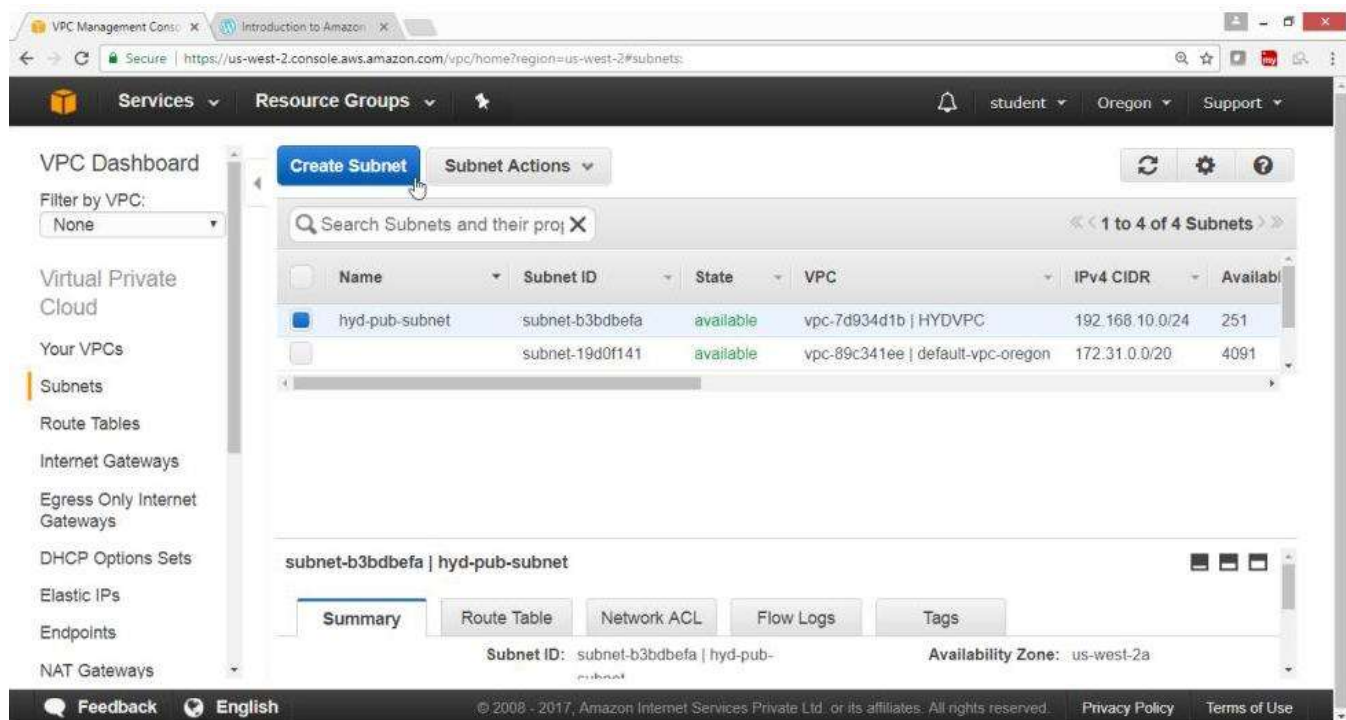
Verify hyd-pub-subnet got created



3) To create private subnet

Click on "Subnet"

Click on "Create Subnet" Button



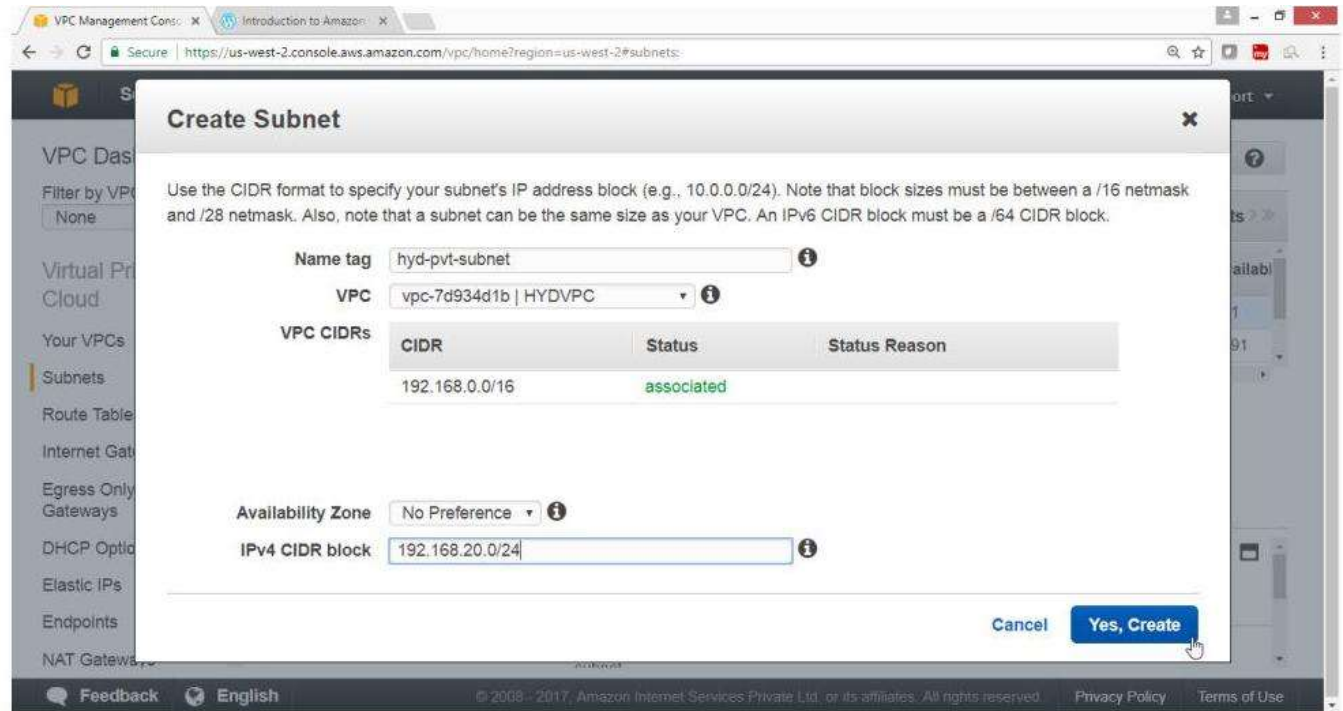
On Create Subnet, page

For Name tag -> hyd-pvt-subnet

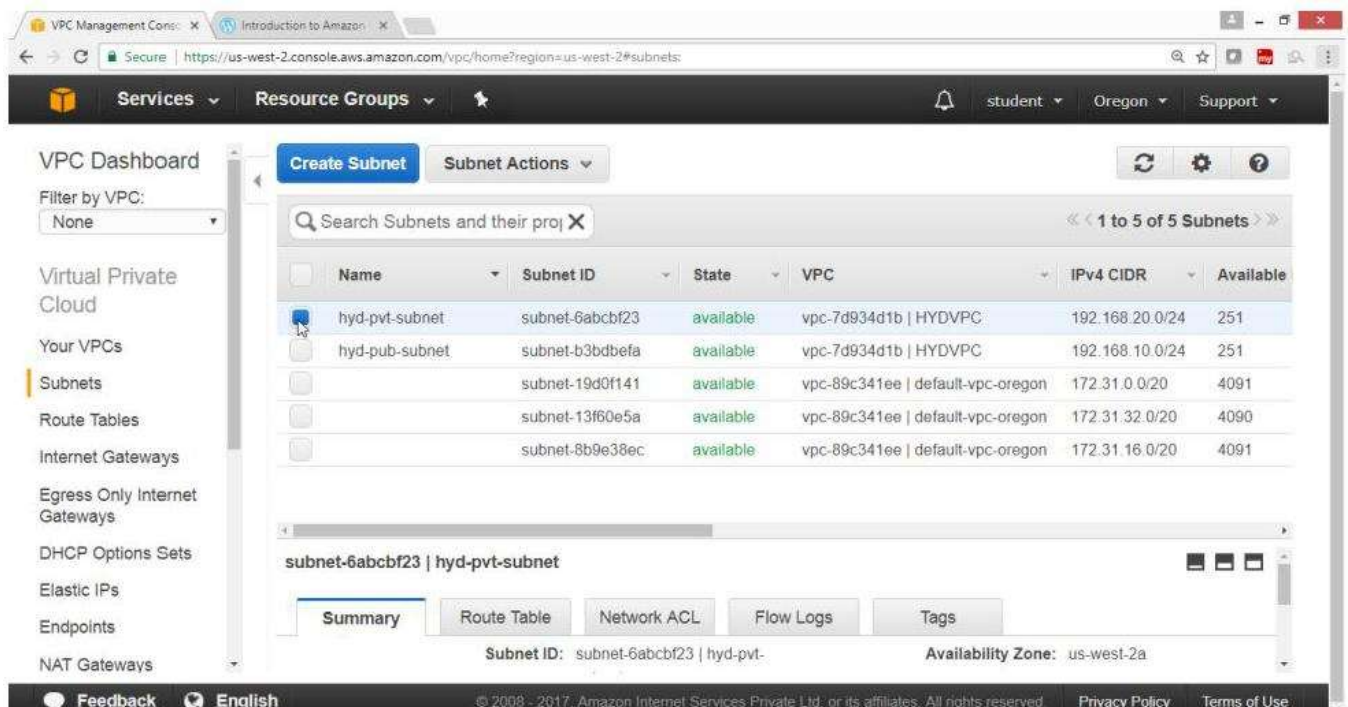
For VPC ->HYDVPC

For IPV4 CIDR Block -> 192.168.20.0/24

Click on "Yes Create" Button



Verify hyd-pvt-subnet got created



4) Create a Internet Gateway and attach to your VPC

In VPC "Dashboard" panel

Click on "Internet Gateway"

VPC Management Console - Introduction to Amazon

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#subnets;

Services Resource Groups student Oregon Support

VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Create Subnet Subnet Actions

Search Subnets and their projects

<< 1 to 5 of 5 Subnets >>

Name	Subnet ID	State	VPC	IPv4 CIDR	Available
hyd-pvt-subnet	subnet-6abcbf23	available	vpc-7d934d1b HYDVPC	192.168.20.0/24	251
hyd-pub-subnet	subnet-b3bdbefa	available	vpc-7d934d1b HYDVPC	192.168.10.0/24	251
	subnet-19d0f141	available	vpc-89c341ee default-vpc-oregon	172.31.0.0/20	4091
	subnet-13f60e5a	available	vpc-89c341ee default-vpc-oregon	172.31.32.0/20	4090
	subnet-8b9e38ec	available	vpc-89c341ee default-vpc-oregon	172.31.16.0/20	4091

subnet-6abcbf23 | hyd-pvt-subnet

Summary Route Table Network ACL Flow Logs Tags

Subnet ID: subnet-6abcbf23 | hyd-pvt- Availability Zone: us-west-2a

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Click on "Create Internet Gateway" button

VPC Management Console - Introduction to Amazon

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#igw;

Services Resource Groups student Oregon Support

VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Create Internet Gateway Delete Attach to VPC Detach from VPC

Search Internet Gateways and projects

<< 1 to 1 of 1 Internet Gateway >>

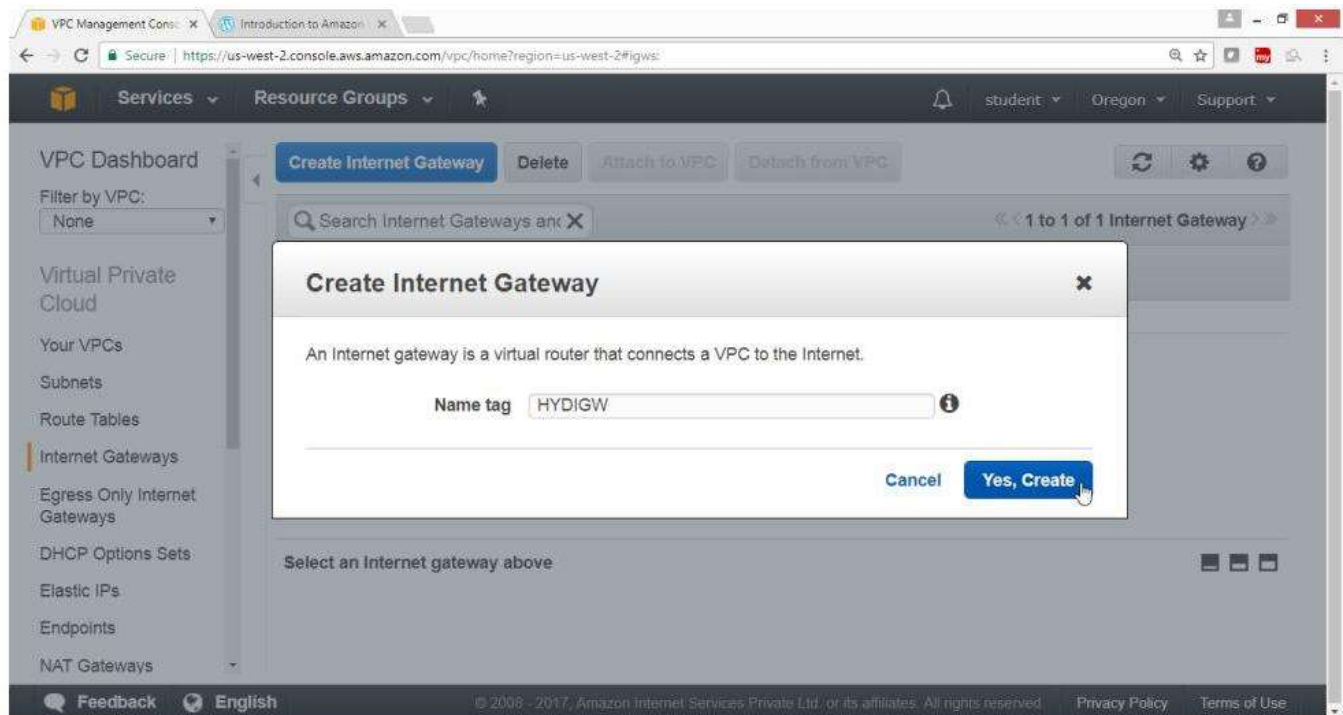
Name	ID	State	VPC
	igw-6ea7f110a	attached	vpc-89c341ee default-vpc-oregon

Select an Internet gateway above

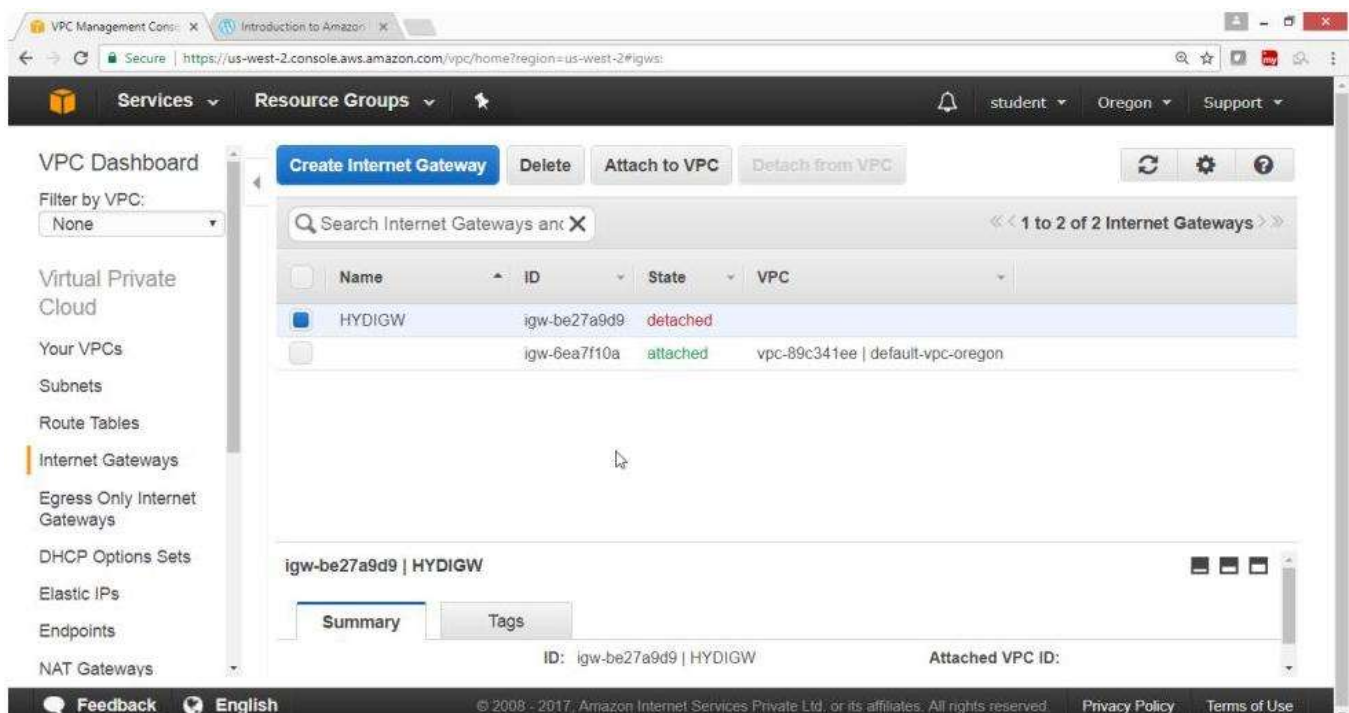
Feedback English

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

In "Create Internet Gateway", box
For Name tag-> HYDIGW
Click on "Yes, Create" button

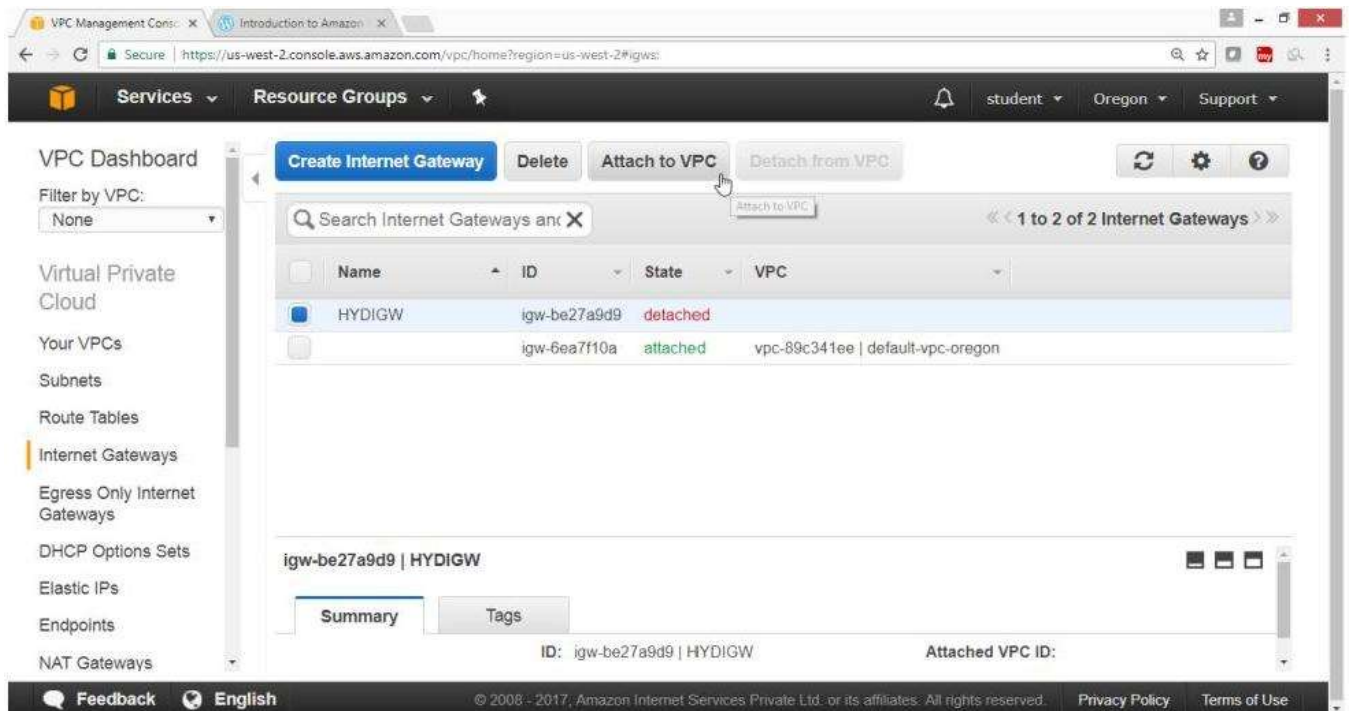


Verify, Internet gateway is created



Select HYDIGW

Click "Attach to VPC"



The screenshot shows the AWS VPC Management Console. On the left sidebar, 'Internet Gateways' is selected. The main panel shows a list of Internet Gateways. The first gateway, 'HYDIGW' (ID: igw-be27a9d9), is in a 'detached' state. A tooltip 'Attach to VPC' is visible over the 'Attach to VPC' button. Below the list, the details for 'igw-be27a9d9 | HYDIGW' are shown, with tabs for 'Summary' and 'Tags'. The 'Summary' tab is active, showing the ID and the 'Attached VPC ID' field.

In "Attach to VPC" box

For VPC->HYDVPC

Click on "Yes, Attach" button



The screenshot shows the 'Attach to VPC' dialog box. It has a title bar with 'Attach to VPC' and a close button. Below the title bar, there is a description: 'Attach an Internet gateway to a VPC to enable communication with the Internet.' Under the 'VPC' label, a dropdown menu shows 'vpc-7d934d1b | HYDVPC' with an information icon. At the bottom right, there are two buttons: 'Cancel' and 'Yes, Attach'. A mouse cursor is pointing at the 'Yes, Attach' button.

Verify the Internet Gateway is connected to your VPC

The screenshot shows the AWS VPC console interface. On the left sidebar, the 'Internet Gateways' link is highlighted. The main panel displays a table of Internet Gateways. The first gateway, 'HYDIGW' (ID: igw-be27a9d9), is in an 'attached' state and is connected to the VPC 'HYDVPC' (ID: vpc-7d934d1b). The second gateway, 'igw-6ea7f10a', is also 'attached' but connected to the 'default-vpc-oregon'.

Name	ID	State	VPC
HYDIGW	igw-be27a9d9	attached	vpc-7d934d1b HYDVPC
	igw-6ea7f10a	attached	vpc-89c341ee default-vpc-oregon

Below the table, the details for 'igw-be27a9d9 | HYDIGW' are shown. The 'Summary' tab is active, displaying the ID 'igw-be27a9d9 | HYDIGW' and the 'Attached VPC ID: vpc-7d934d1b | HYDVPC'.

5) Create Public Routing Table, associate subnet and add routing rules

On VPC Dashboard panel

Click on "Route Table"

This screenshot shows the same AWS VPC console interface, but the 'Route Tables' link in the left sidebar is now highlighted with a mouse cursor. The main panel still shows the Internet Gateway table, but the details section at the bottom is partially obscured by the sidebar's position.

Click on "Create Route Table" button

The screenshot shows the AWS VPC Management Console interface. The left sidebar contains a navigation menu with options like 'Virtual Private Cloud', 'Your VPCs', 'Subnets', 'Route Tables', 'Internet Gateways', 'Egress Only Internet Gateways', 'DHCP Options Sets', 'Elastic IPs', 'Endpoints', and 'NAT Gateways'. The 'Route Tables' option is selected. The main content area displays a table of existing route tables. Above the table, there are buttons for 'Create Route Table', 'Delete Route Table', and 'Set As Main Table'. The 'Create Route Table' button is highlighted with a mouse cursor. Below the table, there is a message 'Select a route table above' and three small icons.

Name	Route Table ID	Explicitly Associat	Main	VPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
	rtb-847d52e2	0 Subnets	Yes	vpc-7d934d1b HYDVPC

On "Create Route Table" box

For Name Tag-> hyd-pub-route

For VPC->HYDVPC

Click on "Yes, Create" Button

The screenshot shows the 'Create Route Table' dialog box. It has a title bar with 'Create Route Table' and a close button. Below the title bar, there is a descriptive text: 'A route table specifies how packets are forwarded between the subnets within your VPC, the Internet, and your VPN connection.' The dialog contains two input fields: 'Name tag' with the value 'hyd-pub-route' and 'VPC' with the value 'vpc-7d934d1b | HYDVPC'. Both fields have an information icon (i) to their right. At the bottom right, there are two buttons: 'Cancel' and 'Yes, Create'. A mouse cursor is pointing at the 'Yes, Create' button.

Create Route Table ✕

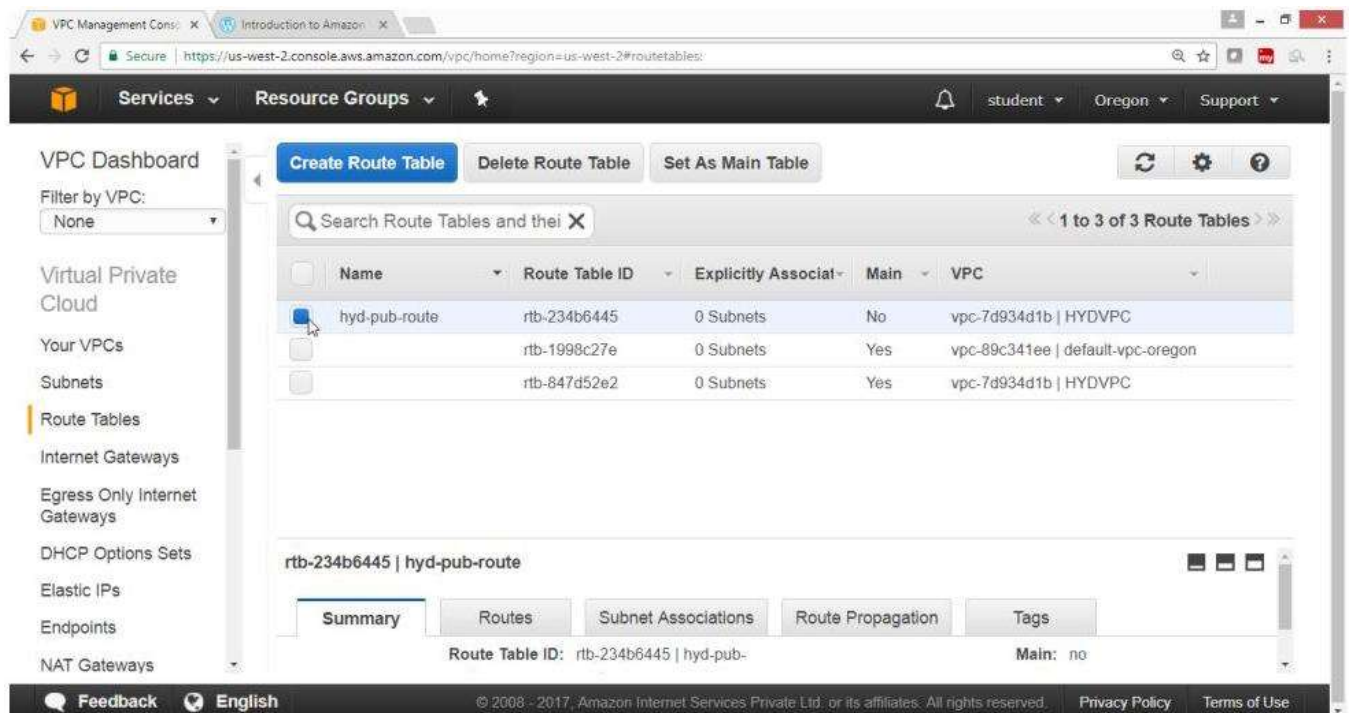
A route table specifies how packets are forwarded between the subnets within your VPC, the Internet, and your VPN connection.

Name tag ⓘ

VPC ⓘ

Cancel **Yes, Create**

Verify, hyd-pub-route table is created



The screenshot shows the AWS VPC Management Console. The left sidebar contains navigation links for VPC Dashboard, Virtual Private Cloud, Your VPCs, Subnets, Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, Elastic IPs, Endpoints, and NAT Gateways. The main content area displays a list of Route Tables. The 'hyd-pub-route' table is highlighted, showing its ID as 'rtb-234b6445', 0 Subnets associated, and it is not the main route table for the VPC 'vpc-7d934d1b | HYDVPC'. Below the list, the details for 'rtb-234b6445 | hyd-pub-route' are shown, with tabs for Summary, Routes, Subnet Associations, Route Propagation, and Tags. The 'Summary' tab is active, showing the Route Table ID and Main status.

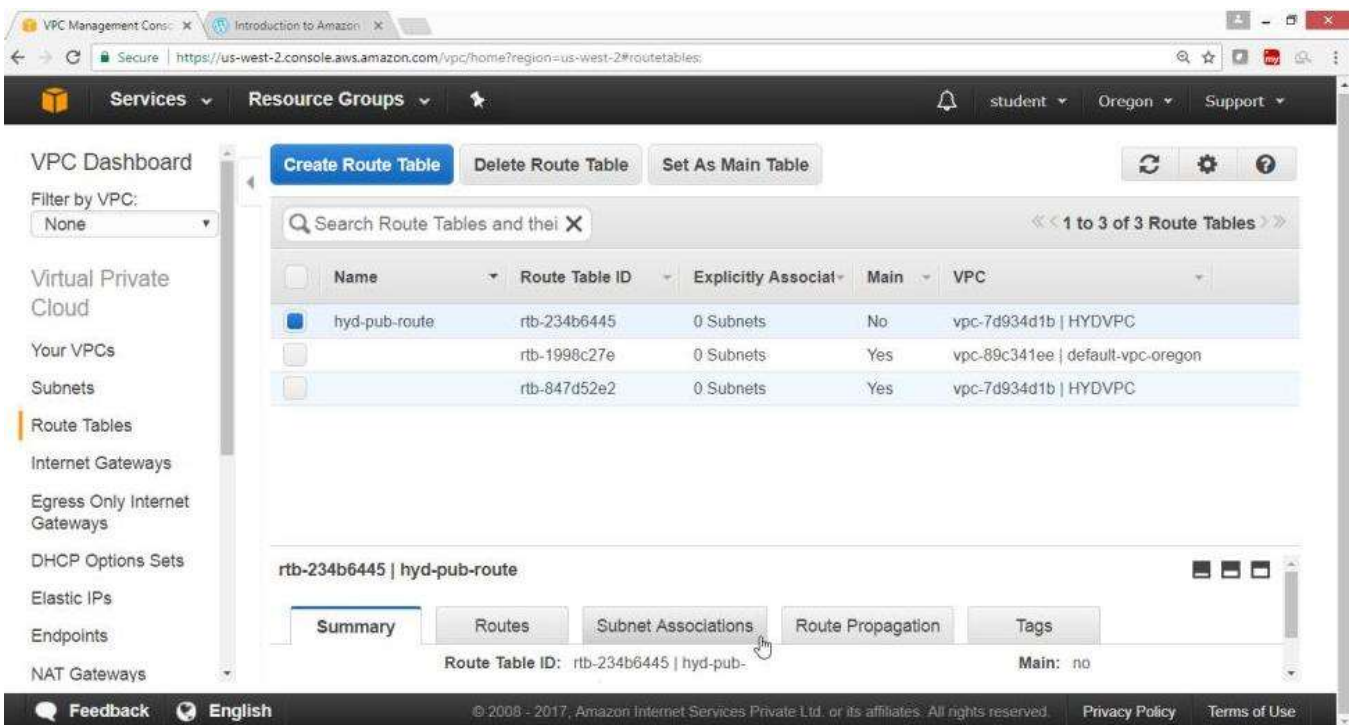
Name	Route Table ID	Explicitly Associat	Main	VPC
hyd-pub-route	rtb-234b6445	0 Subnets	No	vpc-7d934d1b HYDVPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
	rtb-847d52e2	0 Subnets	Yes	vpc-7d934d1b HYDVPC

rtb-234b6445 | hyd-pub-route

Summary Routes Subnet Associations Route Propagation Tags

Route Table ID: rtb-234b6445 | hyd-pub- Main: no

Click on "Subnet Association" button



This screenshot is identical to the previous one, but with the 'Subnet Associations' tab selected in the details view for the 'hyd-pub-route' table. The 'Summary' tab is no longer active, and the 'Subnet Associations' tab is highlighted, indicating the next step in the process.

Click on "Edit" button

The screenshot shows the AWS VPC console interface. On the left is a navigation sidebar with options like 'Virtual Private Cloud', 'Your VPCs', 'Subnets', 'Route Tables', etc. The main area displays a list of route tables. The 'hyd-pub-route' (rtb-234b6445) is selected. Below the list, there are tabs for 'Summary', 'Routes', 'Subnet Associations', 'Route Propagation', and 'Tags'. The 'Subnet Associations' tab is active, showing a message: 'You do not have any subnet associations. The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:'. An 'Edit' button is visible above the 'Subnet' column header.

Name	Route Table ID	Explicitly Associat	Main	VPC
hyd-pub-route	rtb-234b6445	0 Subnets	No	vpc-7d934d1b HYDVPC

rtb-234b6445 | hyd-pub-route

Summary Routes Subnet Associations Route Propagation Tags

Edit

Subnet IPv4 CIDR IPv6 CIDR

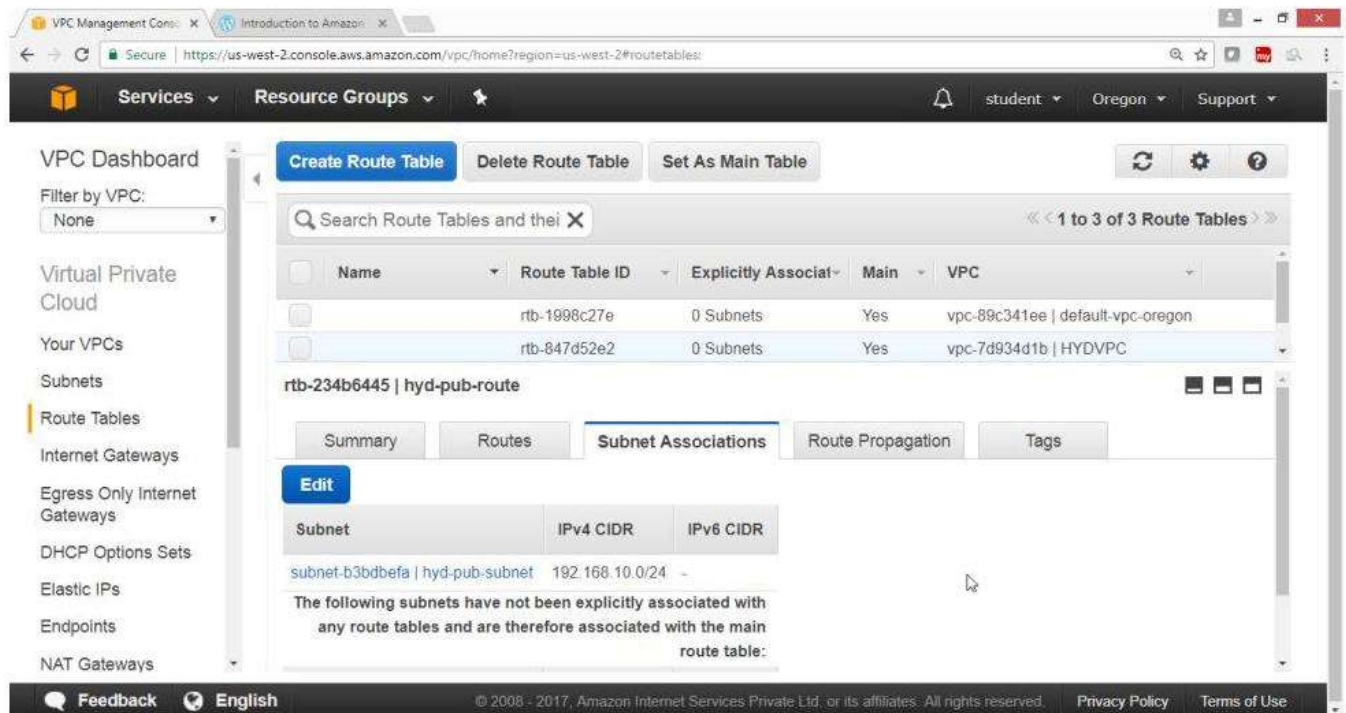
You do not have any subnet associations.
The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Select check box of hyd-pub-subnet ->192.168.10.0/24

This screenshot shows the 'Subnet Associations' tab for the 'hyd-pub-route'. It displays a table of subnets that are associated with the route table. The 'Associate' column has checkboxes for each subnet. The 'Save' button is highlighted, indicating the user is ready to save the changes.

Associate	Subnet	IPv4 CIDR	IPv6 CIDR	Current Route Table
<input checked="" type="checkbox"/>	subnet-b3bdbefa hyd-pub-subnet	192.168.10.0/24	-	Main
<input type="checkbox"/>	subnet-6abcbf23 hyd-pvt-subnet	192.168.20.0/24	-	Main

Verify hyd-pub-subnet is associated with routing table



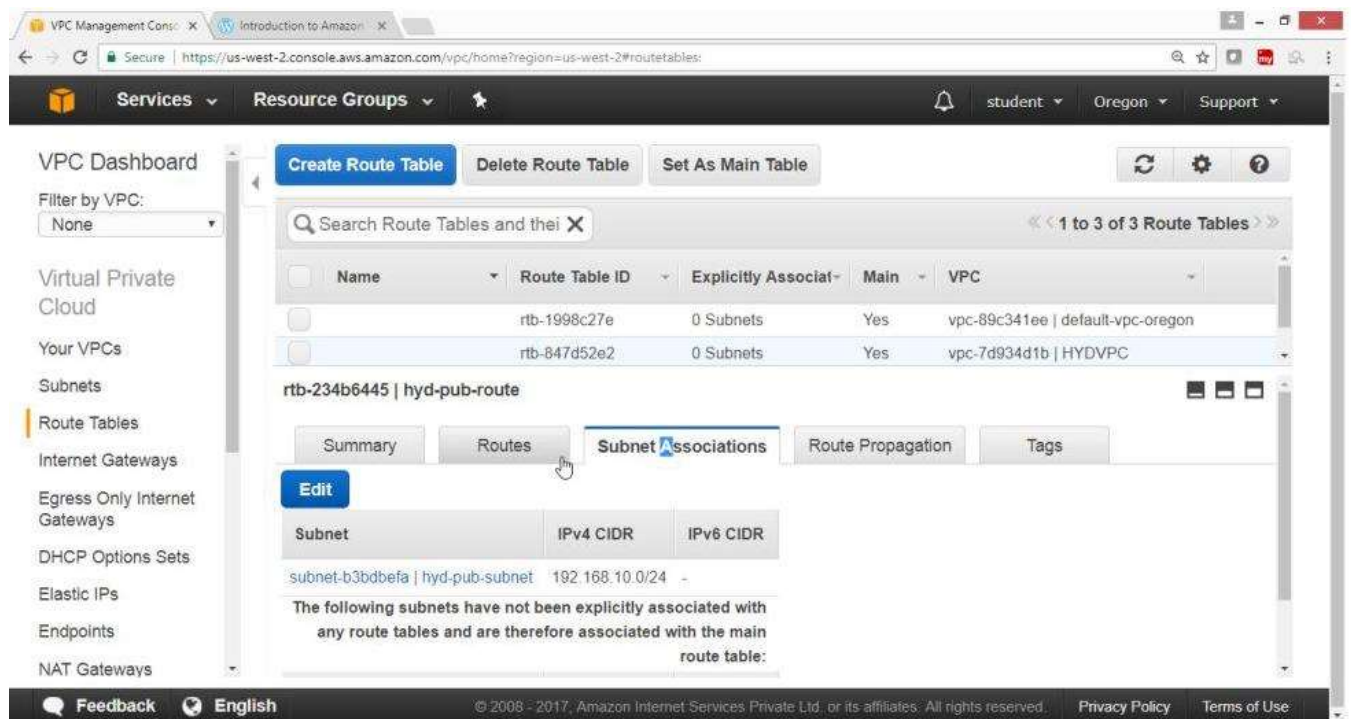
The screenshot shows the AWS VPC console interface. On the left is the VPC Dashboard with a sidebar menu. The main content area displays a list of route tables. The selected route table, **rtb-234b6445 | hyd-pub-route**, has its **Subnet Associations** tab active. This tab shows a table with one entry: **subnet-b3bdbefa | hyd-pub-subnet** with an IPv4 CIDR of **192.168.10.0/24**. Below the table, a message states: "The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:".

Name	Route Table ID	Explicitly Associat	Main	VPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
	rtb-847d52e2	0 Subnets	Yes	vpc-7d934d1b HYDVPC

Subnet	IPv4 CIDR	IPv6 CIDR
subnet-b3bdbefa hyd-pub-subnet	192.168.10.0/24	-

Click on "Route" Button

Click on "Edit" Button



This screenshot is similar to the first one, but the **Routes** tab is selected for the route table **rtb-234b6445 | hyd-pub-route**. The **Edit** button is highlighted with a mouse cursor. The rest of the interface, including the sidebar and the 'Subnet Associations' table, remains the same.

Click on "Add another route" Button

The screenshot shows the AWS VPC console interface. On the left is a navigation sidebar with options like 'Virtual Private Cloud', 'Your VPCs', 'Subnets', 'Route Tables', 'Internet Gateways', 'Egress Only Internet Gateways', 'DHCP Options Sets', 'Elastic IPs', 'Endpoints', and 'NAT Gateways'. The main area displays the 'Routes' tab for the selected route table 'rtb-234b6445 | hyd-pub-route'. It shows a table with one existing route: Destination '192.168.0.0/16' and Target 'local'. Below this table, the 'Add another route' button is highlighted with a mouse cursor. The interface also includes buttons for 'Create Route Table', 'Delete Route Table', and 'Set As Main Table' at the top, and a 'Cancel'/'Save' button pair below the route table tabs.

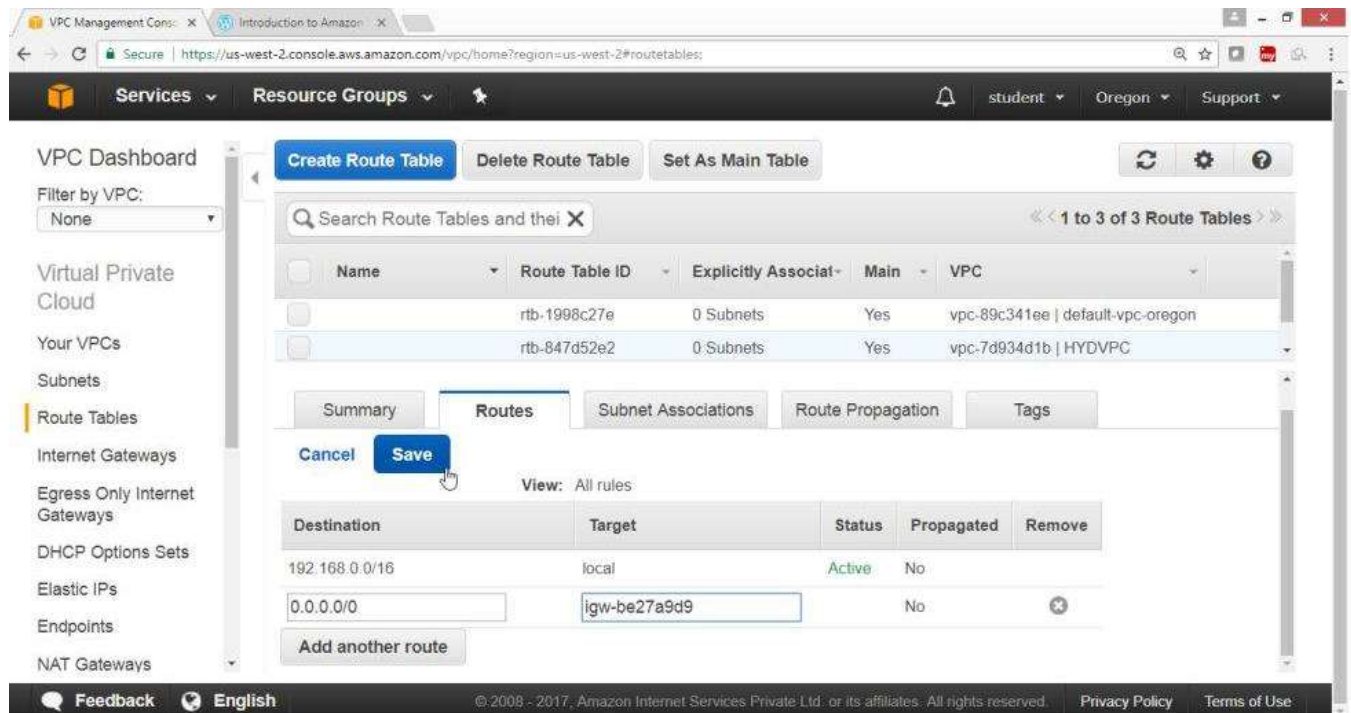
For Destination->0.0.0.0/0

For Target->select HYDIGW

Click on "Save" Button

This screenshot shows the same AWS VPC console interface as the previous one, but with the new route configuration entered. The 'Destination' field in the route table now contains '0.0.0.0/0'. The 'Target' dropdown menu is open, and 'igw-be27a9d9 | HYDIGW' is selected. The 'Save' button is highlighted with a mouse cursor. The rest of the interface, including the navigation sidebar and the existing route, remains the same.

Verification, Public route is added through Internet Gateway

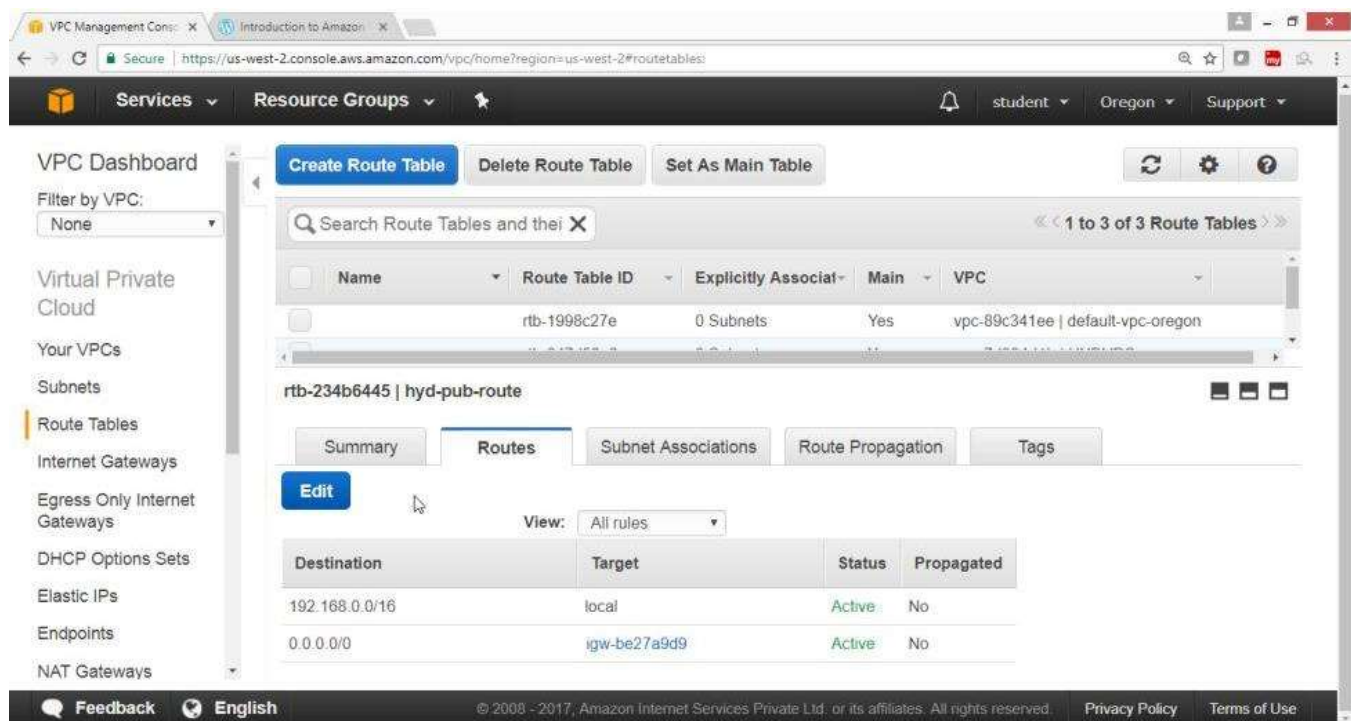


The screenshot shows the AWS VPC console interface. On the left is a navigation menu with options like 'Virtual Private Cloud', 'Your VPCs', 'Subnets', 'Route Tables', 'Internet Gateways', etc. The main area displays the 'Routes' tab for a selected route table. A table lists existing routes, and a new route is being added with the following details:

Destination	Target	Status	Propagated	Remove
192.168.0.0/16	local	Active	No	
0.0.0.0/0	igw-be27a9d9		No	

The 'Save' button is highlighted, indicating the new route is being confirmed.

Verify, Status column show active



This screenshot shows the same AWS VPC console interface, but now the 'Status' column for the new route (0.0.0.0/0) is 'Active'. The 'Edit' button is highlighted, indicating the route has been successfully updated.

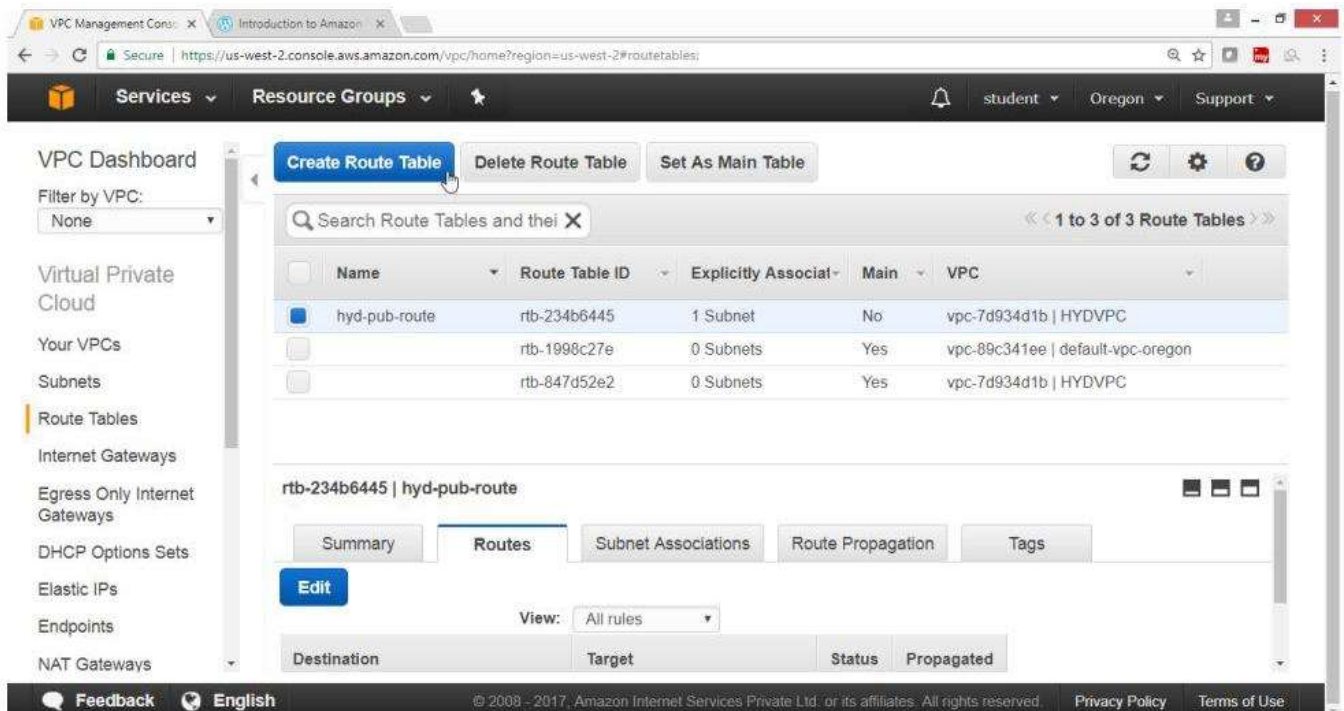
Destination	Target	Status	Propagated
192.168.0.0/16	local	Active	No
0.0.0.0/0	igw-be27a9d9	Active	No

6) Create Private Routing Table, associate subnet and add routing rules

On "VPC Dashboard" panel

Select Route Tables

Click on "Create Route Table"



On "Create Route Table" box

For Name tag -> hyd-pvt-route

For VPC->HYDVPC

Click on "Yes, Create" button

Create Route Table

A route table specifies how packets are forwarded between the subnets within your VPC, the Internet, and your VPN connection.

Name tag

VPC

[Cancel](#) [Yes, Create](#)

Verify

hyd-pvt-route table is created

VPC Management Console - Introduction to Amazon VPC

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#routeTables

Services | Resource Groups | student | Oregon | Support

VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Create Route Table | Delete Route Table | Set As Main Table

Search Route Tables and their associated VPCs

<< 1 to 4 of 4 Route Tables >>

Name	Route Table ID	Explicitly Associated	Main	VPC
hyd-pvt-route	rtb-ac446bca	0 Subnets	No	vpc-7d934d1b HYDVPC
hyd-pub-route	rtb-234b6445	1 Subnet	No	vpc-7d934d1b HYDVPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
	rtb-847d52e2	0 Subnets	Yes	vpc-7d934d1b HYDVPC

rtb-ac446bca | hyd-pvt-route

Summary | Routes | Subnet Associations | Route Propagation | Tags

Edit

View: All rules

Destination	Target	Status	Propagated
-------------	--------	--------	------------

Feedback | English | © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

Click on "Subnet Association" Button

VPC Management Console - Introduction to Amazon VPC

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#routeTables

Services | Resource Groups | student | Oregon | Support

VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Create Route Table | Delete Route Table | Set As Main Table

Search Route Tables and their associated VPCs

<< 1 to 4 of 4 Route Tables >>

Name	Route Table ID	Explicitly Associated	Main	VPC
hyd-pvt-route	rtb-ac446bca	0 Subnets	No	vpc-7d934d1b HYDVPC
hyd-pub-route	rtb-234b6445	1 Subnet	No	vpc-7d934d1b HYDVPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
	rtb-847d52e2	0 Subnets	Yes	vpc-7d934d1b HYDVPC

rtb-ac446bca | hyd-pvt-route

Summary | Routes | Subnet Associations | Route Propagation | Tags

Edit

View: All rules

Destination	Target	Status	Propagated
-------------	--------	--------	------------

Feedback | English | © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

Click on Edit button

The screenshot shows the AWS VPC console interface. On the left is a navigation menu with options like 'Virtual Private Cloud', 'Your VPCs', 'Subnets', 'Route Tables', 'Internet Gateways', etc. The main area displays the 'Subnet Associations' tab for the selected route table 'rtb-ac446bca | hyd-pvt-route'. Below the tabs, there are buttons for 'Summary', 'Routes', 'Subnet Associations', 'Route Propagation', and 'Tags'. The 'Subnet Associations' section shows a table with columns: Associate, Subnet, IPv4 CIDR, IPv6 CIDR, and Current Route Table. The 'Edit' button is highlighted with a mouse cursor.

Name	Route Table ID	Explicitly Associat	Main	VPC
hyd-pvt-route	rtb-ac446bca	0 Subnets	No	vpc-7d934d1b HYDVPC
hyd-pub-route	rtb-234b6445	1 Subnet	No	vpc-7d934d1b HYDVPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
	rtb-847d52e2	0 Subnets	Yes	vpc-7d934d1b HYDVPC

Associate	Subnet	IPv4 CIDR	IPv6 CIDR	Current Route Table
<input type="checkbox"/>	subnet-b3bdbefa hyd-pub-subnet	192.168.10.0/24	-	rtb-234b6445 hyd-pub-route
<input checked="" type="checkbox"/>	subnet-6abcbf23 hyd-pvt-subnet	192.168.20.0/24	-	Main

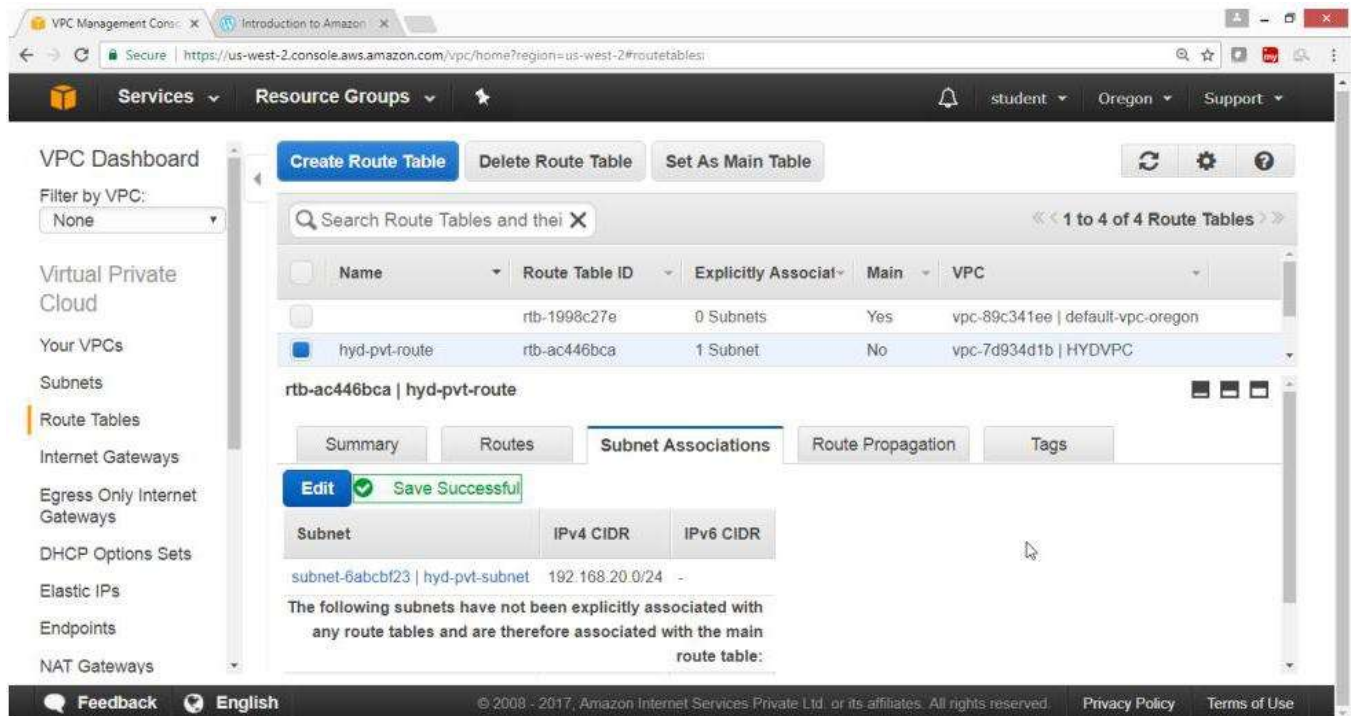
Select checkbox hyd-pvt-subnet -> 192.168.20.0/24

This screenshot shows the same AWS VPC console interface as the previous one, but now the 'Save' button is highlighted with a mouse cursor. The 'Subnet Associations' table is visible, showing the association for 'subnet-6abcbf23 | hyd-pvt-subnet' with the IPv4 CIDR '192.168.20.0/24'.

Name	Route Table ID	Explicitly Associat	Main	VPC
hyd-pvt-route	rtb-ac446bca	0 Subnets	No	vpc-7d934d1b HYDVPC
hyd-pub-route	rtb-234b6445	1 Subnet	No	vpc-7d934d1b HYDVPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
	rtb-847d52e2	0 Subnets	Yes	vpc-7d934d1b HYDVPC

Associate	Subnet	IPv4 CIDR	IPv6 CIDR	Current Route Table
<input type="checkbox"/>	subnet-b3bdbefa hyd-pub-subnet	192.168.10.0/24	-	rtb-234b6445 hyd-pub-route
<input checked="" type="checkbox"/>	subnet-6abcbf23 hyd-pvt-subnet	192.168.20.0/24	-	Main

Click on "Save" Button



VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Create Route Table Delete Route Table Set As Main Table

Search Route Tables and their Subnets

<< 1 to 4 of 4 Route Tables >>

Name	Route Table ID	Explicitly Associated	Main	VPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
hyd-pvt-route	rtb-ac446bca	1 Subnet	No	vpc-7d934d1b HYDVPC

rtb-ac446bca | hyd-pvt-route

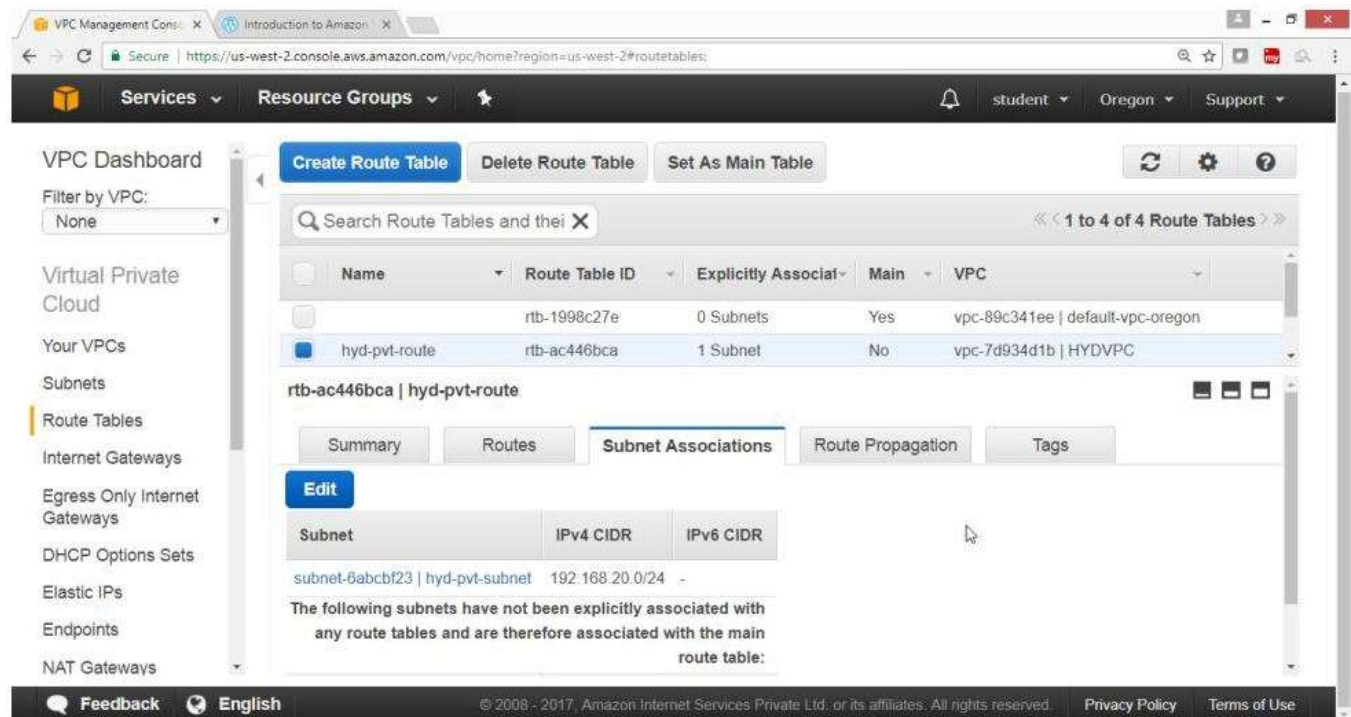
Summary Routes Subnet Associations Route Propagation Tags

Edit Save Successful

Subnet	IPv4 CIDR	IPv6 CIDR
subnet-6abcbf23 hyd-pvt-subnet	192.168.20.0/24	-

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Verify, Hyd-pvt-subnet is associated with hyd-pvt-route table



VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Create Route Table Delete Route Table Set As Main Table

Search Route Tables and their Subnets

<< 1 to 4 of 4 Route Tables >>

Name	Route Table ID	Explicitly Associated	Main	VPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
hyd-pvt-route	rtb-ac446bca	1 Subnet	No	vpc-7d934d1b HYDVPC

rtb-ac446bca | hyd-pvt-route

Summary Routes Subnet Associations Route Propagation Tags

Edit

Subnet	IPv4 CIDR	IPv6 CIDR
subnet-6abcbf23 hyd-pvt-subnet	192.168.20.0/24	-

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Click on "Route" Button

The screenshot shows the AWS VPC console interface. On the left is a navigation sidebar with options like 'Virtual Private Cloud', 'Your VPCs', 'Subnets', 'Route Tables', 'Internet Gateways', 'Egress Only Internet Gateways', 'DHCP Options Sets', 'Elastic IPs', 'Endpoints', and 'NAT Gateways'. The 'Route Tables' section is selected. The main area displays a list of route tables. The table has columns: Name, Route Table ID, Explicitly Associated, Main, and VPC. Two route tables are listed: 'rtb-1998c27e' (Main: Yes, VPC: vpc-89c341ee | default-vpc-oregon) and 'hyd-pvt-route' (Main: No, VPC: vpc-7d934d1b | HYDVPC). The 'hyd-pvt-route' is selected. Below the list, the 'Subnet Associations' tab is active. It shows a table with columns: Subnet, IPv4 CIDR, and IPv6 CIDR. One association is listed: 'subnet-6abcbf23 | hyd-pvt-subnet' with IPv4 CIDR '192.168.20.0/24'. Below this table, a message states: 'The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:'. The bottom of the console shows a footer with 'Feedback', 'English', and copyright information.

Name	Route Table ID	Explicitly Associated	Main	VPC
	rtb-1998c27e	0 Subnets	Yes	vpc-89c341ee default-vpc-oregon
hyd-pvt-route	rtb-ac446bca	1 Subnet	No	vpc-7d934d1b HYDVPC

Subnet	IPv4 CIDR	IPv6 CIDR
subnet-6abcbf23 hyd-pvt-subnet	192.168.20.0/24	-

Note: No need to add IGW in pvt route

This screenshot shows the 'Routes' tab for the same 'hyd-pvt-route' (rtb-ac446bca). The 'Routes' tab is active, showing a table with columns: Destination, Target, Status, and Propagated. A single route is listed with Destination '192.168.0.0/16', Target 'local', Status 'Active', and Propagated 'No'. The 'View' dropdown is set to 'All rules'. The bottom of the console shows the same footer as the previous screenshot.

Destination	Target	Status	Propagated
192.168.0.0/16	local	Active	No

7) To launch Windows instance in Public Subnet

Open the AWS console

Click on Services

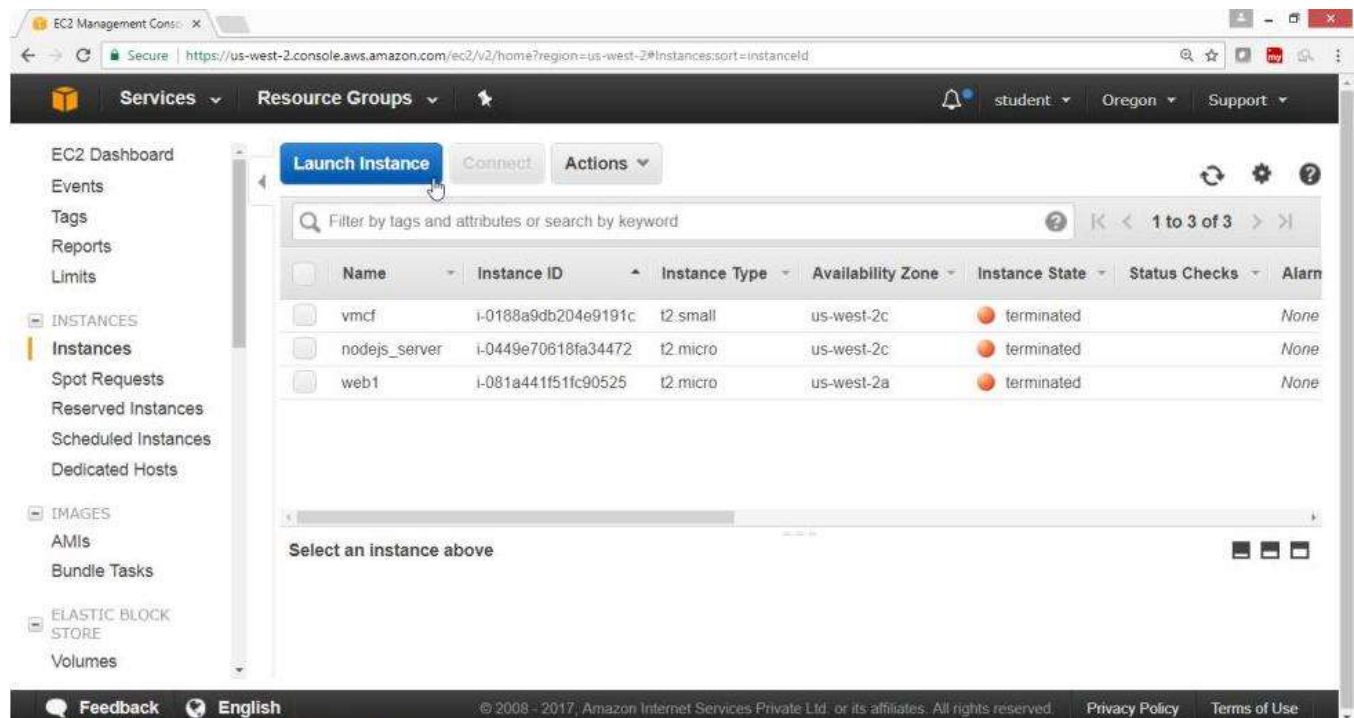
Click on EC2 Services

The image shows two screenshots of the AWS Management Console. The top screenshot is the 'Services' page for the 'us-west-2' region. It features a search bar and a grid of service categories: Compute (EC2, EC2 Container Service, Lightsail, Elastic Beanstalk, Lambda, Batch), Developer Tools (CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, X-Ray), Analytics (Athena, EMR, CloudSearch, Elasticsearch Service, Kinesis, Data Pipeline, QuickSight), Application Services (Step Functions, SWF, API Gateway, Elastic Transcoder), Storage (S3, EFS, Glacier, Storage Gateway), Management Tools (CloudWatch, CloudFormation, CloudTrail, Config, OpsWorks, Service Catalog, Trusted Advisor), Database, Artificial Intelligence (Lex, Polly, Rekognition, Machine Learning), Messaging (Simple Queue Service, Simple Notification Service, SES), Business Productivity (WorkDocs, WorkMail, Amazon Chime), and Internet of Things. The bottom screenshot is the 'EC2 Management Console' for the 'us-west-2' region. It shows the 'Resources' section with a summary of EC2 resources: 0 Running Instances, 0 Elastic IPs, 0 Dedicated Hosts, 0 Snapshots, 0 Volumes, 0 Load Balancers, 0 Key Pairs, 2 Security Groups, and 0 Placement Groups. A promotional banner for Amazon Lightsail is visible. The 'Create Instance' section provides instructions on how to launch a virtual server. The right sidebar contains 'Account Attributes' (Supported Platforms, Default VPC, Resource ID length management) and 'Additional Information' (Getting Started Guide, Documentation, All EC2 Resources, Forums, Pricing, Contact Us). The footer includes a feedback link, language selection (English), and copyright information (© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.).

On the EC2 Dashboard Panel

Click on Instance

Click on Launch Instance Button



EC2 Management Console

Services Resource Groups

student Oregon Support

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Spot Requests

Reserved Instances

Scheduled Instances

Dedicated Hosts

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Launch Instance Connect Actions

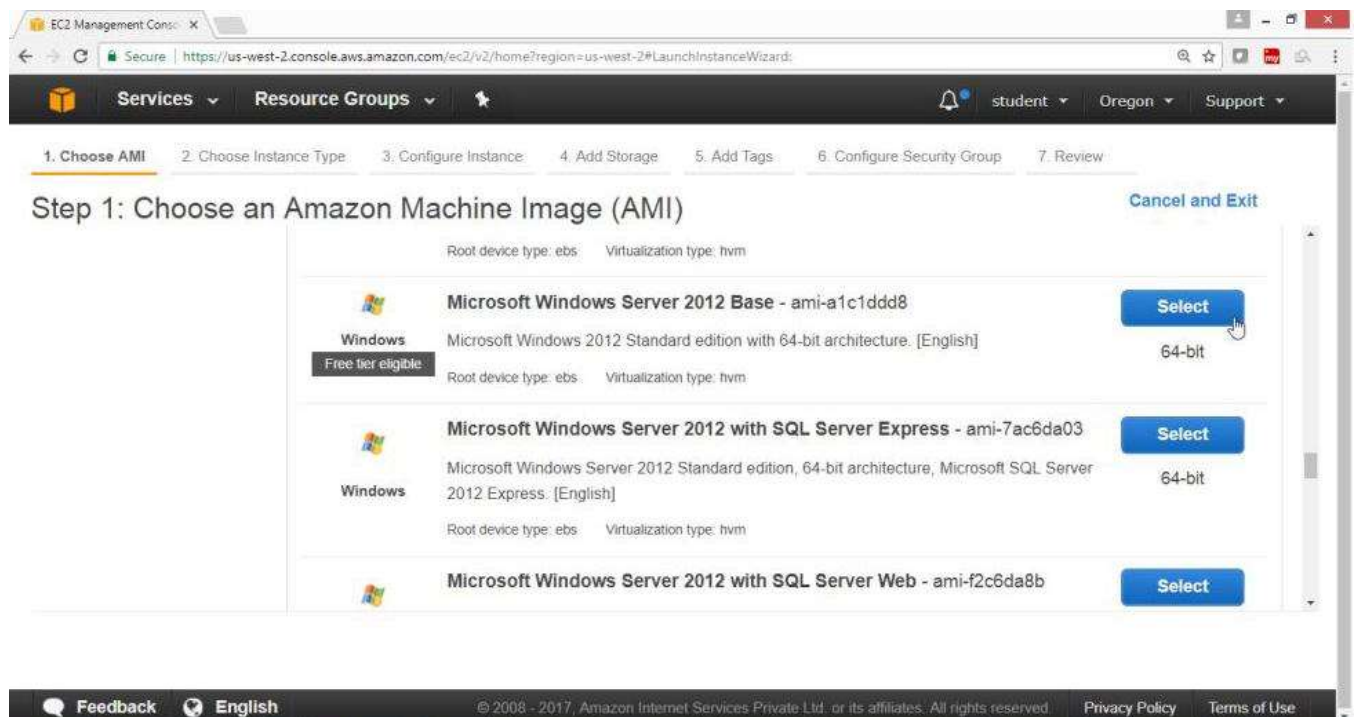
Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm
vmcf	i-0188a9db204e9191c	t2.small	us-west-2c	terminated		None
nodejs_server	i-0449e70618fa34472	t2.micro	us-west-2c	terminated		None
web1	i-081a441f51fc90525	t2.micro	us-west-2a	terminated		None

Select an instance above

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Select AMI "Microsoft Windows Server 2012 Base-ami-a1c1ddd8" Free tier eligible



EC2 Management Console

Services Resource Groups




student Oregon Support

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)

Cancel and Exit

Root device type: ebs Virtualization type: hvm

 Windows Free tier eligible	Microsoft Windows Server 2012 Base - ami-a1c1ddd8 Microsoft Windows 2012 Standard edition with 64-bit architecture. [English] Root device type: ebs Virtualization type: hvm	Select 64-bit
 Windows	Microsoft Windows Server 2012 with SQL Server Express - ami-7ac6da03 Microsoft Windows Server 2012 Standard edition, 64-bit architecture, Microsoft SQL Server 2012 Express. [English] Root device type: ebs Virtualization type: hvm	Select 64-bit
 Windows	Microsoft Windows Server 2012 with SQL Server Web - ami-f2c6da8b	Select

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

On the "choose an Instance Type" Page

Select "General purposed2.micro"

Click on "Next Configure Instance Details" Button

Step 2: Choose an Instance Type

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

Cancel Previous Review and Launch Next: Configure Instance Details

On the Configuration Instance Details" Page

- For "Number of Instances" ->1
- For "Network"->HYDVPC
- For "Subnet"->hyd-pub-subnet
- For "Auto-assign Public IP" -> Enable
- Click on "Next: Add Storage" Button

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-7d934d1b | HYDVPC Create new VPC

Subnet subnet-b3bdbefa | hyd-pub-subnet | us-west-2a Create new subnet
251 IP Addresses available

Auto-assign Public IP Enable

Domain join directory None Create new directory

Cancel Previous Review and Launch Next: Add Storage

On the "Add Storage" page

Take default values

Click on "Next: Add tags" button

The screenshot shows the AWS Management Console interface for the 'Add Storage' step of an EC2 instance launch wizard. The breadcrumb trail at the top indicates the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage (current step), 5. Add Tags, 6. Configure Security Group, and 7. Review. The page title is 'Step 4: Add Storage'. Below the title, a paragraph explains that the instance will be launched with the following storage device settings and that additional EBS volumes can be attached before or after launch. A table lists the storage settings for the root volume. The table has columns for Volume Type, Device, Snapshot, Size (GiB), Volume Type, IOPS, Throughput (MB/s), Delete on Termination, and Encrypted. The root volume is shown with a size of 30 GiB, General Purpose (SSD) volume type, 100 / 3000 IOPS, N/A throughput, and is not encrypted. Below the table is an 'Add New Volume' button. At the bottom of the page, there are buttons for 'Cancel', 'Previous', 'Review and Launch', and 'Next: Add Tags'. The 'Next: Add Tags' button is highlighted with a mouse cursor. The footer contains a feedback link, language selection (English), copyright information (© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.), and links to the Privacy Policy and Terms of Use.

EC2 Management Console

Services Resource Groups

student Oregon 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-01e5be77f781e7266	30	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

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

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

Click on "Add tag" button

EC2 Management Console

Services Resource Groups

student Oregon 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
This resource currently has no tags.			
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)

Cancel Previous **Review and Launch** Next: Configure Security Group

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

For "Key" -> Name

For Value ->Winpubvm

Click on "Next: Configure Security Group"

EC2 Management Console

Services Resource Groups

student Oregon 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	Winpubvm	<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**

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

On the "Configure Security Group"

Take Default Values

Click on "Review and Launch" Button

The screenshot shows the 'Step 6: Configure Security Group' page in the AWS Management Console. The breadcrumb trail at the top indicates the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The page title is 'Step 6: Configure Security Group'. Below the title, there is a paragraph explaining security groups. The 'Assign a security group' section has two radio buttons: 'Create a new security group' (selected) and 'Select an existing security group'. Below this, the 'Security group name' is 'launch-wizard-1' and the 'Description' is 'launch-wizard-1 created 2017-07-31T05:02:04.626+05:30'. A table with columns 'Type', 'Protocol', 'Port Range', and 'Source' contains one rule: Type 'RDP', Protocol 'TCP', Port Range '3389', and Source 'Custom' with IP '0.0.0.0/0'. An 'Add Rule' button is below the table. At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Review and Launch' (highlighted with a mouse cursor). The footer includes 'Feedback', 'English', copyright information, and links to 'Privacy Policy' and 'Terms of Use'.

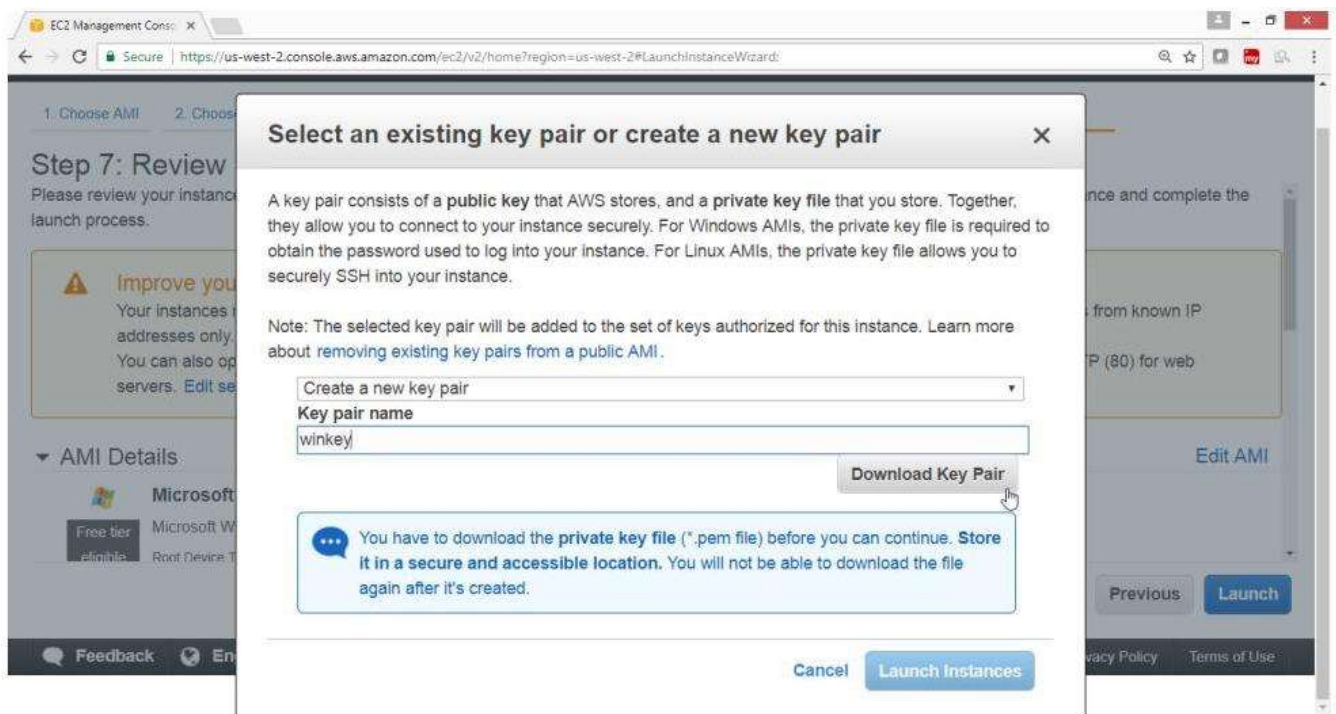
Click on "Launch" button

The screenshot shows the 'Step 7: Review Instance Launch' page in the AWS Management Console. The breadcrumb trail at the top indicates the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The page title is 'Step 7: Review Instance Launch'. Below the title, there is a paragraph explaining the review process. A yellow warning box contains the text: 'Improve your instances' security. Your security group, launch-wizard-1, 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'. Below the warning box, there is a section for 'AMI Details' with a dropdown arrow and an 'Edit AMI' link. The AMI is 'Microsoft Windows Server 2012 Base - ami-a1c1ddd8' with a 'Free tier eligible' badge. Below the AMI details, there are three buttons: 'Cancel', 'Previous', and 'Launch' (highlighted with a mouse cursor). A tooltip 'Define key pair and launch' is visible near the 'Launch' button. The footer includes 'Feedback', 'English', copyright information, and links to 'Privacy Policy' and 'Terms of Use'.

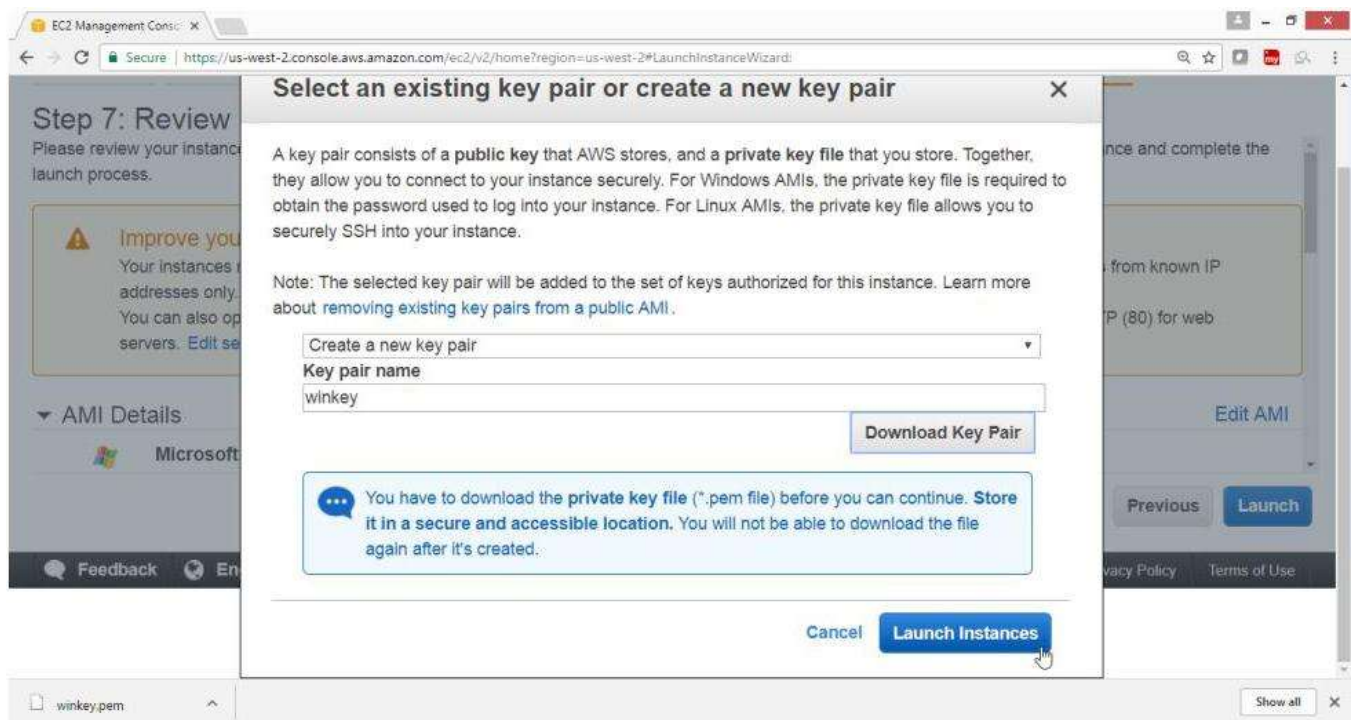
Select "Create a new key pair"

For "Key pair name" -> winkey

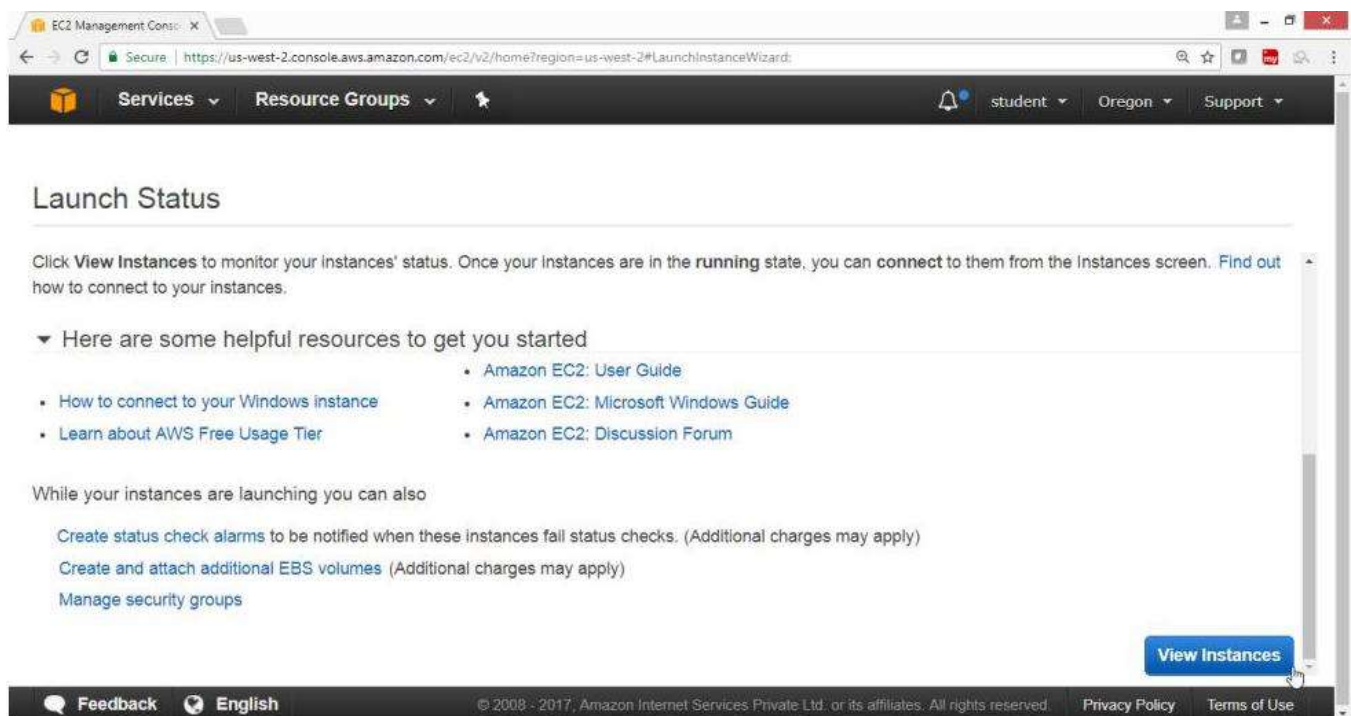
Click on "Download Key Pair"



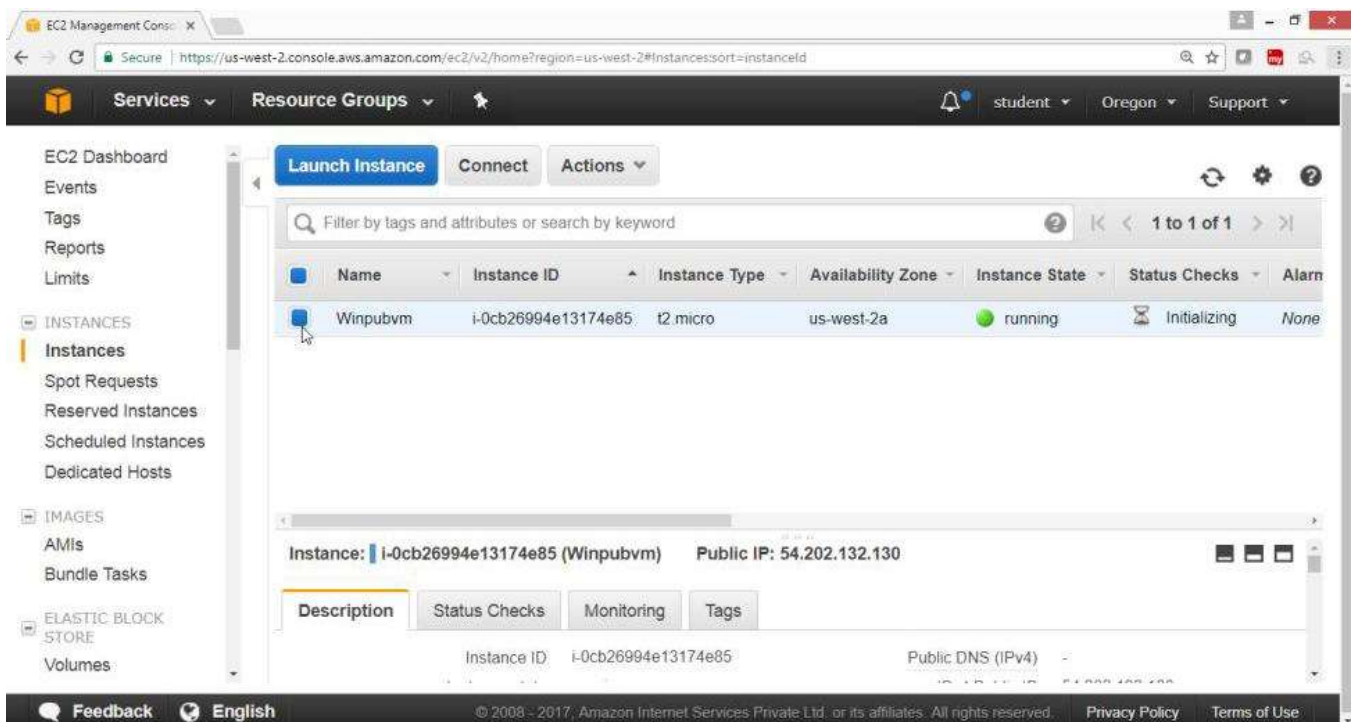
Click on "Launch Instance" Button



Check Summary, Drag down Click on "View Instance" Button



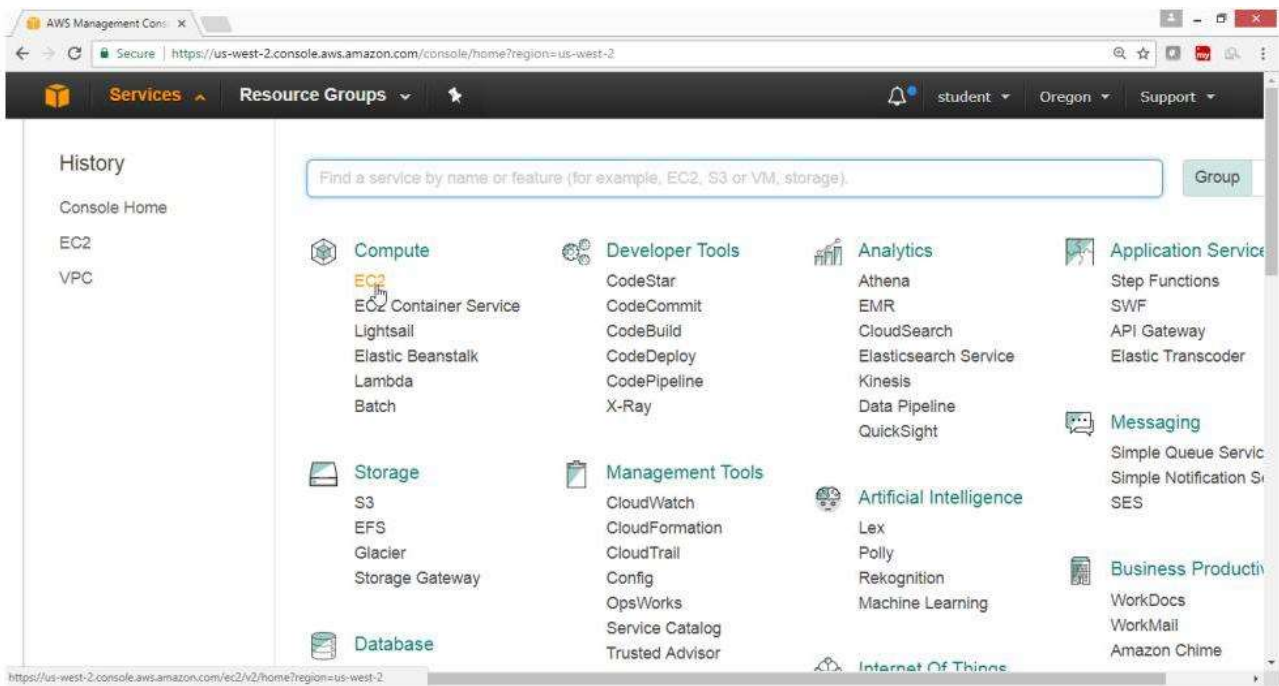
Verify that instance is Running



8) To Launch Windows Instance in Private Subnet under HYDVPC VPC

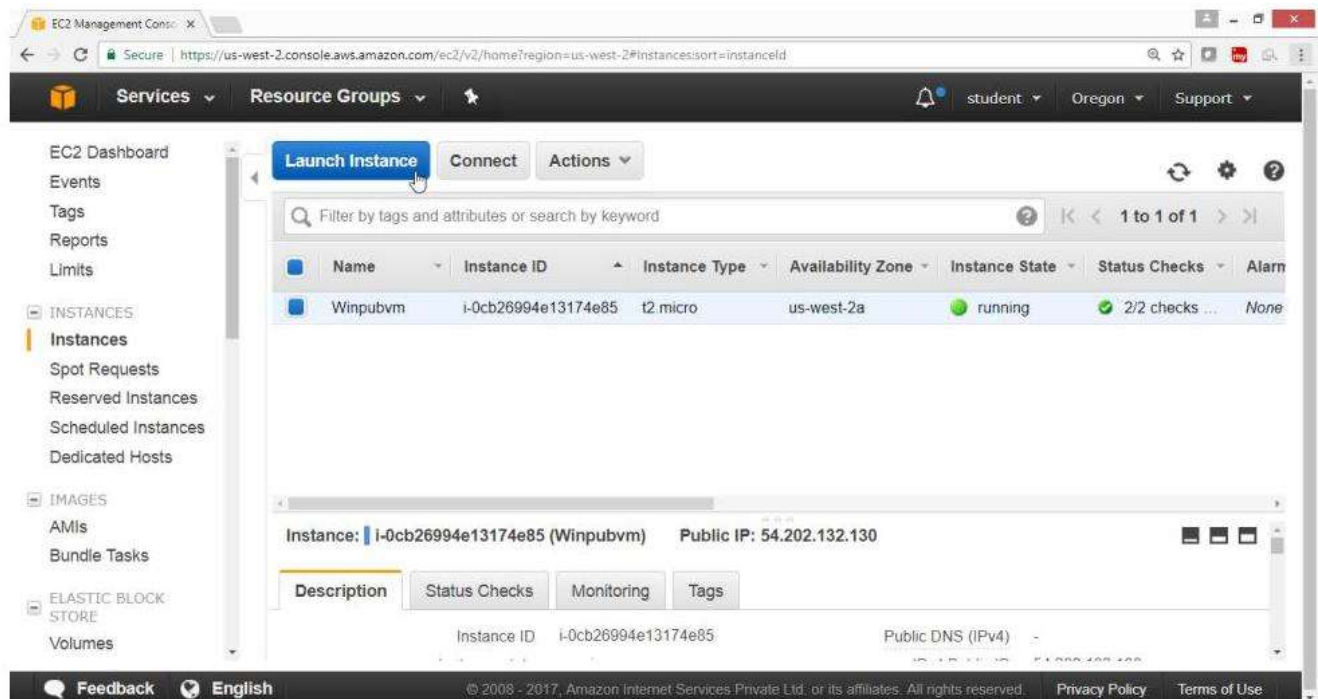
Open the AWS console

- Click on Services
- Click on EC2 Services



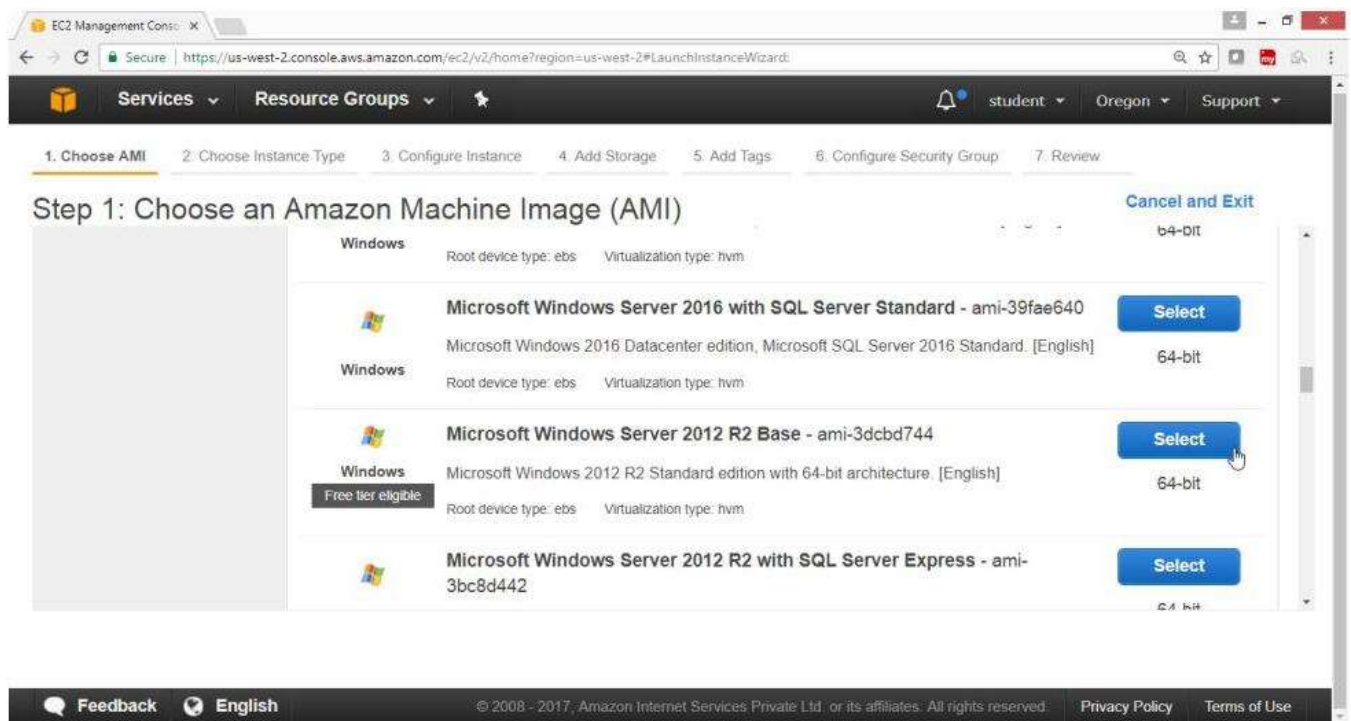
On the EC2 Dashboard Panel

- Click on Instance
- Click on "Launch Instance" Button



On the "Choose an Amazon Machine Image (AMI)" page

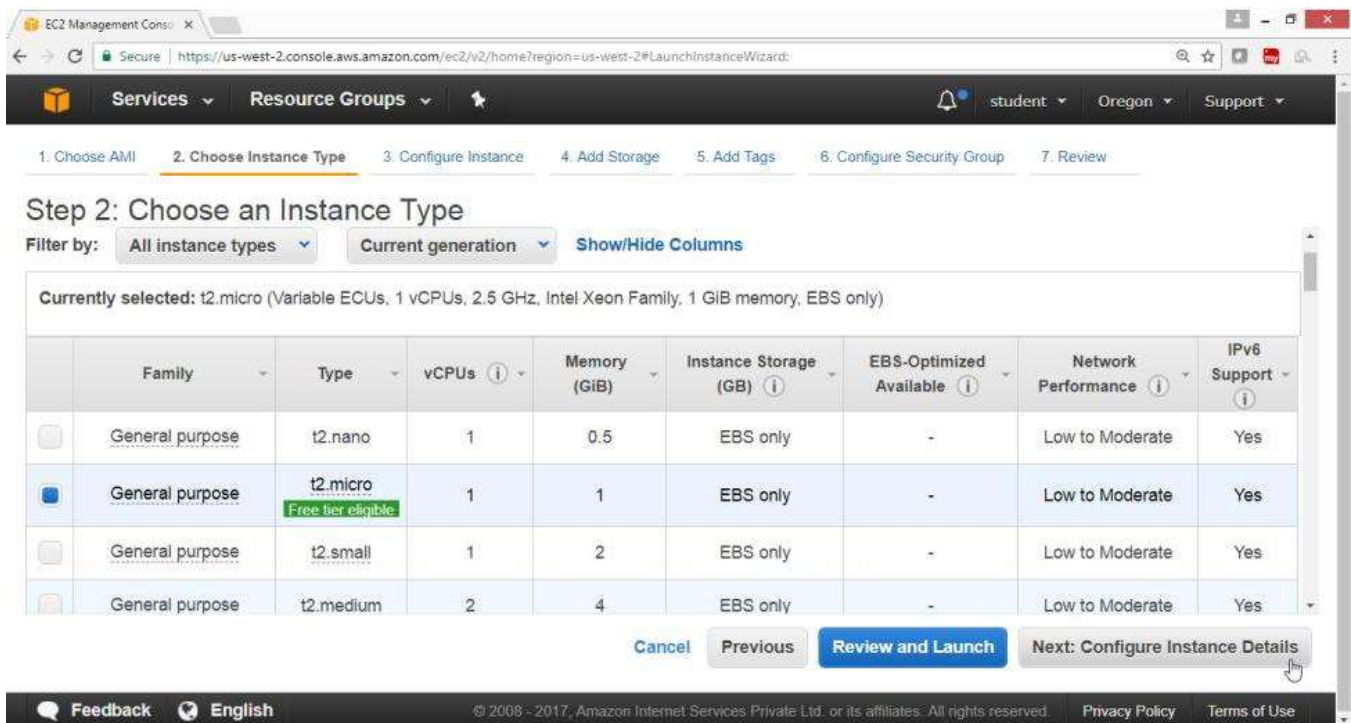
Select AMI "Microsoft Windows Server 2012 Base-ami-a1c1ddd8" Free tier eligible



On the "choose an Instance Type" Page

Select "General purposed2.micro"

Click on "Next Configure Instance Details" Button



On the Configuration Instance Details" Page

- For "Number of Instances" ->1
- For "Network"->HYDVPC
- For "Subnet"->hyd-pvt-subnet
- For "Auto-assign Public IP" -> Disabled
- Click on "Next: Add Storage" Button

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-7d934d1b | HYDVPC [Create new VPC](#)

Subnet: subnet-6abcbf23 | hyd-pvt-subnet | us-west-2a [Create new subnet](#)
251 IP Addresses available

Auto-assign Public IP: Disable

Domain join directory: None [Create new directory](#)

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

On the "Add Storage" page

Take default values

Click on "Next: Add tags" button

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-08c5b8b7p19187ab8	30	General Purpose E	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

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

Click on "Add tag" button

EC2 Management Console

Services Resource Groups

student Oregon 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
This resource currently has no tags			
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)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Security Group](#)

[Feedback](#) [English](#) © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

For "Key" -> Name

For Value ->Winpubvm

Click on "Next: Configure Security Group"

EC2 Management Console

Services Resource Groups

student Oregon 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	Winpubvm	<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](#)

[Feedback](#) [English](#) © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

On the "Configure Security Group"

Take Default Values

Click on "Review and Launch" Button

The screenshot shows the 'Step 6: Configure Security Group' page in the AWS Management Console. The breadcrumb trail at the top indicates the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The page title is 'Step 6: Configure Security Group'. Below the title, there is a paragraph explaining security groups. The 'Assign a security group' section has two radio buttons: 'Create a new security group' (selected) and 'Select an existing security group'. Below this, the 'Security group name' is 'launch-wizard-2' and the 'Description' is 'launch-wizard-2 created 2017-07-31T05:27:45.080+05:30'. A table lists the configured rules with columns: Type, Protocol, Port Range, and Source. One rule is shown: Type 'RDP', Protocol 'TCP', Port Range '3389', and Source 'Custom 0.0.0.0/0'. At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Review and Launch'. The 'Review and Launch' button is highlighted with a mouse cursor.

Type	Protocol	Port Range	Source
RDP	TCP	3389	Custom 0.0.0.0/0

Click on "Launch" button

The screenshot shows the 'Step 7: Review Instance Launch' page in the AWS Management Console. The breadcrumb trail at the top indicates the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The page title is 'Step 7: Review Instance Launch'. Below the title, there is a paragraph explaining the review process. A yellow warning box states: 'Improve your instances' security. Your security group, launch-wizard-1, 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'. Below the warning box, the 'AMI Details' section shows 'Microsoft Windows Server 2012 Base - ami-a1c1ddd8' with a 'Free tier eligible' badge. At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Launch'. The 'Launch' button is highlighted with a mouse cursor, and a tooltip 'Define key pair and launch' is visible.

Microsoft Windows Server 2012 Base - ami-a1c1ddd8

Free tier eligible

Microsoft Windows 2012 Standard edition with 64-bit architecture. [English]

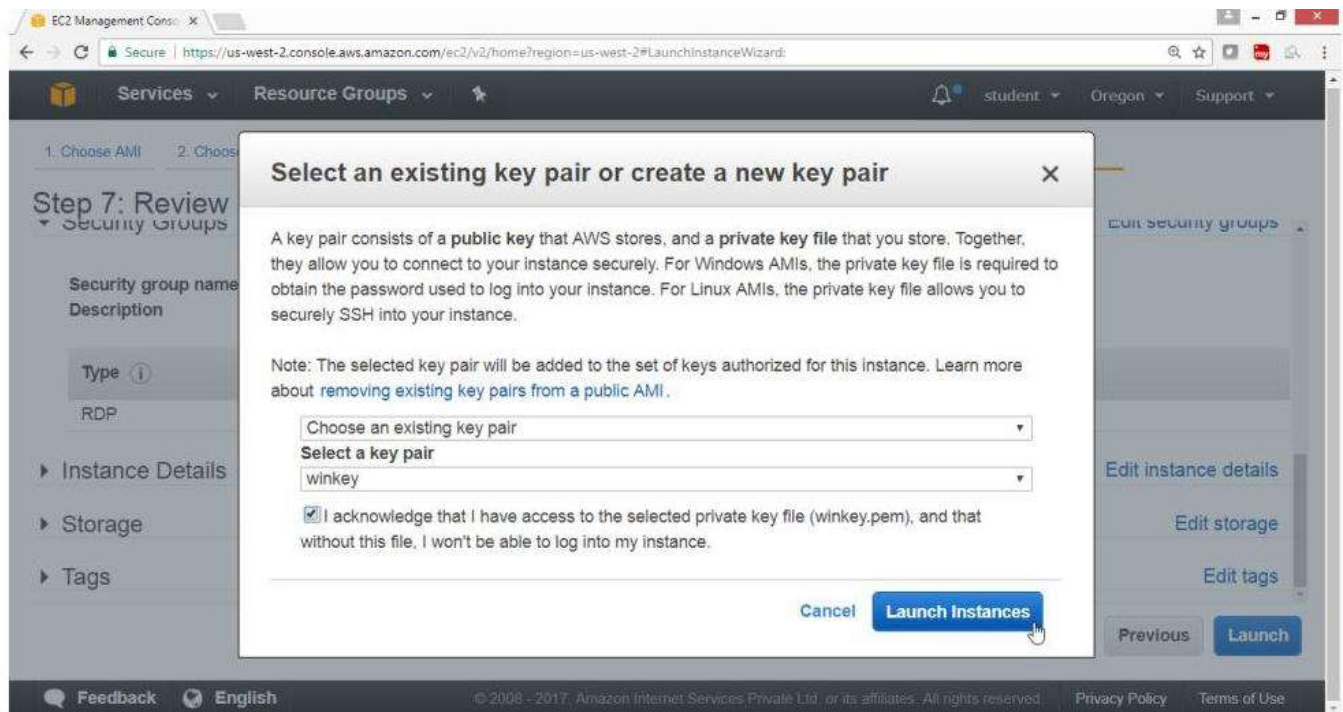
Root Device Type: ebs Virtualization type: hvm

Select "Create a new key pair"

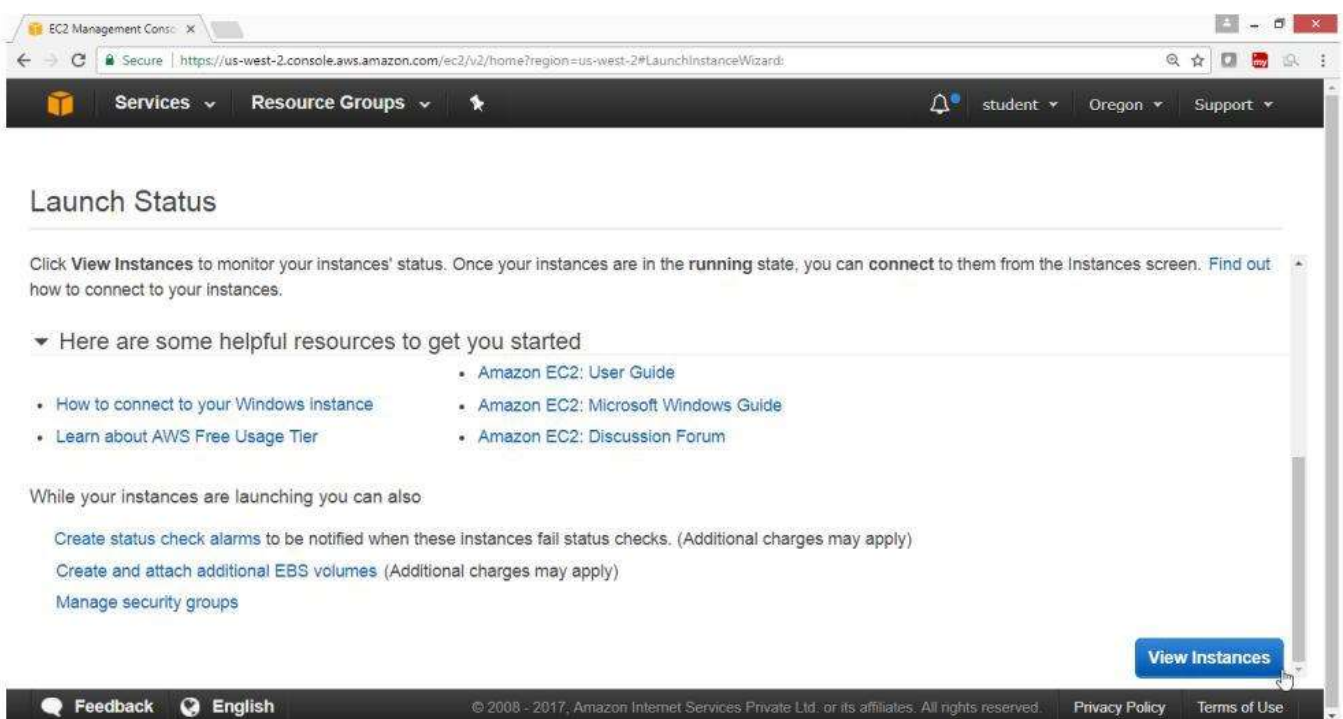
For "Key pair name" -> winkey

Click on "Download Key Pair"

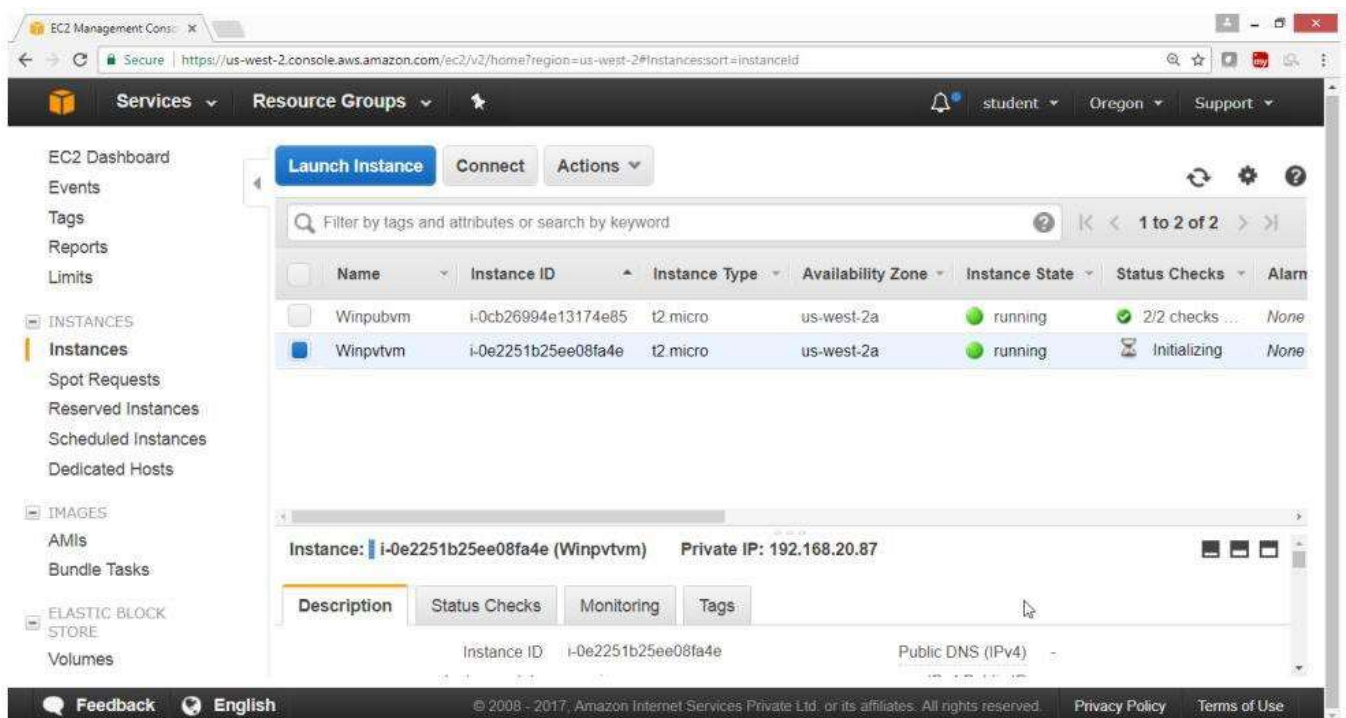
Click on "Launch Instance" Button



Check Summary, Drag down
Click on "View Instance" Button

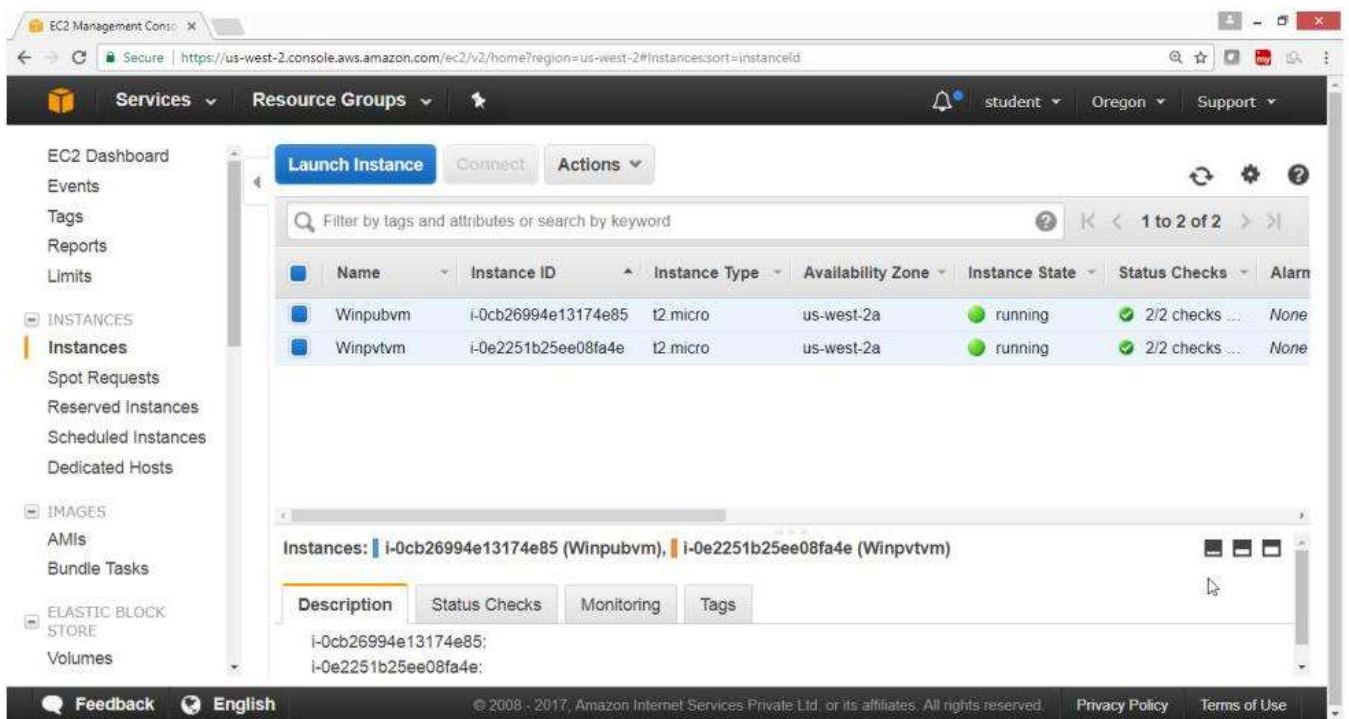


Verify that instance is Running

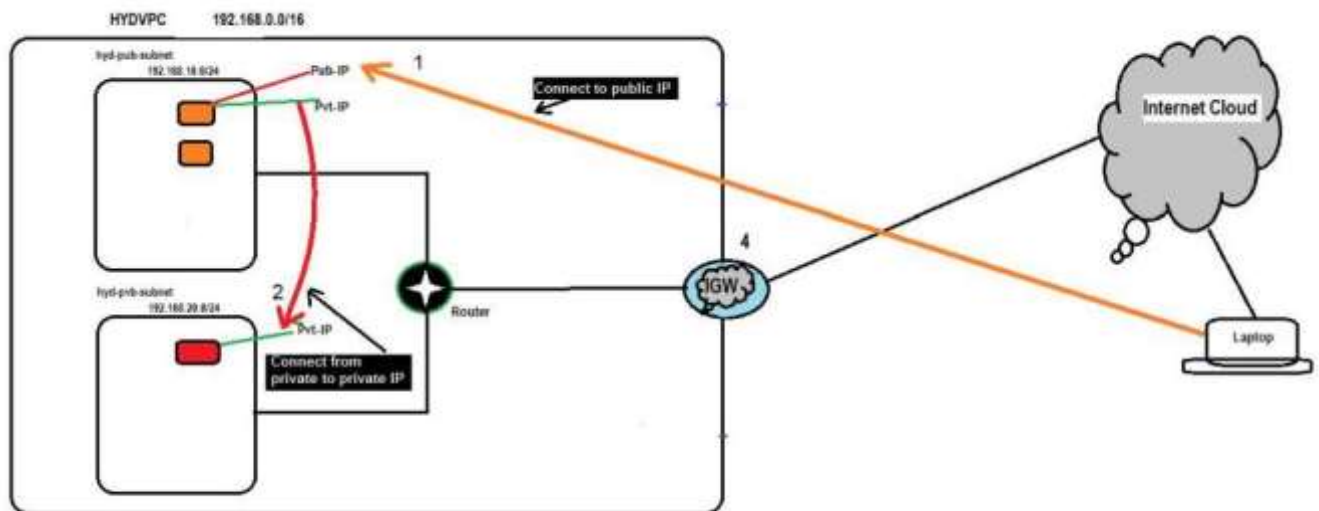


Verification

Output shows that both instance in public & private subnet are running



Now to connect an instance in private subnet first connect an instance in public network then from there connect to an instance in private subnet as shown in diagram



9) To connect to Public Subnet instance

First locate the public IP of a public instance

The screenshot shows the AWS Management Console for the 'us-west-2' region. The 'Instances' list shows two instances: 'Winpubvm' and 'Winprivvm'. The 'Winpubvm' instance is highlighted, and its details are shown below. The 'Public IP' is 54.202.132.130, which is circled in orange.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm
Winpubvm	i-0cb26994e13174e85	t2.micro	us-west-2a	running	2/2 checks ...	None
Winprivvm	i-0e2251b25ee08fa4e	t2.micro	us-west-2a	running	2/2 checks ...	None

Instance: **i-0cb26994e13174e85 (Winpubvm)** Public IP: 54.202.132.130

Description	Status Checks	Monitoring	Tags
Instance ID	i-0cb26994e13174e85	Public DNS (IPv4)	-
Instance state	running	IPv4 Public IP	54.202.132.130
Instance type	t2.micro	IPv6 IPs	-

Click on "Connect" button

The screenshot shows the AWS Management Console interface. On the left, the navigation pane includes 'EC2 Dashboard', 'Events', 'Tags', 'Reports', 'Limits', 'INSTANCES', 'Spot Requests', 'Reserved Instances', 'Scheduled Instances', 'Dedicated Hosts', 'IMAGES', 'AMIs', 'Bundle Tasks', 'ELASTIC BLOCK STORE', and 'Volumes'. The main content area displays a table of EC2 instances. The 'Connect' button is highlighted with a mouse cursor. Below the table, the details for the instance 'Winpubvm' (ID: i-0cb26994e13174e85) are shown, including its public IP address: 54.202.132.130.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm
Winpubvm	i-0cb26994e13174e85	t2.micro	us-west-2a	running	2/2 checks ...	None
Winpvtvm	i-0e2251b25ee08fa4e	t2.micro	us-west-2a	running	2/2 checks ...	None

Instance: i-0cb26994e13174e85 (Winpubvm) Public IP: 54.202.132.130

Public DNS (IPv4): -
IPv4 Public IP: 54.202.132.130
IPv6 IPs: -

Click on "Download Remote Desktop File"

Click on "Get Password"

The screenshot shows the 'Connect To Your Instance' dialog box. It provides instructions on how to connect to a Windows instance using a remote desktop client. The dialog includes a 'Download Remote Desktop File' button, which is highlighted with a mouse cursor. Below this, it lists the connection details: Public IP (54.202.132.130), User name (Administrator), and Password (with a 'Get Password' button). The dialog also mentions that directory credentials can be used if the instance is joined to a directory, and provides a link to the connection documentation. A 'Close' button is located at the bottom right of the dialog.

Connect To Your Instance

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

Download Remote Desktop File

When prompted, connect to your instance using the following details:

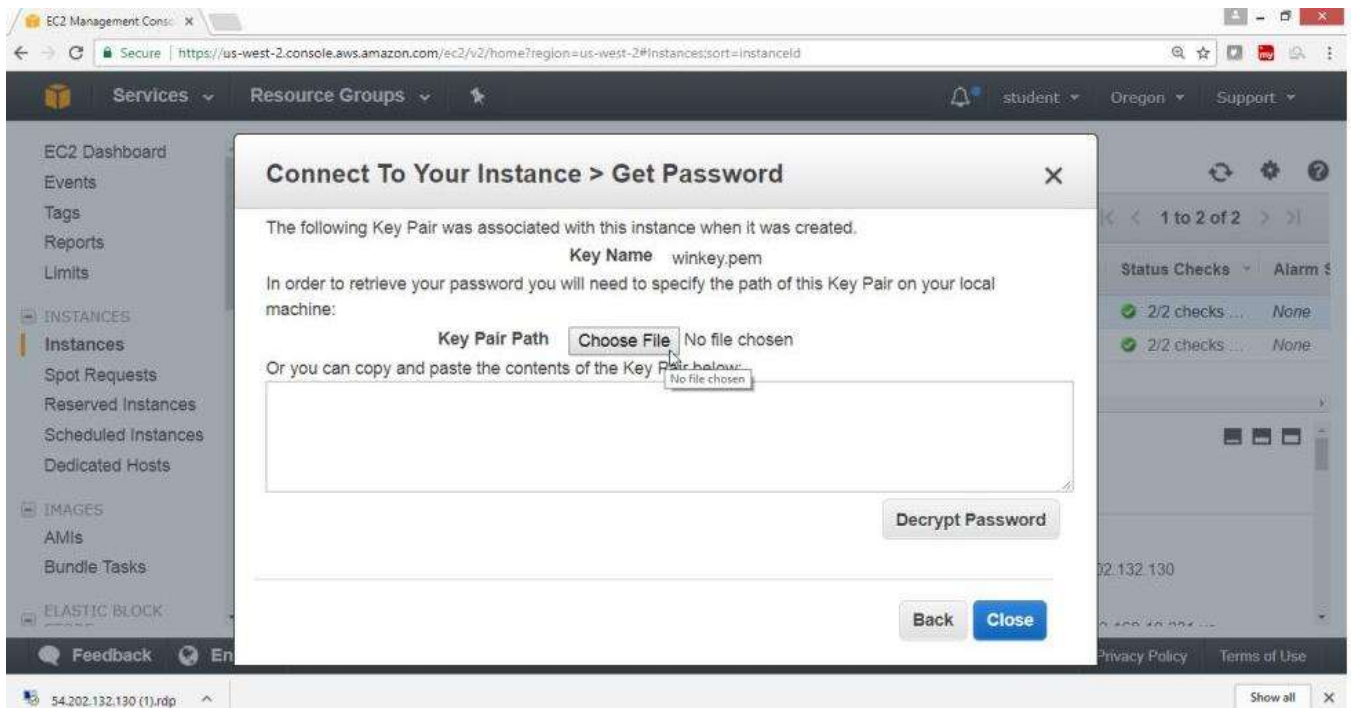
Public IP 54.202.132.130
User name Administrator
Password **Get Password**

If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.
If you need any assistance connecting to your instance, please see our [connection documentation](#).

Close

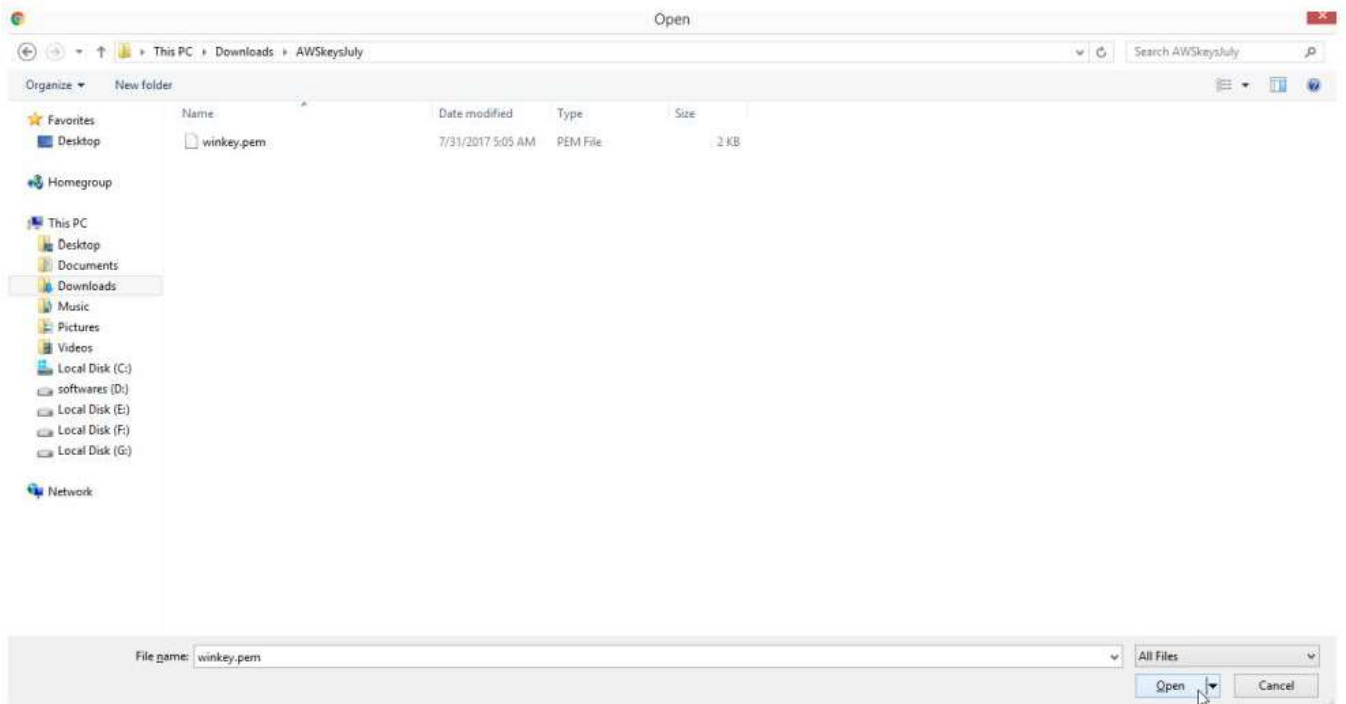
Provide the path of Key file

Click on "Choose file" button

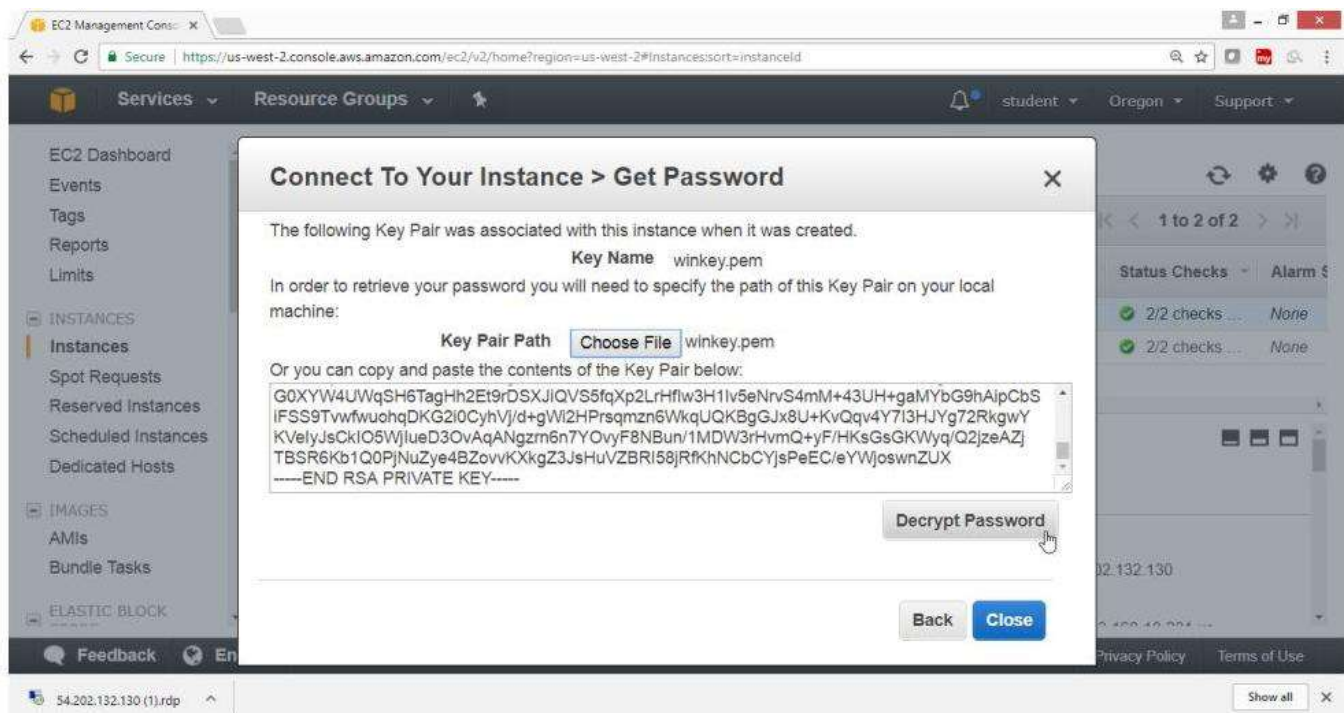


Select the key file

Click on "Open" button



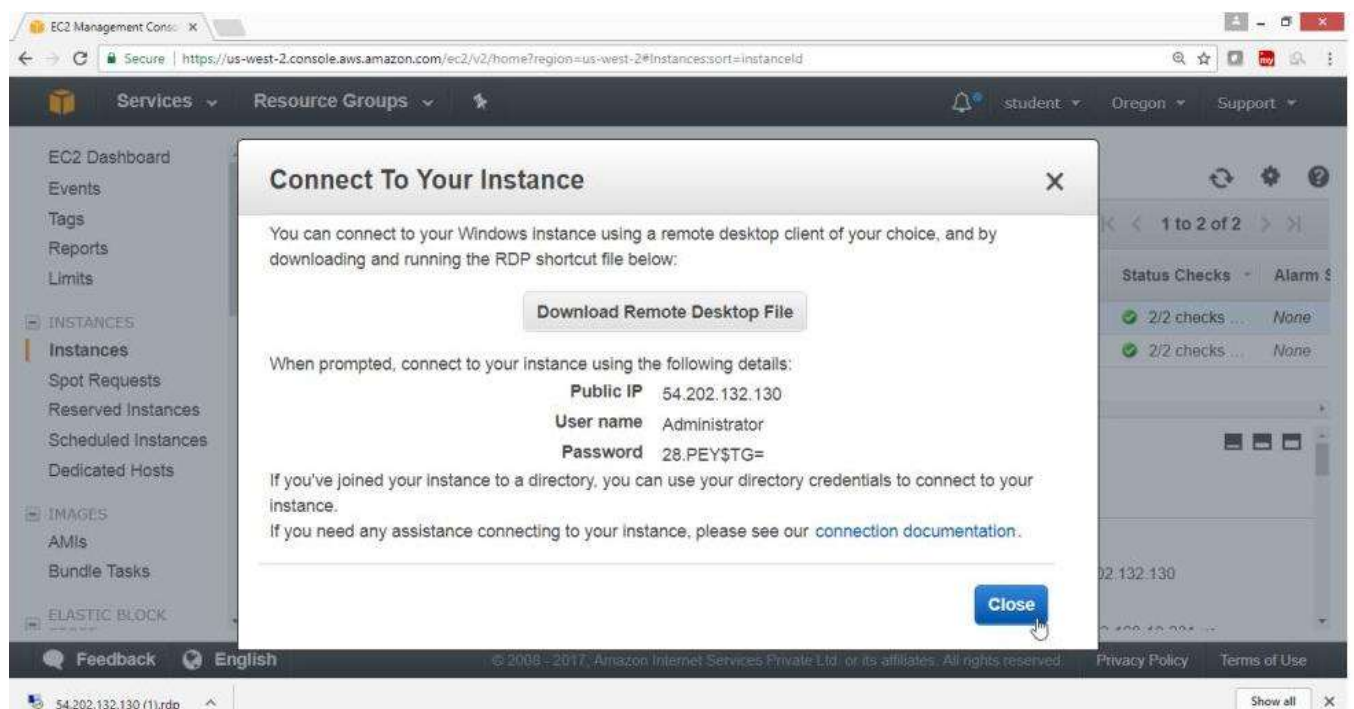
Now click on "Decrypt Password" button



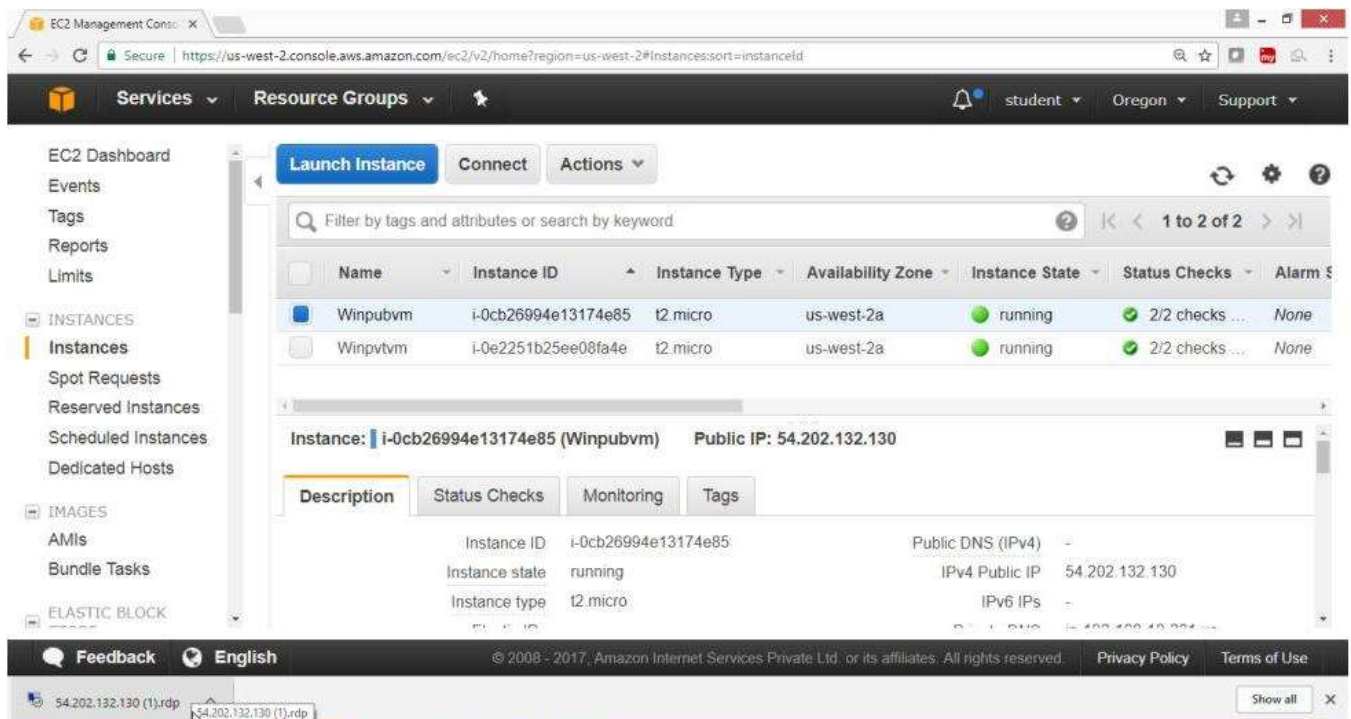
Verification

Password is generated copy in notepad

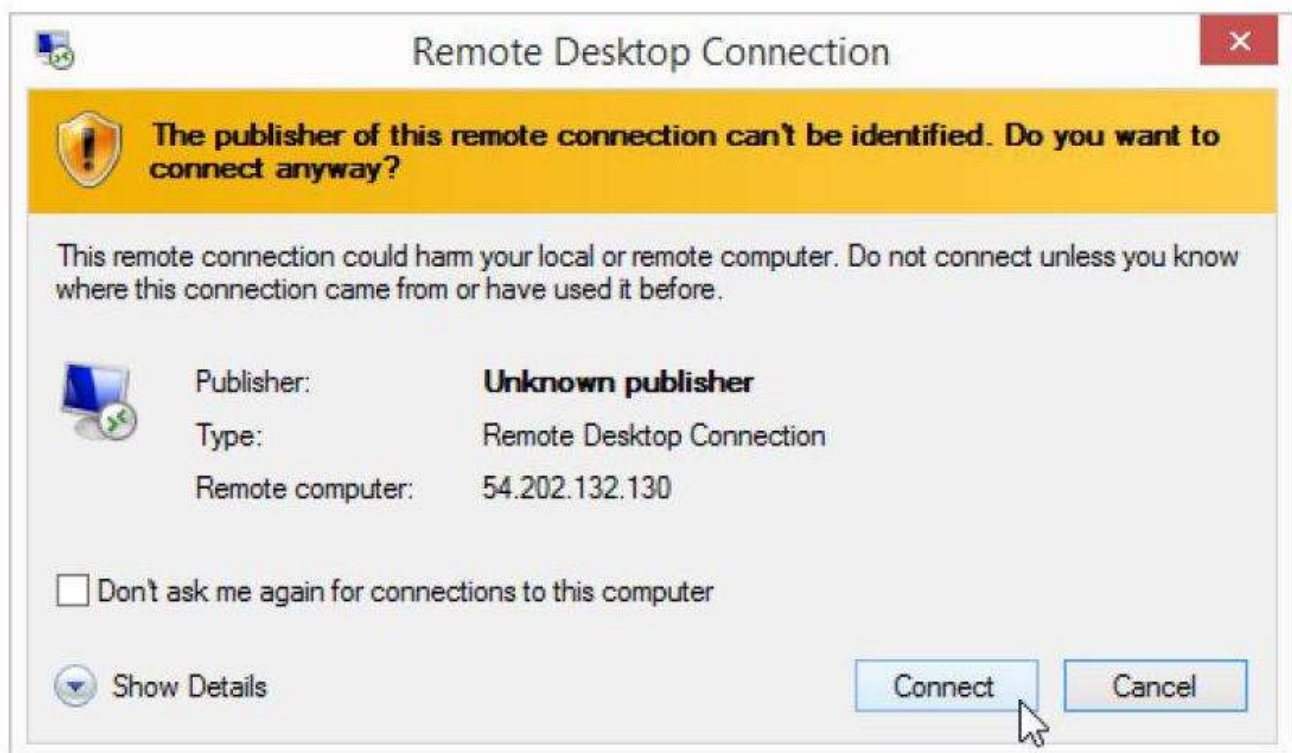
Click on "Close" button



Double Click on RDP file Provide Windows Username -> Administrator
Password-> "28.PEY\$TG=", as shown above



Click on "Connect" button



Paste the password
Click on "OK" button

Windows Security



Enter your credentials

These credentials will be used to connect to 54.202.132.130.



Administrator



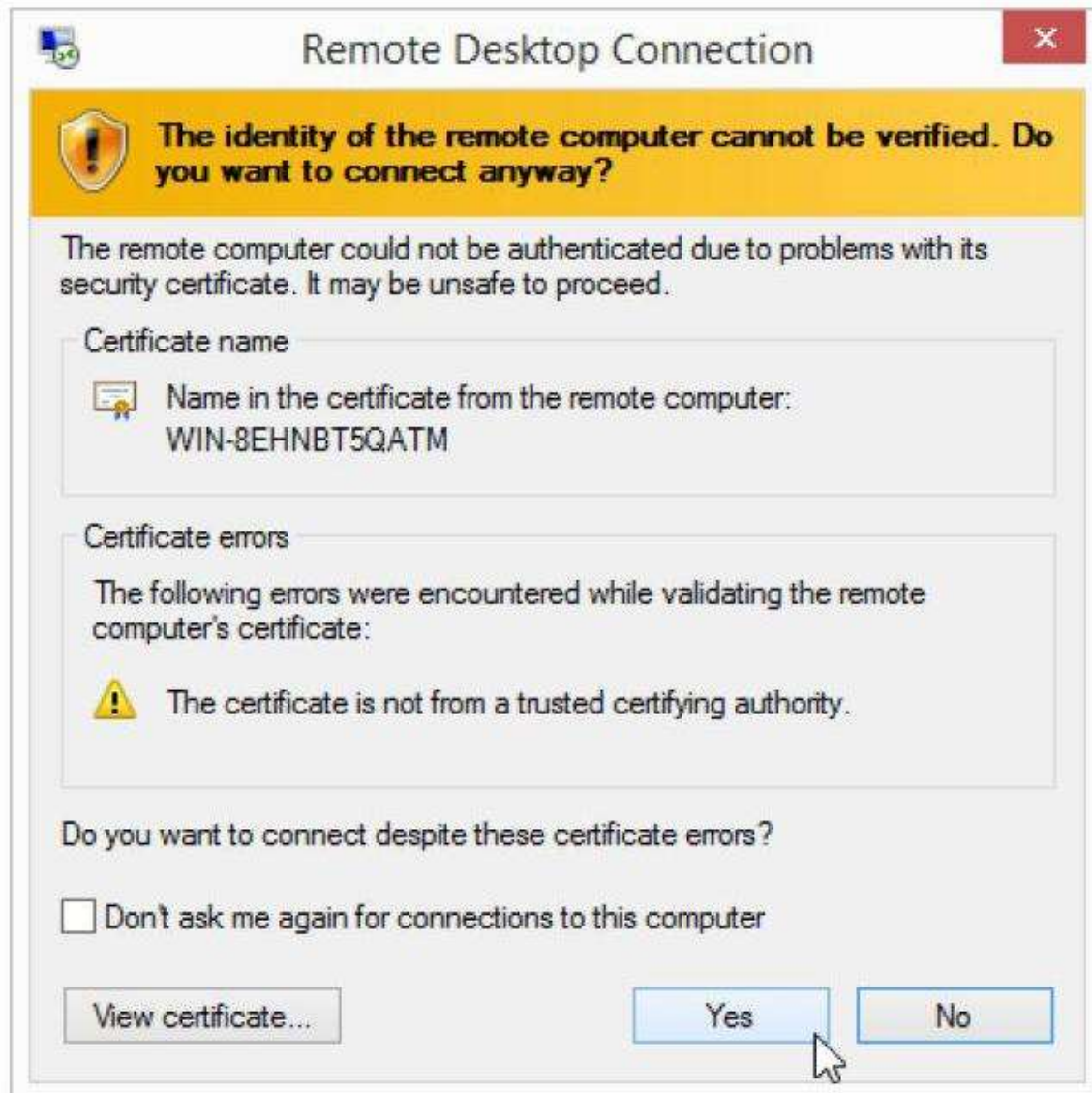
Use another account

☐ Remember my credentials

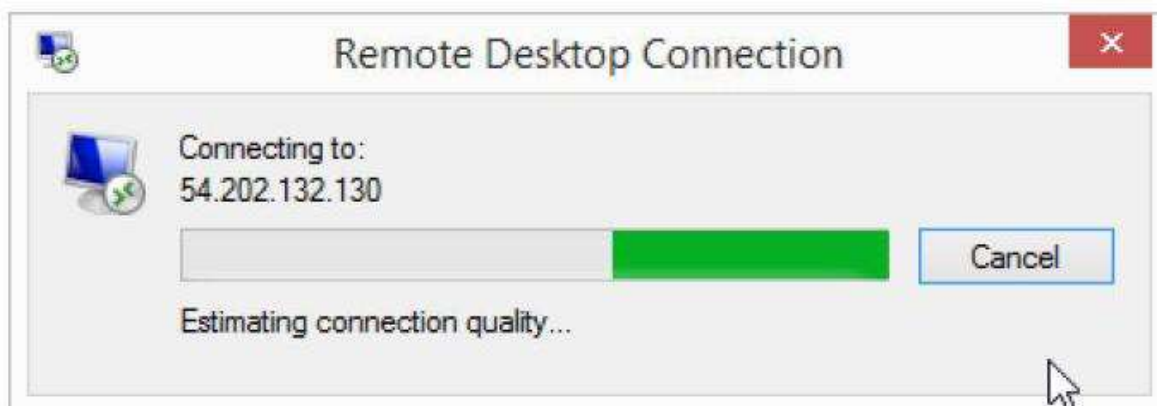
OK

Cancel

Click on Yes button



Verify



Verification

Now you are connected to windows Public Instance

On Windows Desktop public and private both IP's are displayed

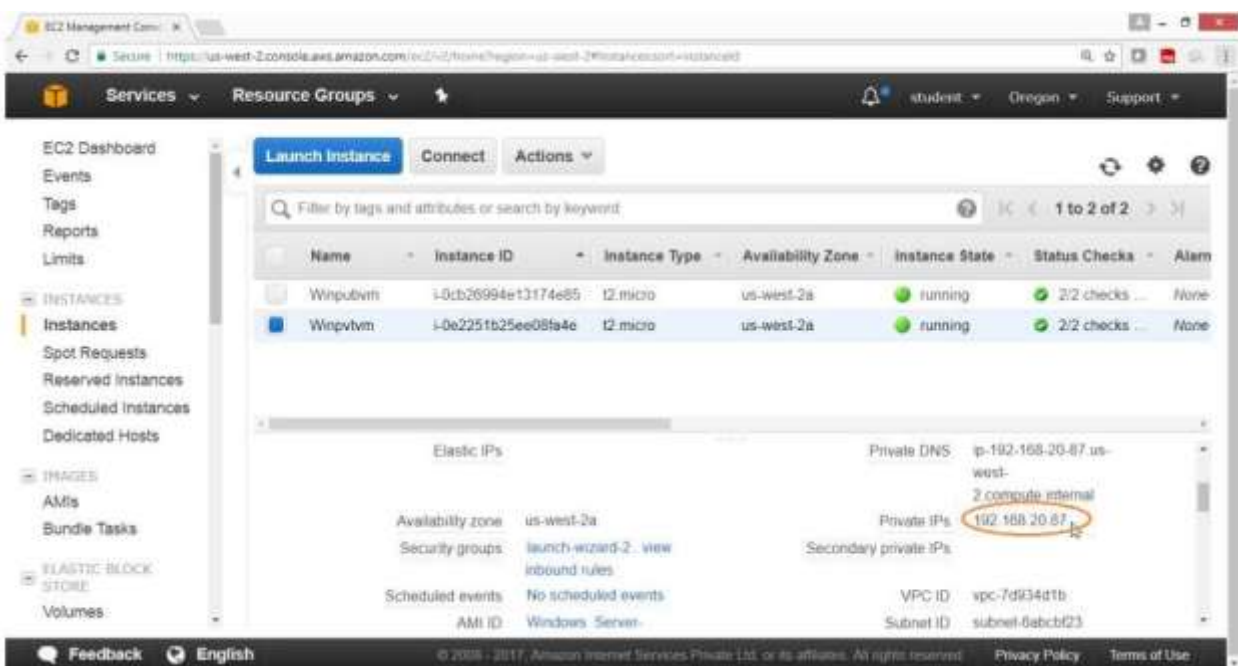


10) To Connect to Private subnet instance

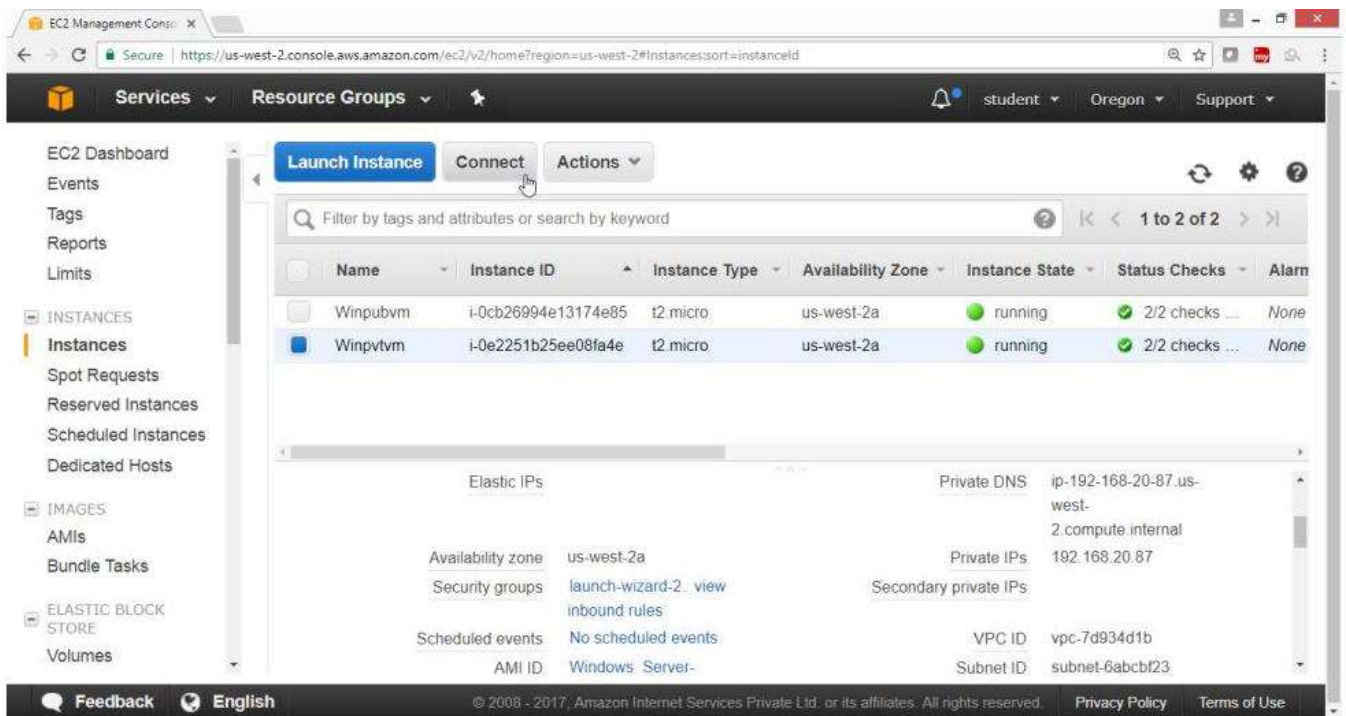
Go to EC2 Dashboard

Select private instance

Get the private IP of the instance



Click on Connect button



To get the password

Click on "Get Password" button

Connect To Your Instance

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

[Download Remote Desktop File](#)

When prompted, connect to your instance using the following details:

Private IP	192.168.20.87
User name	Administrator
Password	Get Password

If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.

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

[Close](#)

Click on "Decrypt Password"

Connect To Your Instance > Get Password



The following Key Pair was associated with this instance when it was created.

Key Name winkey.pem

In order to retrieve your password you will need to specify the path of this Key Pair on your local machine:

Key Pair Path No file chosen

Or you can copy and paste the contents of the Key Pair below:

```
-----BEGIN RSA PRIVATE KEY-----
MIIIEowIBAAKCAQEAsrhLs36UXn01ILHgG/mv0QHxJMq6p3NPPFedup5gUUYge2z8j8QQf1sn2AKs
Ye9PBAwBxMwlhdUPy0GbiRuBSI7CYOcTkdXjpuhTgG2Ylnkpxuql0BYkw3n9B3AMDmVbSyvsrenC
Lcg05A1sSSmOtTrBqUqkoANQZa+uZO7xDEkQS3G6rTft6XTtcjci5Wp4erJfMPneJYCdg7ui/Rm
TCdbD9m8h/ND5+nqajv80X3QsROGyTddRf29/M1VRh1/FXdl7NV+qK6n3te/lmP2ZP4OIH6uiFuY
-----
```

Verify, IP and password of private subnet instance is provided

Connect To Your Instance



You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

When prompted, connect to your instance using the following details:

Private IP 192.168.20.87

User name Administrator

Password G-oV;n\$.@i

If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.

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

Now logging to public instance

Open Run and type mstsc to connect to window private instance



Recycle Bin

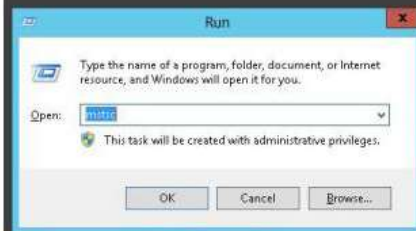


EC2
Feedback



EC2
Micros...

Hostname : WIN-BEHNB5QATM
Instance ID : i-0cb26994e13174e85
Public IP Address : 54.202.132.130
Private IP Address : 192.168.10.231
Availability Zone : us-west-2a
Instance Size : t2.micro
Architecture : AMD64



Windows Server 2012



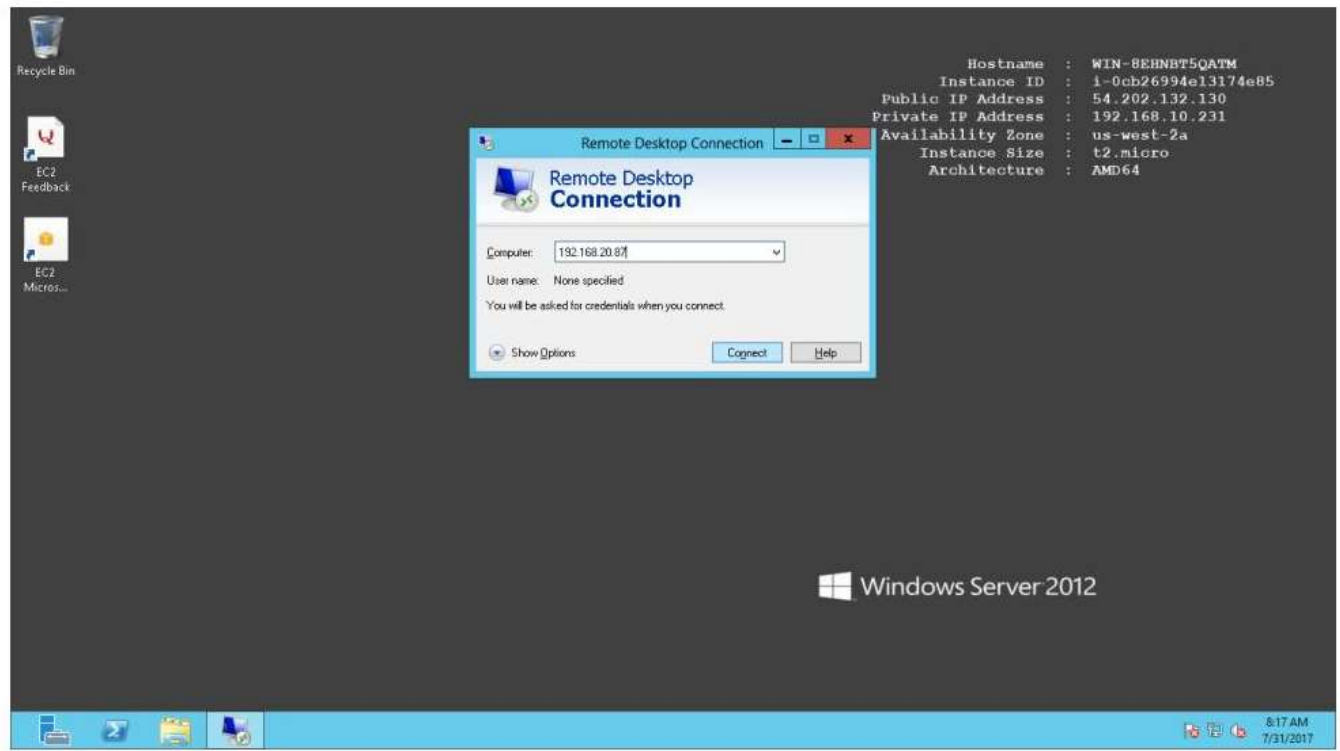
8:06 AM
7/31/2017

Provide Private instance

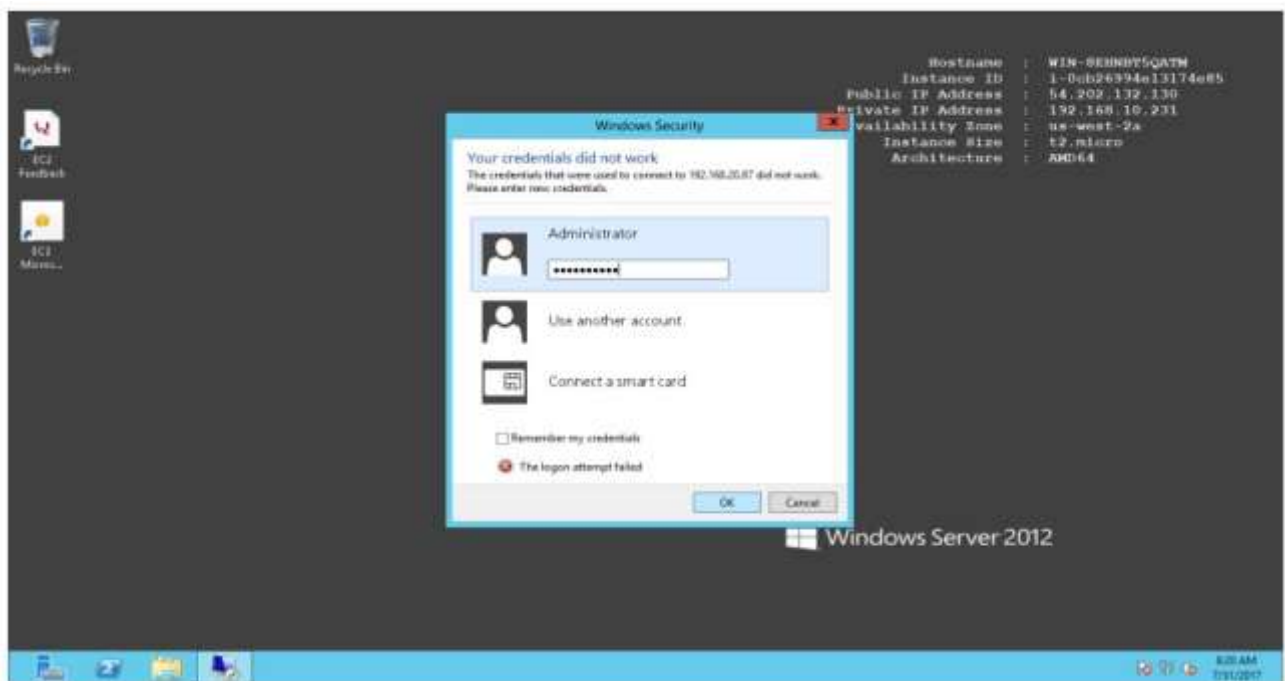
Private IP ->192.168.20.87

Username -> Administrator

Password ->G-oV;n\$@i;



Now provide Username & Password



Verification

Check private IP at Right top corner

Now you are connected to windows private instance



Recycle Bin



EC2
Feedback




EC2
Micros...

```
Hostname : WIN-V394191MM55
Instance ID : i-0e2251b25ee08fa4e
Private IP Address : 192.168.20.87
Availability Zone : us-west-2a
Instance Size : t2.micro
Architecture : AMD64
```

 Windows Server 2012 R2

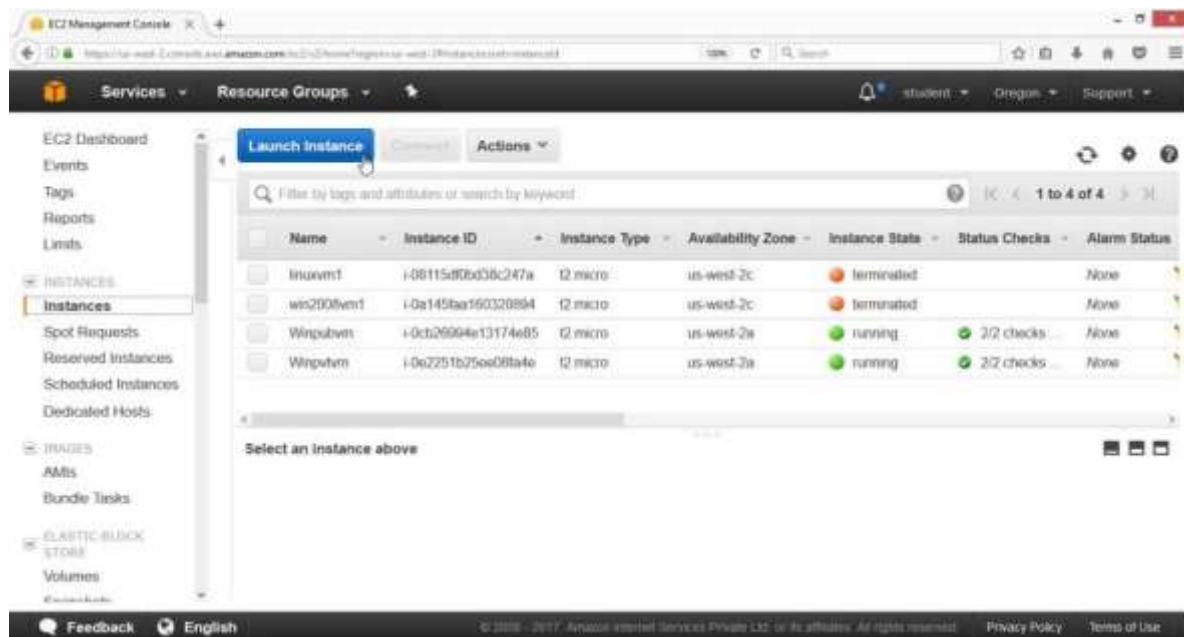


 8:21 AM
7/31/2017

11) To connect to Linux instance in private subnet

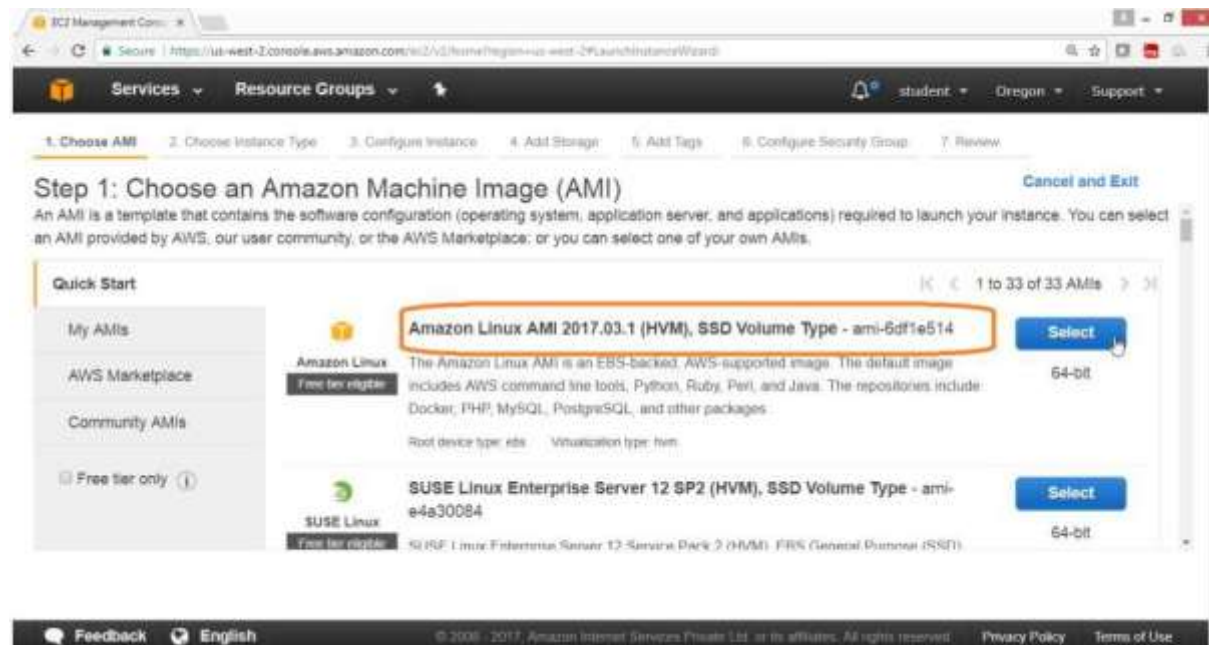
Launch Linux instance in public subnet -> hyd-pub-subnet

- Open the AWS console
- Click on Services
- Click on Instance
- Click on "Launch Instance" Button



On the "Choose an Amazon Machine Image (AMI)", page

- Select AMI "Amazon Linux AMI 2017.03.1 (HVM), SSD Volume Type - ami-6df1e514"
- Click on Select Button



On the "Choose an Instance Type"

- Select "General Purpose"
- Type->t2.micro
- Click on "Next: Configure Instance Details"

EC2 Management Console

Services Resource Groups

student Oregon Support

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

Step 2: Choose an Instance Type

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

Cancel Previous **Review and Launch** Next: Configure Instance Details

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

On the "Configure Instance Details" page

- Number of Instance ->1
- Network -> HYDVPC
- Subnet ->hyd-pub-subnet
- Auto-assign Public IP -> Enable

EC2 Management Console

Services Resource Groups

student Oregon 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 [Launch into Auto Scaling Group](#)

Purchasing option ☐ Request Spot instances

Network [Create new VPC](#)

Subnet [Create new subnet](#)
250 IP Addresses available

Auto-assign Public IP

IAM role [Create new IAM role](#)

Cancel Previous **Review and Launch** Next: Add Storage

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

On the "Add Storage" page

Leave the values as default

Click on "Next: Add Tags" button

EC2 Management Console

Secure | https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#LaunchInstanceWizard

Services

Resource Groups

student

Oregon

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/xvda	snap-0e8e196a52ed7efc3	8	General Purpose S	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

Cancel

Previous

Review and Launch

Next: Add Tags

Feedback

English

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

Privacy Policy

Terms of Use

On the "Add Tags" Page

Key->Name

Value->Linuxpubvm

Click on "Next: Configure Security Group" Button

The screenshot shows the AWS Management Console interface for the 'Add Tags' step in the EC2 Launch Wizard. The breadcrumb navigation at the top indicates the following steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags (current step), 6. Configure Security Group, and 7. Review. The main heading is 'Step 5: Add Tags'. Below this, a descriptive text states: '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.'

The tag configuration area contains a table with columns for 'Key' (127 characters maximum), 'Value' (255 characters maximum), 'Instances' (with an information icon), and 'Volumes' (with an information icon). A single tag is added with the key 'Name' and the value 'Linuxpubvm'. Checkmarks in the 'Instances' and 'Volumes' columns indicate the tag is applied to both. An 'Add another tag' button is present below the table, with a note '(Up to 50 tags maximum)'. At the bottom of the wizard, there are four buttons: 'Cancel', 'Previous', 'Review and Launch' (highlighted in blue), and 'Next: Configure Security Group' (which is the button being pointed to by a mouse cursor).

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

On the "Configure Security Group" page

Assign a security group -> Create a new security group

Leave remaining values as default

Click on "Review and Launch" Button

The screenshot shows the 'Step 6: Configure Security Group' page in the AWS Management Console. The breadcrumb trail at the top indicates the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The page title is 'Step 6: Configure Security Group'. Below the title, there is a description of a security group and a link to 'Learn more about Amazon EC2 security groups'. The 'Assign a security group' section has two radio buttons: 'Create a new security group' (selected) and 'Select an existing security group'. Below this, there are input fields for 'Security group name' (filled with 'launch-wizard-5') and 'Description' (filled with 'launch-wizard-5 created 2017-08-01T13:31:54.220+05:30'). A table with columns 'Type', 'Protocol', 'Port Range', and 'Source' contains one rule: 'SSH' (Type), 'TCP' (Protocol), '22' (Port Range), and 'Anywhere' (Source). Below the table is an 'Add Rule' button. At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Review and Launch' (highlighted with a mouse cursor).

On the "Review and Launch", page

Click on Launch Button

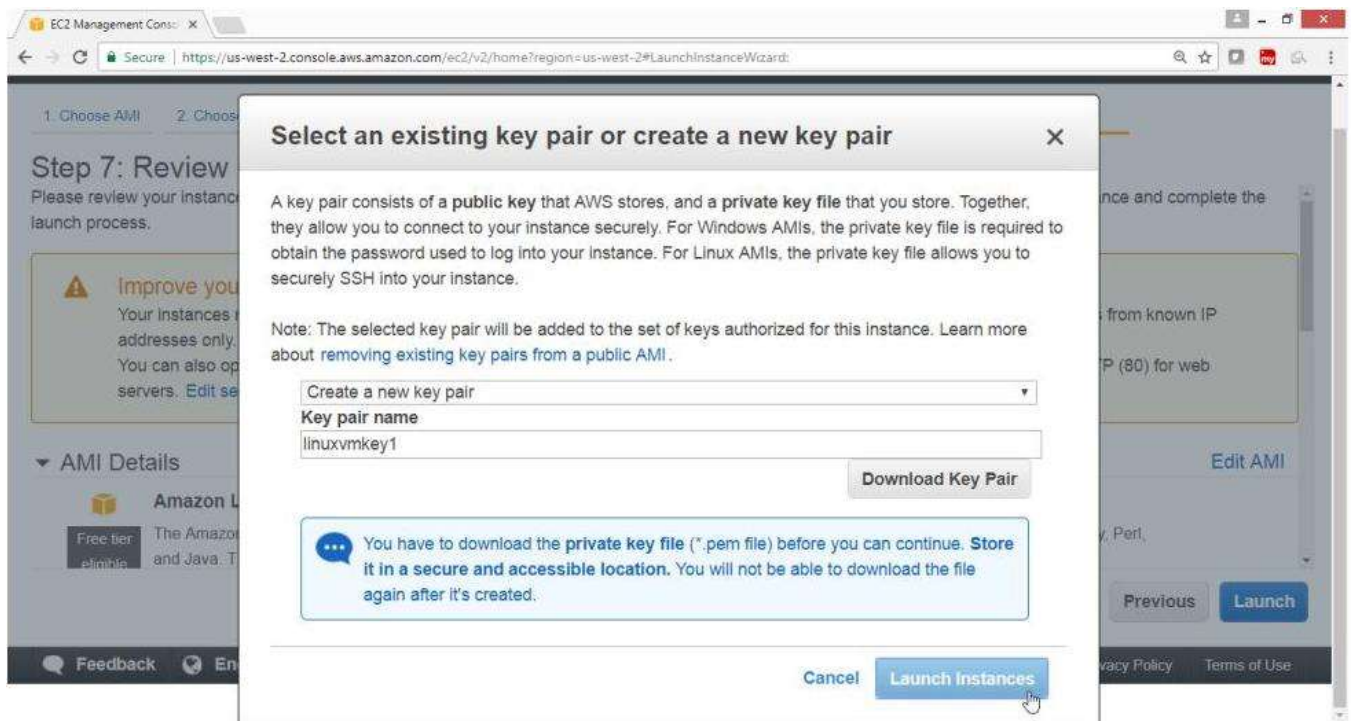
The screenshot shows the 'Step 7: Review Instance Launch' page in the AWS Management Console. The breadcrumb trail at the top indicates the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The page title is 'Step 7: Review Instance Launch'. Below the title, there is a description of the review process. A yellow warning box contains a message: 'Improve your instances' security. Your security group, launch-wizard-5, 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'. Below the warning box, there is a section for 'AMI Details' with a dropdown arrow and an 'Edit AMI' link. The AMI is 'Amazon Linux AMI 2017.03.1 (HVM), SSD Volume Type - ami-6df1e514'. Below this, there is a 'Free tier eligible' badge and a description of the AMI. At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Launch' (highlighted with a mouse cursor).

On the "Select an existing key pair or create a new key pair" page

Select "Create a new key pair"

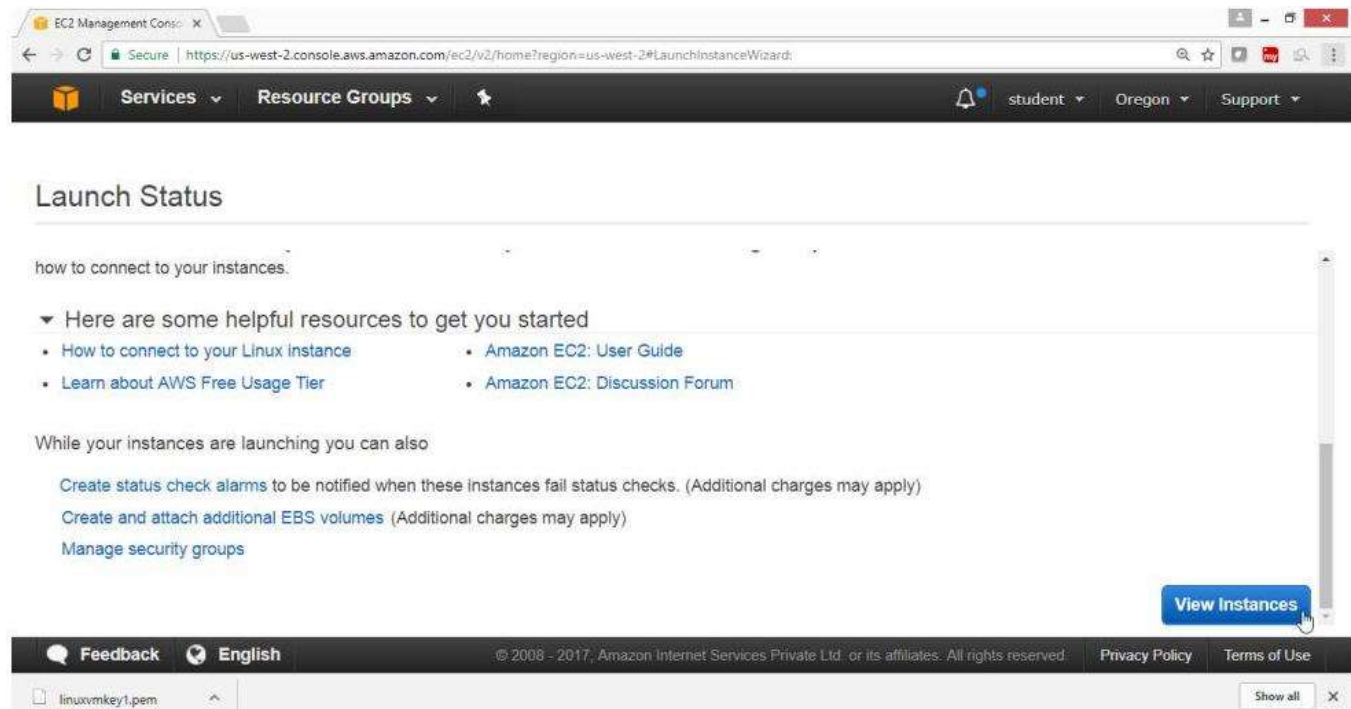
Key pair name->linuxvmkey1

Click on "Launch Instances" Button



Check the summary

Click on "View Instance" Button



Verification

Linux Instance in public subnet is launched

EC2 Management Console

Secure | https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#Instances:sort=tag:Name

Services | Resource Groups | student | Oregon | Support

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Spot Requests

Reserved Instances

Scheduled Instances

Dedicated Hosts

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK

Launch Instance | Connect | Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm S
Linuxpubvm	i-0c53f560c48fd5f80	t2.micro	us-west-2a	running	Initializing	None
linuxvm1	i-08115df0bd38c247a	t2.micro	us-west-2c	terminated		None
Winpubvm	i-0cb26994e13174e85	t2.micro	us-west-2a	running	2/2 checks ...	None

Instance: i-0c53f560c48fd5f80 (Linuxpubvm) | Public IP: 54.202.241.190

Description

Status Checks

Monitoring

Tags

Instance ID

i-0c53f560c48fd5f80

Public DNS (IPv4)

Instance state

running

IPv4 Public IP

54.202.241.190

Instance type

t2.micro

IPv6 IPs

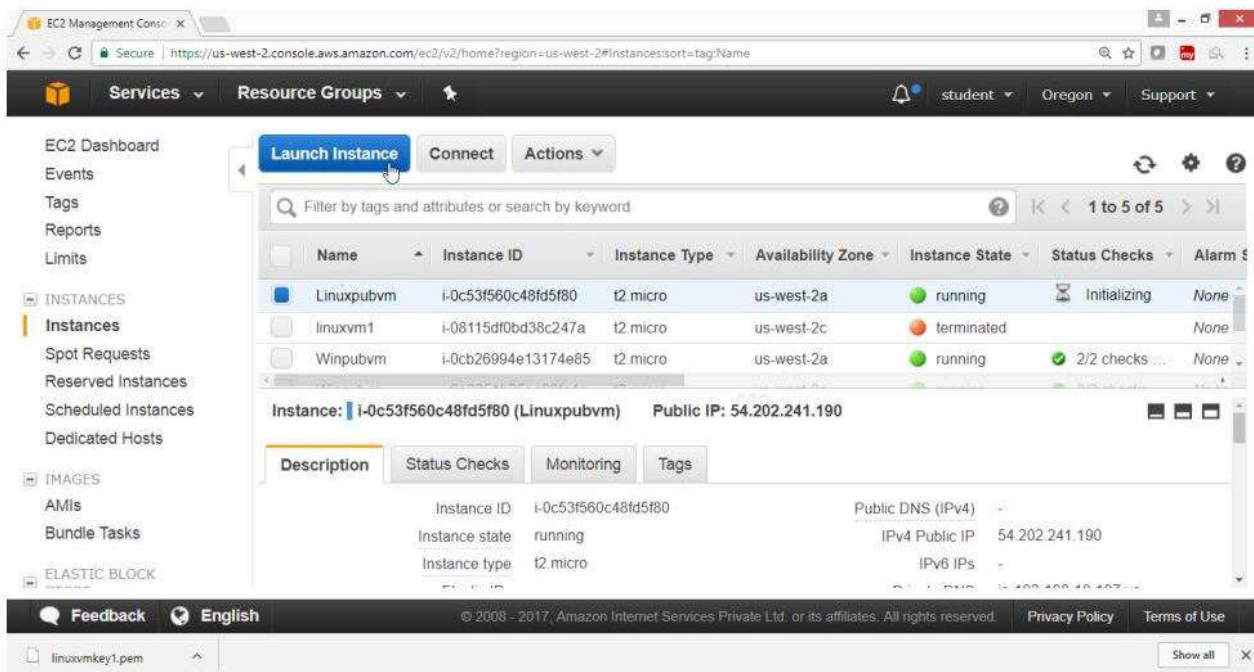
Feedback | English | © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

linuxvmkey1.pem | Show all

12) To connect to Linux instance in private subnet

Launch Linux instance in public subnet ->hyd-pvt-subnet

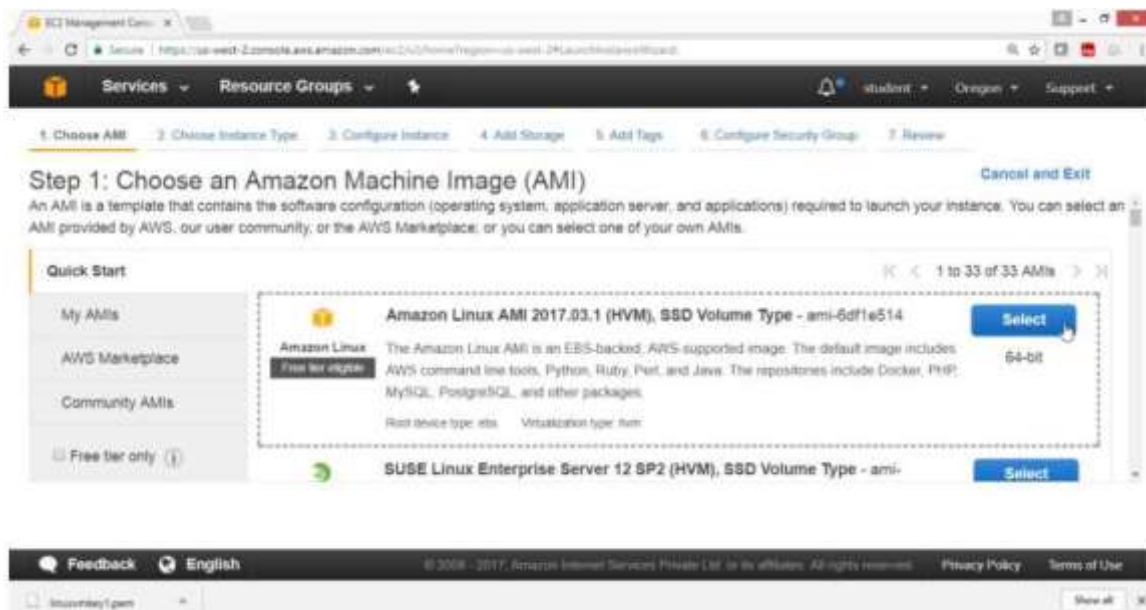
- Open the AWS console
- Click on Services
- Click on Instance
- Click on "Launch Instance" Button



On the "Choose an Amazon Machine Image (AMI)", page

Select AMI "Amazon Linux AMI 2017.03.1(HVM), SSD Volume Type -ami-6df1e514

Click on Select Button



On the "Choose an Instance Type"


Select "General Purpose"

Type->t2.micro

Click on "Next: Configure Instance Details"

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

Step 2: Choose an Instance Type

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

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Instance Details](#)

On the "Configure Instance Details" page

Number of Instance ->1

Network -> HYDVPC

Subnet ->hyd-pvt-subnet

Auto-assign Public IP -> Disable

The screenshot shows the AWS Management Console interface for the 'Configure Instance Details' step of the EC2 Launch Wizard. The breadcrumb navigation at the top indicates the steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance (current step), 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The page title is 'Step 3: Configure Instance Details'. Below the title, a descriptive text states: '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.'

The configuration fields are as follows:

- Number of instances:** A text input field containing the value '1'. To its right is a link 'Launch into Auto Scaling Group' with an information icon.
- Purchasing option:** A dropdown menu with 'Request Spot instances' selected.
- Network:** A dropdown menu showing 'vpc-7d934d1b | HYDVPC'. To its right is a link 'Create new VPC' with a refresh icon.
- Subnet:** A dropdown menu showing 'subnet-6abcbf23 | hyd-pvt-subnet | us-west-2a'. Below the dropdown, it says '250 IP Addresses available'. To its right is a link 'Create new subnet'.
- Auto-assign Public IP:** A dropdown menu with 'Disable' selected.

At the bottom of the configuration section, there are four buttons: 'Cancel', 'Previous', 'Review and Launch' (highlighted in blue), and 'Next: Add Storage'. A mouse cursor is pointing at the 'Next: Add Storage' button.

The footer of the console includes a 'Feedback' link, the language 'English', copyright information '© 2006 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.', links for 'Privacy Policy' and 'Terms of Use', and a 'Show all' link next to a list of items including 'linuxvmkey1.pem'.

On the "Add Storage" page
Leave the values as default
Click on "Next: Add Tags" button

The screenshot shows the 'Step 4: Add Storage' page in the AWS Management Console. The breadcrumb trail at the top indicates the sequence: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage (current), 5. Add Tags, 6. Configure Security Group, 7. Review. The main heading is 'Step 4: Add Storage', followed by a paragraph explaining storage options. Below this is a table with columns: Volume Type, Device, Snapshot, Size (GiB), Volume Type, IOPS, Throughput (MB/s), Delete on Termination, and Encrypted. The first row shows the root volume with a size of 8 GiB, General Purpose volume type, 100 / 3000 IOPS, N/A throughput, and 'Not Encrypted'. An 'Add New Volume' button is below the table. At the bottom right, there are buttons for 'Cancel', 'Previous', 'Review and Launch', and 'Next: Add Tags'. The footer includes 'Feedback', 'English', copyright information, and links to 'Privacy Policy' and 'Terms of Use'.

Click on "Add Tags" Button

The screenshot shows the 'Step 5: Add Tags' page in the AWS Management Console. The breadcrumb trail indicates: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags (current), 6. Configure Security Group, 7. Review. The heading is 'Step 5: Add Tags', followed by a paragraph explaining tagging. Below this is a form with columns for 'Key' (127 characters maximum), 'Value' (255 characters maximum), 'Instances', and 'Volumes'. A message states 'This resource currently has no tags' and provides instructions to 'Choose the Add tag button or click to add a Name tag'. An 'Add Tag' button is present, with a note '(Up to 50 tags maximum)'. At the bottom right, there are buttons for 'Cancel', 'Previous', 'Review and Launch', and 'Next: Configure Security Group'. The footer is identical to the previous screenshot.

On the "Add Tags" Page
Key->Name
Value->Linuxpvtvm
Click on "Next: Configure Security Group" Button

EC2 Management Console

Secure | https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#LaunchInstanceWizard:

ServicesResource Groups

studentOregonSupport

1. Choose AMI2. Choose Instance Type3. Configure Instance4. Add Storage5. Add Tags6. Configure Security Group7. 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	Linuxpvtvm	<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

FeedbackEnglish

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

Privacy PolicyTerms of Use

On the "Configure Security Group" page

Assign a security group -> Create a new security group

Leave remaining values as default

Click on "Review and Launch" Button

The screenshot shows the 'Configure Security Group' step (Step 6) of the AWS Launch Wizard. The page title is 'Step 6: Configure Security Group'. Below the title, there is a description of a security group and instructions on how to add rules. The 'Assign a security group' section has two radio buttons: 'Create a new security group' (selected) and 'Select an existing security group'. Below this, the 'Security group name' is 'launch-wizard-6' and the 'Description' is 'launch-wizard-6 created 2017-08-01T13:51:38.571+05:30'. A table for adding rules is visible with columns: Type, Protocol, Port Range, and Source. One rule is added: Type 'SSH', Protocol 'TCP', Port Range '22', and Source 'Anywhere' with IP range '0.0.0.0/0'. At the bottom right, there are buttons for 'Cancel', 'Previous', and 'Review and Launch'.

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

Description:

Type	Protocol	Port Range	Source
SSH	TCP	22	Anywhere 0.0.0.0/0 ::/0

[Add Rule](#)

[Cancel](#) [Previous](#) [Review and Launch](#)

On the "Review and Launch", page

Click on Launch Button

The screenshot shows the 'Review Instance Launch' step (Step 7) of the AWS Launch Wizard. The page title is 'Step 7: Review Instance Launch'. It displays the security group details: 'launch-wizard-6' with the same description as in the previous step. Below this is a table with the same rule as in the previous step. At the bottom, there are expandable sections for 'Instance Details', 'Storage', and 'Tags', each with an 'Edit' link. At the bottom right, there are buttons for 'Cancel', 'Previous', and 'Launch'.

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

Step 7: Review Instance Launch

Security group name: launch-wizard-6

Description: launch-wizard-6 created 2017-08-01T13:51:38.571+05:30

Type	Protocol	Port Range	Source
SSH	TCP	22	0.0.0.0/0
SSH	TCP	22	::/0

[Edit instance details](#)

[Edit storage](#)

[Edit tags](#)

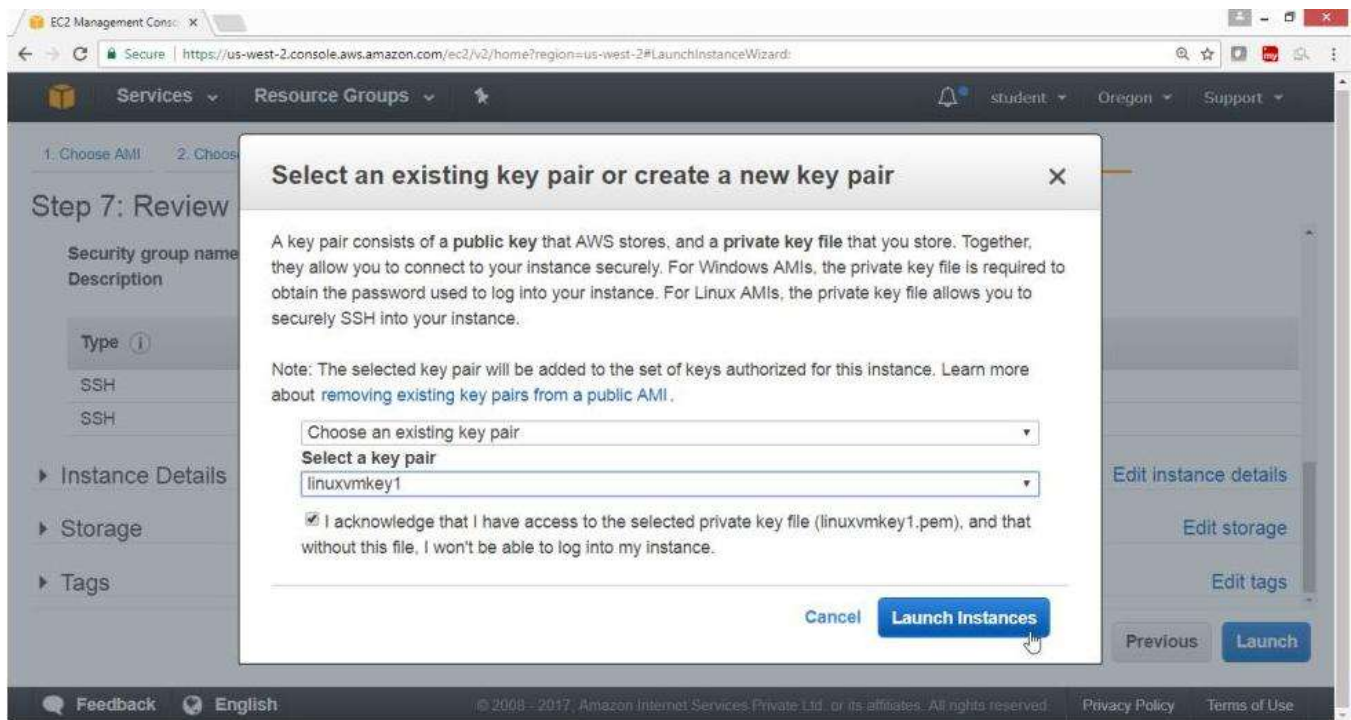
[Cancel](#) [Previous](#) [Launch](#)

On the "Select an existing key pair or create a new key pair" page

Select "Create a new key pair"

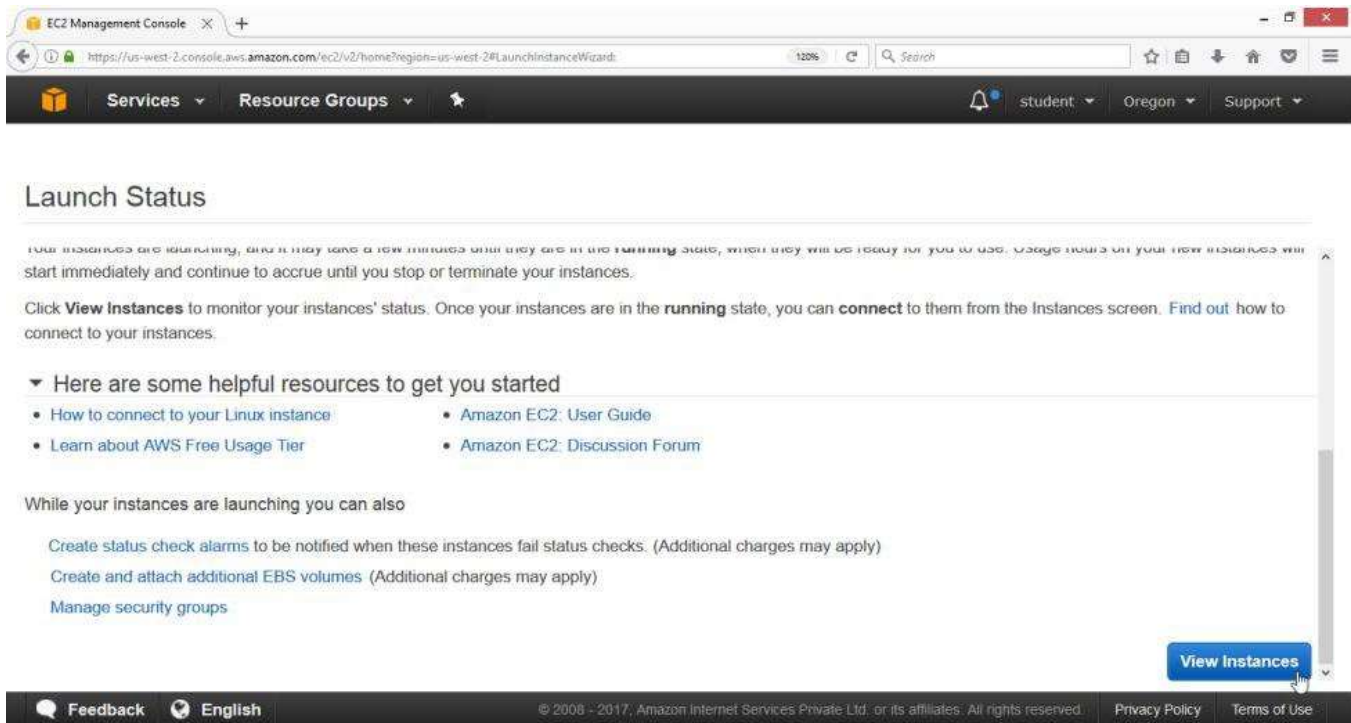
Key pair name->linuxvmkey1

Click on "Launch Instances" Button



Check the summary

Click on "View Instance" Button



Verification

Linux Instance in public subnet is launched

EC2 Management Console

https://us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#instances:sort=desc:tag:Name

133%

Search

☆

📄

⬇️

🏠

🔔

☰

Services

Resource Groups

🌟

🔔

student

Oregon

Support

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Spot Requests

Reserved Instances

Scheduled Instances

Dedicated Hosts

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

1 to 4 of 4

<input type="checkbox"/>	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks
<input type="checkbox"/>	Winpvtvm	i-0e2251b25ee08fa4e	t2.micro	us-west-2a	running	2/2 checks ...
<input type="checkbox"/>	Winpubvm	i-0cb26994e13174e85	t2.micro	us-west-2a	running	2/2 checks ...
<input checked="" type="checkbox"/>	Linuxpvtvm	i-0da6594c71079c242	t2.micro	us-west-2a	running	Initializing
<input type="checkbox"/>	Linuxpubvm	i-0c53f560c48fd5f80	t2.micro	us-west-2a	running	2/2 checks ...

Instance: i-0da6594c71079c242 (Linuxpvtvm)

Private IP: 192.168.20.101

Description

Status Checks

Monitoring

Tags

Instance ID

i-0da6594c71079c242

Public DNS (IPv4)

-

Instance state

running

IPv4 Public IP

-

Feedback

English

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

Privacy Policy

Terms of Use

To connect to Linux private instance

- First copy the key to Linux in public subnet
 - Now connect to Linux instance in public
 - Then connect to Linux instance in private
-
- Open MobaXterm
 - Coping *.pem file to Linux instance in public
-
- Select public Linux instance click on connect

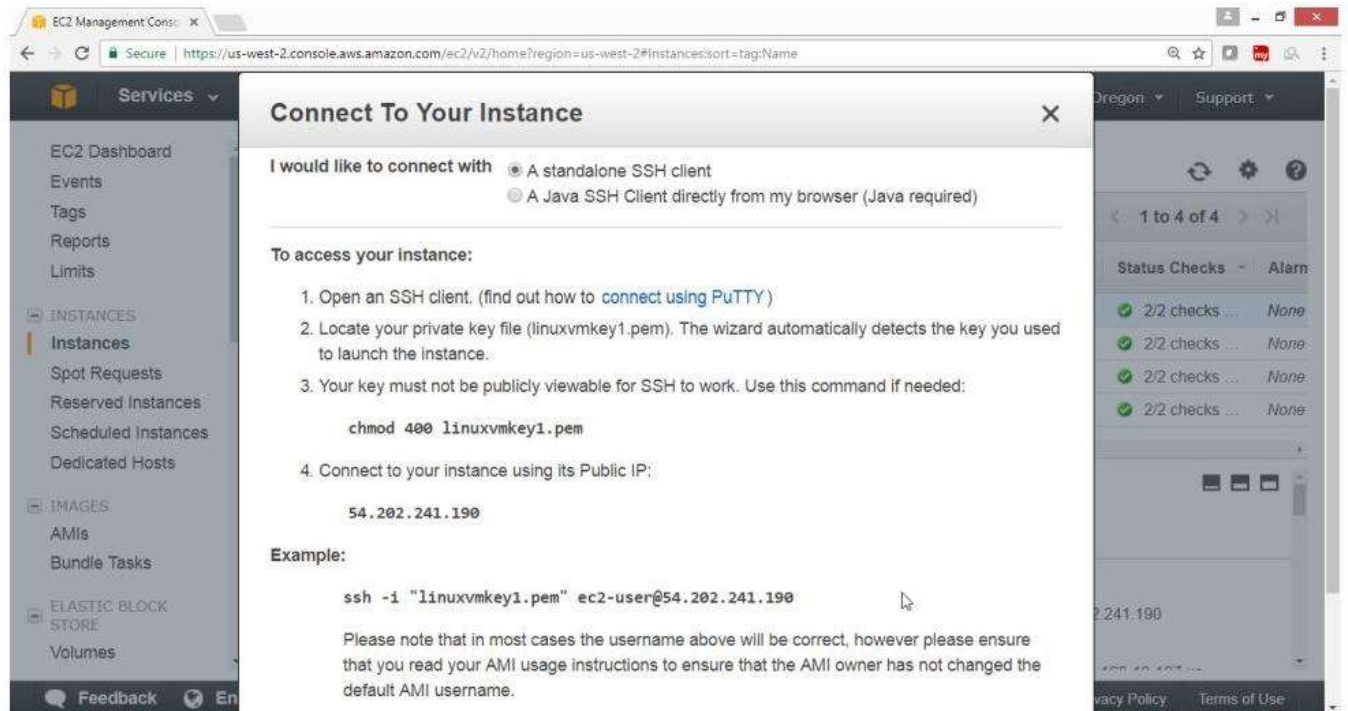
The screenshot shows the AWS Management Console for the EC2 service. The left sidebar contains navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES, IMAGES, ELASTIC BLOCK STORE, and Volumes. The main content area displays a table of instances. The instance 'Linuxpubvm' is selected, and its details are shown below the table. The public IP address is highlighted as 54.202.241.190.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm
Linuxpubvm	i-0c53f560c48fd5f80	t2.micro	us-west-2a	running	2/2 checks ...	None
Linuxprivvm	i-0da6594c71079c242	t2.micro	us-west-2a	running	2/2 checks ...	None
Winpubvm	i-0cb26994e13174e85	t2.micro	us-west-2a	running	2/2 checks ...	None
Winprivvm	i-0e2251b25ee08fa4e	t2.micro	us-west-2a	running	2/2 checks ...	None

Instance: **i-0c53f560c48fd5f80 (Linuxpubvm)** Public IP: 54.202.241.190

Description	
Instance ID	i-0c53f560c48fd5f80
Instance state	running
Instance type	t2.micro
Public DNS (IPv4)	-
IPv4 Public IP	54.202.241.190
IPv6 IPs	-

View the guide lines



Use the above public ip of Linux instance in mobaxterm

Copy *.pem file to pun Linux instance using scp command

```
[2017-08-01 14:21.18] /drives/e/awskeys
[shaikh.pc_mas] > ls
doom.mp3      linuxvmkey1.pem  putty.exe      puttygen.exe   winkey.pem

[2017-08-01 14:21.20] /drives/e/awskeys
[shaikh.pc_mas] > scp -i "linuxvmkey1.pem" linuxvmkey1.pem ec2-user@54.202.241.190:/home/ec2-user
linuxvmkey1.pem                                100% 1692    1.7KB/s   00:00

[2017-08-01 14:21.50] /drives/e/awskeys
[shaikh.pc_mas] > |
```

Verify

Use Commands, pwd, ls to check *.pem file

```
2. /drives/e/awskeys
[2017-08-01 14:22:27] /drives/e/awskeys
[shaikh.pc_mas] > pwd
/drives/e/awskeys

[2017-08-01 14:22:29] /drives/e/awskeys
[shaikh.pc_mas] > ls
doom.mp3      linuxvmkey1.pem  putty.exe      puttygen.exe    winkey.pem

[2017-08-01 14:22:30] /drives/e/awskeys
[shaikh.pc_mas] > 
```

Now connect to public instance using ssh command

```
2. ec2-user@ip-192-168-10-197:~
[2017-08-01 14:22:43] /drives/e/awskeys
[shaikh.pc_mas] > ssh -i "linuxvmkey1.pem" ec2-user@54.202.241.190
X11 forwarding request failed on channel 0
Last login: Tue Aug 1 08:50:19 2017 from 183.82.211.216

 _ _ | _ _ | _ _ |
|_| ( _ _ | _ _ |
|_| \ _ _ | _ _ |

Amazon Linux AMI

https://aws.amazon.com/amazon-linux-ami/2017.03-release-notes/
1 package(s) needed for security, out of 3 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-192-168-10-197 ~]$ 
```

Select private instance and get private ip

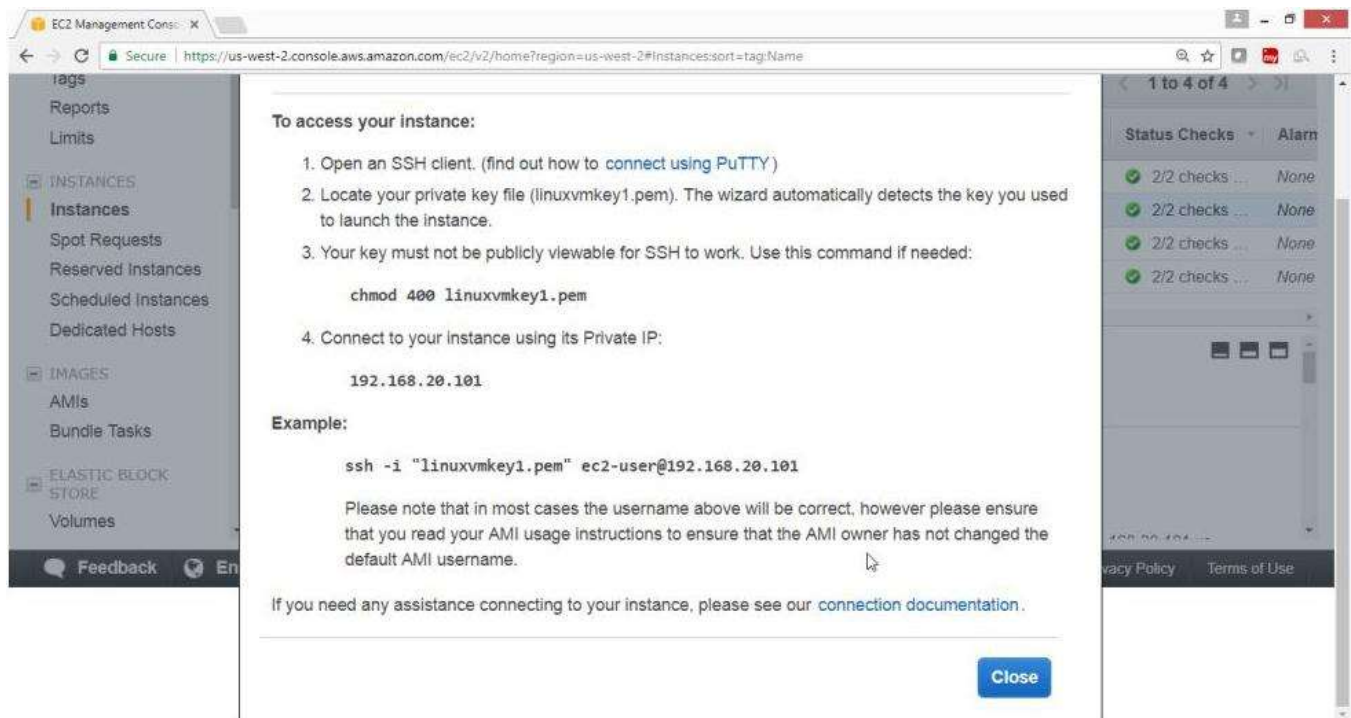
The screenshot shows the AWS Management Console interface. On the left, there is a navigation menu with options like EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES, Images, and Elastic Block Store. The main area displays a table of EC2 instances. The instance 'Linuxpvtvm' is selected, and its details are shown below the table. The private IP address is 192.168.20.101.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm
Linuxpubvm	i-0c53f560c48fd5f80	t2.micro	us-west-2a	running	2/2 checks ...	None
Linuxpvtvm	i-0da6594c71079c242	t2.micro	us-west-2a	running	2/2 checks ...	None
Winpubvm	i-0cb26994e13174e85	t2.micro	us-west-2a	running	2/2 checks ...	None
Winpvtvm	i-0e2251b25ee08fa4e	t2.micro	us-west-2a	running	2/2 checks ...	None

Instance: **i-0da6594c71079c242 (Linuxpvtvm)** Private IP: 192.168.20.101

Description	Status Checks	Monitoring	Tags
Instance ID	i-0da6594c71079c242	Public DNS (IPv4)	
Instance state	running	IPv4 Public IP	
Instance type	t2.micro	IPv6 IPs	

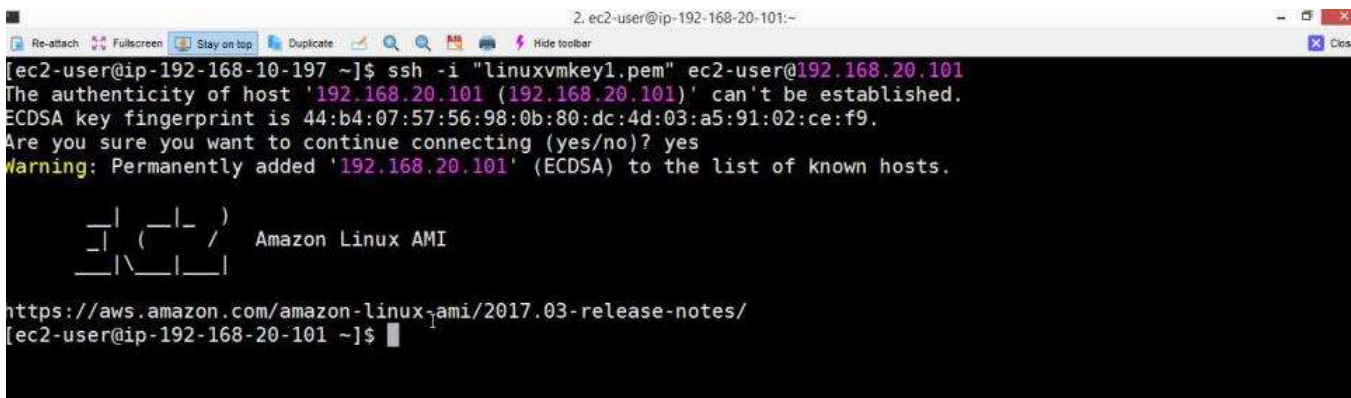
View the details of private instance



Verification

Run ssh command to login to private instance

Now you are connected to private instance in private subnet



Networking Basics

What is TCP/IP Model?

OSI Model	TCP/IP Hierarchy	Protocols					
7 th Application Layer	Application Layer	HTTP	SMTP	POP3	FTP	...	
6 th Presentation Layer							
5 th Session Layer							
4 th Transport Layer	Transport Layer	TCP		UDP			
3 rd Network Layer	Network Layer	IP					ICMP
2 nd Link Layer	Link Layer	ARP RARP		PPP	...		
1 st Physical Layer		Ethernet					

Link Layer: Includes device driver and network interface card

Network Layer: Handles the movement of packets, i.e. Routing

Transport Layer: Provides a reliable flow of data between two hosts

Application Layer: Handles the details of the particular application

IP

Responsible for end to end transmission, sends data in individual packets, Maximum size of packet is determined by the networks Fragmented if too large Unreliable Packets might be lost, corrupted, duplicated, delivered out of order.

IP addresses

4 bytes

e.g. 163.1.125.98

Each device normally gets one (or more)

In theory there are about 4 billion available

Routing

How does a device know where to send a packet?

All devices need to know what IP addresses are on directly attached networks. If the destination is on a local network, send it directly there.

Suppose, If the destination address isn't local. Most non-router devices just send everything to a single local router. Routers need to know which network corresponds to each possible IP address.

Allocation of addresses

Controlled centrally by ICANN

- Fairly strict rules on further delegation to avoid wastage
- Have to demonstrate actual need for them
- Organizations that got in early have bigger allocations than they really need

IP packets

Source and destination addresses

Protocol number

1 = ICMP, 6 = TCP, 17 = UDP

Various options

e.g. to control fragmentation

Time to live (TTL)

Prevent routing loops

IP Datagram

0	4	8	16	19	24	31
Vers	Len	TOS	Total Length			
Identification			Flags	Fragment Offset		
TTL		Protocol	Header Checksum			
Source Internet Address						
Destination Internet Address						
Options...					Padding	
Data...						

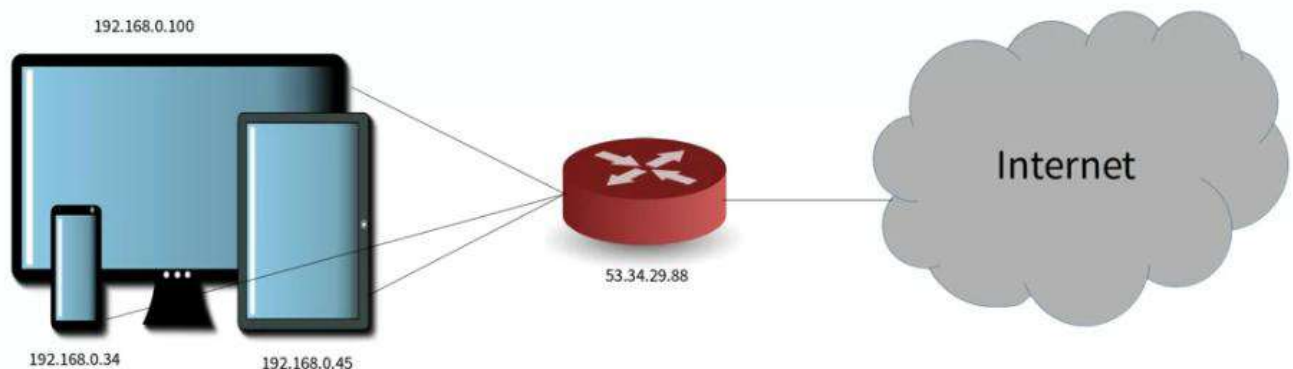
Field Purpose

Vers IP version number

Len Length of IP header (4 octet units)

TOS	Type of Service
T. Length	Length of entire datagram (octets)
Ident.	IP datagram ID (for frag/reassembly)
Flags	Don't/More fragments
Frag Off	Fragment Offset
TTL	Time To Live - Max # of hops
Protocol	Higher level protocol (1=ICMP, 6=TCP, 17=UDP)
Checksum	Checksum for the IP header
Source IA	Originator's Internet Address
Dest. IA	Final Destination Internet Address
Options	Source route, time stamp, etc.
Data...	Higher level protocol data

NAT Translation



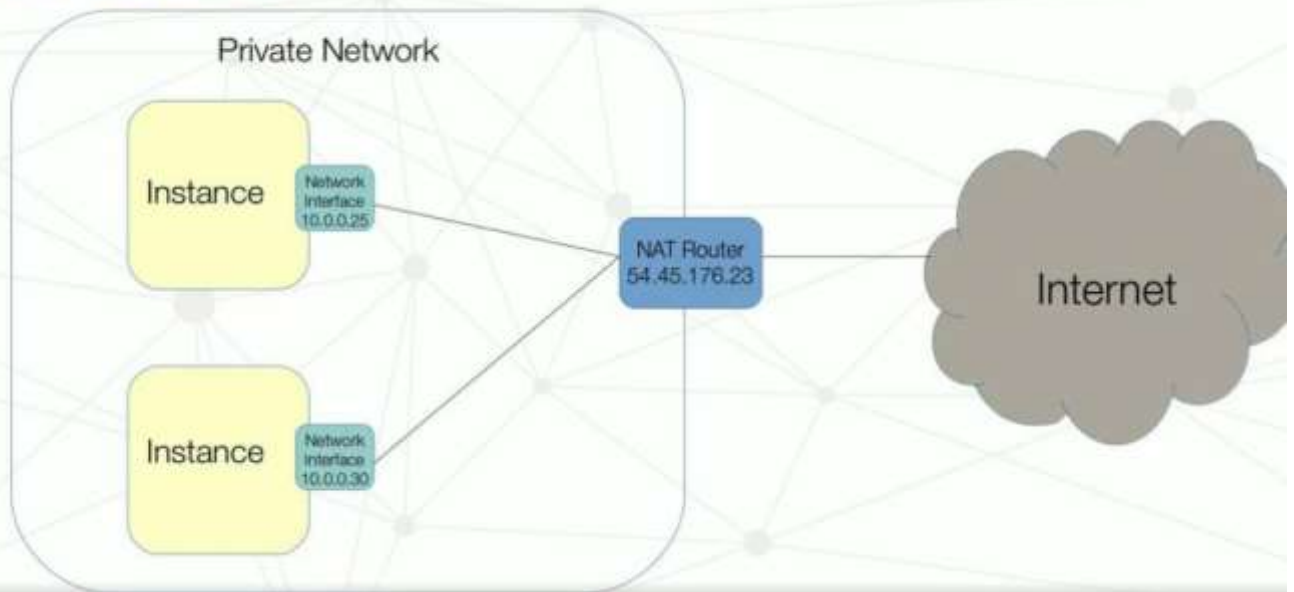
Reserved NAT Ranges:

Start:	End:	Number of addresses:
10.0.0.0	10.255.255.255	16,777,216
172.16.0.0	172.31.255.255	1,048,576
192.168.0.0	192.168.255.255	65,536

EC2 – Classic Vs VPC Networks

EC2 Classic	VPC
Part of AWS Network	Discrete Networks
Instance bound to group	Apply new group to running instance
Instance and group must be from the same region	Able to attach group from any region

NAT IP Translation:



Reserved NAT Ranges:

Start	End	No. of addresses
10.0.0.0	10.255.255.255	16777216
172.16.0.0	172.31.255.255	1048576
192.168.0.0	192.168.255.255	65536

What is Amazon VPC?

It enables you to launch Amazon Web Services (AWS) resources into a virtual network that you have defined. This virtual network closely resembles a traditional network that you would operate in your own data center, with the benefits of using the scalable infrastructure of AWS.

What is DNS?

If you have used the internet, you have used DNS. DNS is used to convert human friendly domain names (e.g. <http://amazon.com>) into an Internet Protocol (IP) address (e.g. <http://192.68.56.1>)

IP addresses are used by computers to identify each other on the network. IP addresses commonly come in two different forms such as IPV4 and IPV6.

What is CIDR?

Classless inter-domain routing (CIDR) is a set of Internet protocol (IP) standards that is used to create unique identifiers for networks and individual devices. The IP addresses allow particular information packets to be sent to specific computers. ... That system is known as CIDR notation.

What is Subnet in AWS?

VPC and Subnet Basics. A virtual private cloud (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch your AWS resources, such as Amazon EC2 instances, into your VPC.

A subnetwork or subnet is a logical subdivision of an IP network. The practice of dividing a network into two or more networks is called subnetting. Computers that belong to a subnet are addressed with a common, identical, most-significant bit-group in their IP address.

What is Route Tables?

A route table contains a set of rules, called routes, that are used to determine where network traffic is directed. Each subnet in your VPC must be associated with a route table; the table controls the routing for the subnet. A subnet can only be associated with one route table at a time, but you can associate multiple subnets with the same route table.

The following are the basic things that you need to know about route tables:

- Your VPC has an implicit router.
- Your VPC automatically comes with a main route table that you can modify.
- You can create additional custom route tables for your VPC.
- Each subnet must be associated with a route table, which controls the routing for the subnet. If you don't explicitly associate a subnet with a particular route table, the subnet is implicitly associated with the main route table.
- You cannot delete the main route table, but you can replace the main route table with a custom table that you've created (so that this table is the default table each new subnet is associated with).

What is Internet Gateways?

An Internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the Internet. It therefore imposes no availability risks or bandwidth constraints on your network traffic.

An Internet gateway serves two purposes: to provide a target in your VPC route tables for Internet-routable traffic, and to perform network address translation (NAT) for instances that have been assigned public IPv4 addresses.

An Internet gateway supports IPv4 and IPv6 traffic.

To enable access to or from the Internet for instances in a VPC subnet, you must do the following:

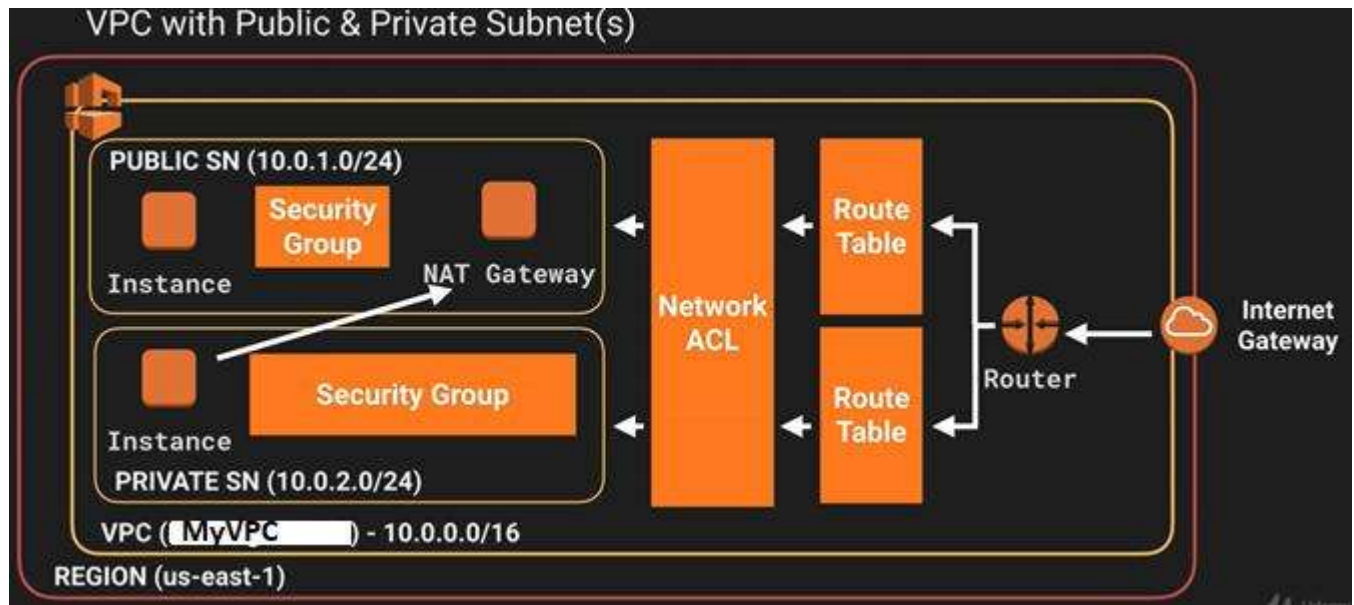
- Attach an Internet gateway to your VPC.
- Ensure that your subnet's route table points to the Internet gateway.
- Ensure that instances in your subnet have a globally unique IP address (public IPv4 address, Elastic IP address, or IPv6 address).
- Ensure that your network access control and security group rules allow the relevant traffic to flow to and from your instance.

What is IPV4 and IPV6?

IPV4 space is a 32-bit field and has over 4 billion different address (4,294,967,296 to be precise)

IPV6 was created to solve the depletion issue and has an address space of 128 bits which is 340 undecillion addresses (340,282,366,920,938,463,374,607,431,768,211,456)

Sketch the VPC Flow?



What is NAT Instances?

- When creating a NAT instance, disable source | destination check on the Instance
- NAT instances must be in a public subnet
- There must be a route out of the private subnet to the NAT instance, in order for this to work
- The amount of traffic that NAT instances can support depends on the instance size. If you are bottlenecking, increase the instance size.
- You can create high availability using autoscaling groups, multiple subnets in different AZs, and a script to automate failover
- Behind a Security Group

What is NAT Gateway?

- Preferred by the enterprise
- Scale automatically up to 10Gbps
- No need to patch
- Not associated with security groups
- Automatically assigned a public IP address
- Remember to update your route tables
- No need to disable Source | Destination checks
- More secure than a NAT instance

What is Network ACLS?

- Your VPC automatically comes a default network ACL, and by default it allows all outbound and inbound traffic
- You can create custom network ACLS. By default, each custom network ACL denies all in-bound and outbound traffic until you add rules
- Each subnet in your VPC must be associated with a network ACL. If you don't explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default ACL.
- You can associate a network ACL with multiple subnets; however, a subnet can be associated with only one network ACL at a time. When you associate a network ACL with a subnet, the previous association is removed
- Network ACLs contain a numbered list of rules that is evaluated in order, starting with the lowest numbered rule.
- Network ACLs have separate inbound and outbound rules, and each rule can either allow or deny traffic
- Network ACLs are stateless; responses to allowed inbound traffic are subject to the rules for outbound traffic and vice versa.
- Block IP Addresses using network ACLs not Security Groups

What is VPC Flow Logs?



- You cannot enable flow logs for VPCs that are peered with your VPC unless the peer VPC is in your account
- You cannot tag a flow log
- After you have created a flow log, you cannot change its configuration; for example, you can't associate a different IAM role with the flow log.
- Traffic generated by instances when they contact the Amazon DNS server. If you use your own DNS server, then all traffic to that DNS server is logged.
- Traffic generated by a Windows instance for Amazon Windows license activation
- Traffic to and from 169.254.169.254 for instance metadata
- DHCP traffic
- Traffic to the reserved IP address for the default VPC router

In VPC with private and public subnets, database servers should ideally be launched into which subnet?

- With private and public subnets in VPC, database servers should ideally launch into private subnets.

What action is required to establish an Amazon VPC?

We need to assign a static internet-routable IP address to an Amazon VPC customer gateway.

Network ACLs: A network access control List (ACL) is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets. You might set up the network ACLs with rules similar to your security groups in order to add an additional layer of security to your VPC. For more information about the differences between security groups and network ACLs.

Security Groups: A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. When you launch an instance in a VPC, you can assign the instance to up to five security groups. Security groups act at the instance level, not at the subnet level. Therefore, each instance in a subnet in your VPC could be assigned to a different set of security groups. If you don't specify a particular group at launch time, the instance is automatically assigned to the default security group for the VPC.

Why use VPC in AWS?

Normally, each EC2 instance you launch is randomly assigned a public IP address in the Amazon EC2 address space. VPC allows you to create an isolated portion of the AWS cloud and launch EC2 instances that have private address in the range of your choice. (10.0.0.0 for instance)

Can you describe the steps to create default VPC in AWS?

We can create a default VPC, we do the following to set it up for you: -

1. Create a default subnet in each availability zone
2. Create an Internet gateway and connect it to your default VPC
3. Create a main route table for your default VPC with a rule that sends all traffic destined for the Internet gateway.
4. Create a default security group and associate it with your default VPC.
5. Create a default network access control list (ACL) and associate it with your default VPC.
6. Associate the default DHCP options set for your AWS account with your default VPC.

What are the three features provided by Amazon that you can increase and monitor the security?

Amazon VPC provides three features that you can use to increase and monitor the security for your VPC.

Security groups: Acts as a firewall for associated Amazon EC2 instances, controlling both inbound and outbound traffic at the instance level

Network Access Control List (ACLs) Act as a firewall for associated subnets, controlling both inbound and outbound traffic at the subnet level

Flow logs Capture information about the IP traffic going to and from network interfaces in your VPC.

What is the difference between Network ACLS and Security groups in AWS?

Network ACLS: A network access control list is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets. You might set up Network ACLS with rules similar to your security groups in order to add an additional layer of security to your VPC. For more information about the differences between security groups and network ACLs.

Security Groups: A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. When you launch an instance in a VPC, you can assign the instance up to five security groups. Security groups act at the instance level, not the subnet level. Therefore, each instance in a subnet in your VPC could be assigned to a different set of security groups. If you don't specify a particular group at launch time, the instance is automatically assigned to the default security group for the VPC.

The basic differences between network ACLs and Security groups are: -

1. Security Group operates at instance level, while network ACL operates at the subnet level
2. Supports allow rules and deny rules. Supports allow rules only is stateless.
3. Return traffic must be explicitly allowed by rules is stateful. Return traffic is automatically allowed, regardless of any rules
4. We process rules in number order when deciding whether to allow traffic. We evaluate all rules before deciding whether to allow traffic
5. Automatically applies to all instances in the subnets it is associated with. (not rely on security group). Applies to an instance only if someone specifies the security group when launching the instance or associates the security group with the instance later on.

What benefits to VPC security groups give you that EC2 security groups don't?

1. Being able to change the security group after the instance is launched
2. Being able to specify any protocol with a standard number, rather than just TCP, UDP or ICMP

We can get following benefits by using Virtual Private Cloud (VPC) in an AWS account: We can assign Static IPv4 addresses to our instances in VPC. These static IP addresses will persist even after restarting an instance. We can even use IPv6 addresses with our instances in VPC.

VPC also allows us to run our instances on single tenant hardware. We can define Access Control List (ACL) to add another layer of security to our instances in VPC. VPC also allows for changing the security group membership of instances while they are running.

If you want to launch Amazon Elastic Compute Cloud (EC2) instances and assign each instance a predetermined private IP address you should:

- A. Launch the instance from a private Amazon Machine Image (AMI).
- B. Assign a group of sequential Elastic IP address to the instances.
- C. Launch the instances in the Amazon Virtual Private Cloud (VPC).
- D. Launch the instances in a Placement Group.

Answer C

Explanation: The best way of connecting to your cloud resources (for ex- ec2 instances) from your own data center (for eg- private cloud) is a VPC. Once you connect your datacenter to the VPC in which your instances are present, each instance is assigned a private IP address which can be accessed from your datacenter. Hence, you can access your public cloud resources, as if they were on your own network.

Can I connect my corporate datacenter to the Amazon Cloud?

Yes, you can do this by establishing a VPN (Virtual Private Network) connection between your company's network and your VPC (Virtual Private Cloud), this will allow you to interact with your EC2 instances as if they were within your existing network.

Is it possible to change the private IP addresses of an EC2 while it is running/stopped in a VPC?

Primary private IP address is attached with the instance throughout its lifetime and cannot be changed, however secondary private addresses can be unassigned, assigned or moved between interfaces or instances at any point.

Why do you make subnets?

- A. Because there is a shortage of networks
- B. To efficiently utilize networks that have a large no. of hosts.**
- C. Because there is a shortage of hosts.
- D. To efficiently utilize networks that have a small no. of hosts.

Answer B

Explanation: If there is a network which has a large no. of hosts, managing all these hosts can be a tedious job. Therefore, we divide this network into subnets (sub-networks) so that managing these hosts becomes simpler.

Which of the following is true?

- A. You can attach multiple route tables to a subnet
- B. You can attach multiple subnets to a route table**
- C. Both A and B
- D. None of these.

Answer B

Explanation: Route Tables are used to route network packets, therefore in a subnet having multiple route tables will lead to confusion as to where the packet has to go. Therefore, there is only one route table in a subnet, and since a route table can have any no. of records or information, hence attaching multiple subnets to a route table is possible.

In CloudFront what happens when content is NOT present at an Edge location and a request is made to it?

- A. An Error “404 not found” is returned
- B. CloudFront delivers the content directly from the origin server and stores it in the cache of the edge location**
- C. The request is kept on hold till content is delivered to the edge location
- D. The request is routed to the next closest edge location

Answer B

Explanation: CloudFront is a content delivery system, which caches data to the nearest edge location from the user, to reduce latency. If data is not present at an edge location, the first time the data may get transferred from the original server, but from the next time, it will be served from the cached edge.

If I'm using Amazon CloudFront, can I use Direct Connect to transfer objects from my own data center?

Yes. Amazon CloudFront supports custom origins including origins from outside of AWS. With AWS Direct Connect, you will be charged with the respective data transfer rates.

If my AWS Direct Connect fails, will I lose my connectivity?

If a backup AWS Direct connect has been configured, in the event of a failure it will switch over to the second one. It is recommended to enable Bidirectional Forwarding Detection (BFD) when configuring your connections to ensure faster detection and failover. On the other hand, if you have configured a

backup IPsec VPN connection instead, all VPC traffic will failover to the backup VPN connection automatically. Traffic to/from public resources such as Amazon S3 will be routed over the Internet. If you do not have a backup AWS Direct Connect link or a IPsec VPN link, then Amazon VPC traffic will be dropped in the event of a failure.



Amazon CloudFront

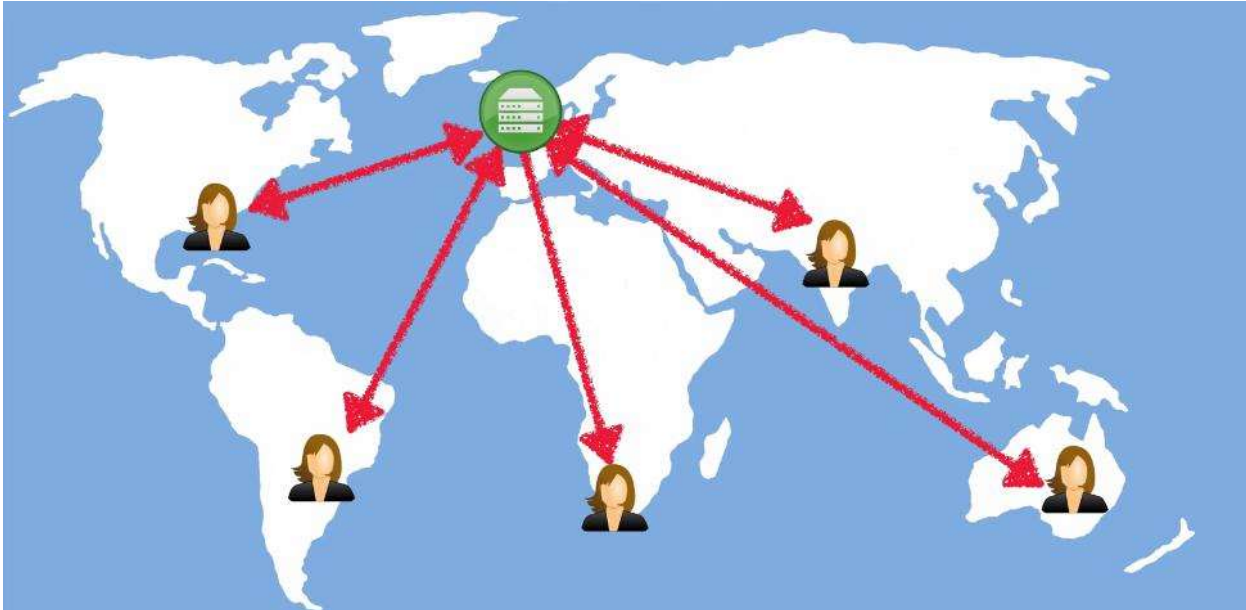
CloudFront Highlights

- **CloudFront** is a web service that speeds up distribution of your static and dynamic web content, for example, html, css, php, and image files, to end users. CloudFront delivers your content through a worldwide network of data centers called edge locations.
- When a user requests content that you're serving with CloudFront, the user is routed to the edge location that provides the lowest latency (time delay), so content is delivered with the best possible performance.
- If the content is already in the edge location with the lowest latency, CloudFront delivers it immediately. If the content is not currently in that edge location, CloudFront retrieves it from an Amazon S3 bucket or an HTTP server (for example, a web server) that you have identified as the source for the definitive version of your content.
- **Edge Location:** This is the location where content will be cached. This is separate to an AWS Region/AZ. There around 50+ edge locations
- **Origin:** This is the origin of all the files that the CDN will distribute. This can be either an S3 bucket, an EC2 instance, an Elastic Load Balancer or Route 53. Even it can be a Non-AWS Resource.
- **Distribution:** This is the name given the CDN which consists of a collection of Edge Locations. Two types => 1. Web distribution. 2. Rtmp [for media streaming]
- **Web Distribution:** Typically used for Websites
- Edge locations are not just **READ** only, you can write to them too (put an object on to them)
- Objects are cached for the life of the **TTL** (Time To Live)
- You can **clear** cached objects, but you will be **charged**

What is Content Delivery Network?

CDN stands for **CONTENT DELIVERY NETWORK**:

It is a system of distributed servers that deliver webpages and other web contents to the user based on the **geographic locations** of the user, the **origin** of the webpage & a **content delivery server**.



For example,

Server is in UK => the users, all over the world are accessing their webpages in UK server

They can access,

- ✓ a webpage – static/ dynamic
- ✓ movie file
- ✓ streaming file, etc

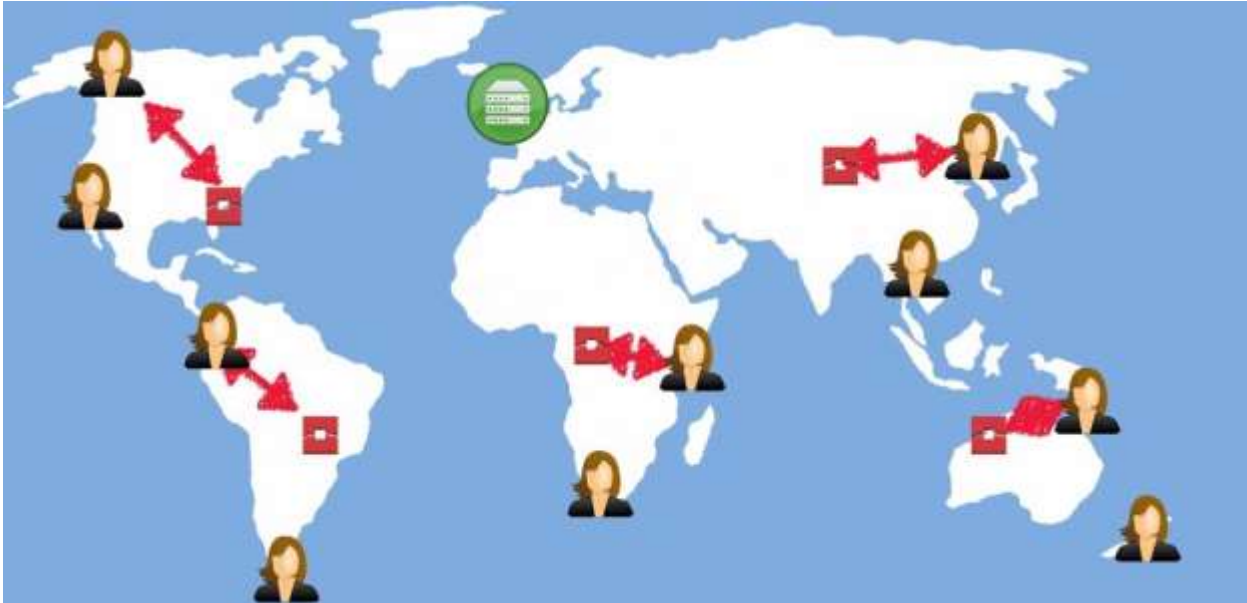
Multiple users in multiple part of the world:



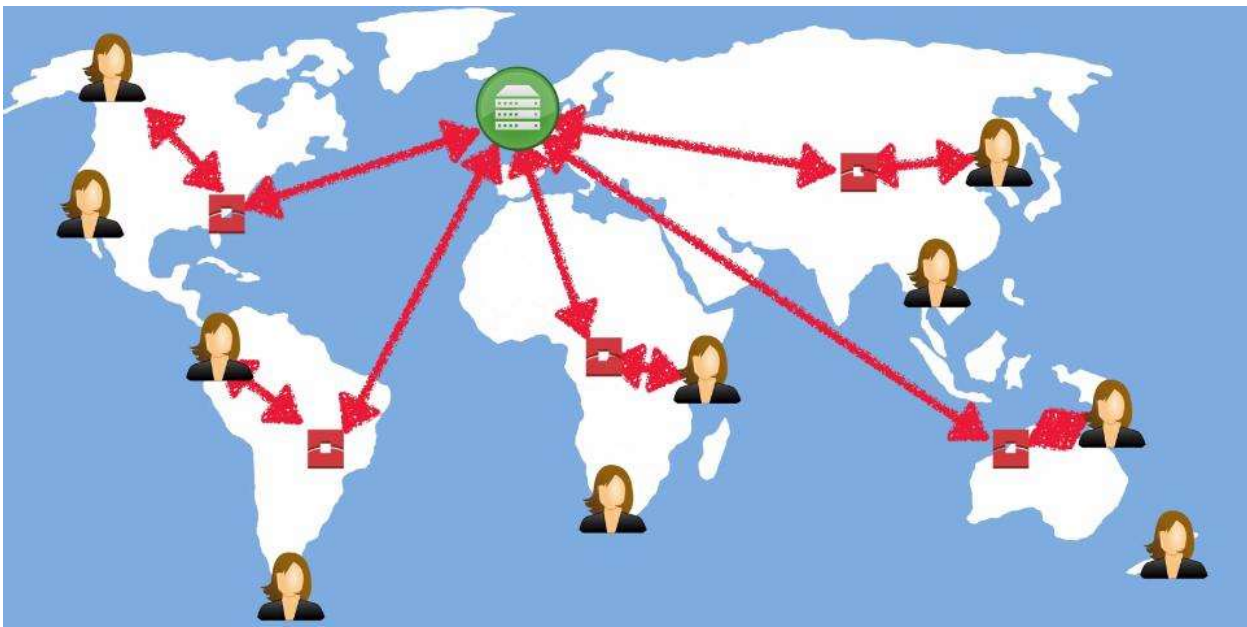
Edge locations spreads all across the world:



When the first user access the content & that goes to the edge locations:



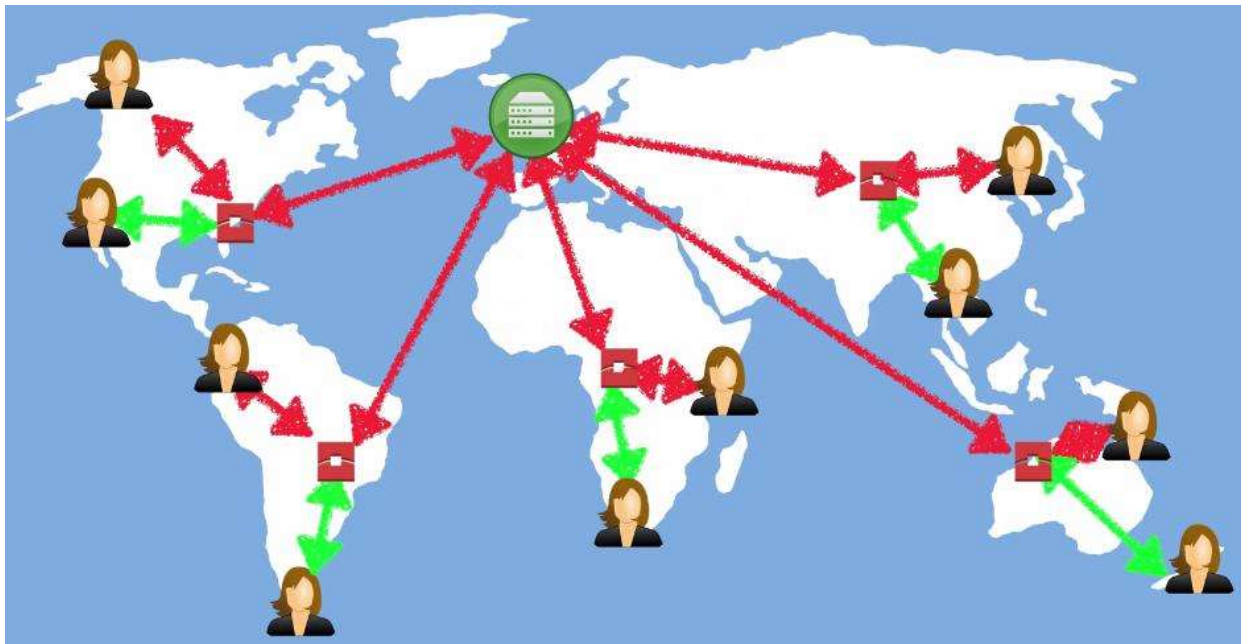
If it not cached in the edge loc => then as per distribution, it routes to the CDN server:



Thus, the first user accesses the content with so specialty, rather than a normal case

USER TO E.L => E.L TO ORIGIN [S3] => ORIGIN TO E.L => CACHES THE CONENT => SERVES THE USER.

But when the second user accesses the same data, it retrieves from the cached:



Important things about CDN:

- Edge Locations are not just READ only, you **CAN WRITE** new files to the E.L
- Objects are cached for the life of **TTL**
- You can **Clear the Cached Objects** from the Edge Location, but it will be charged

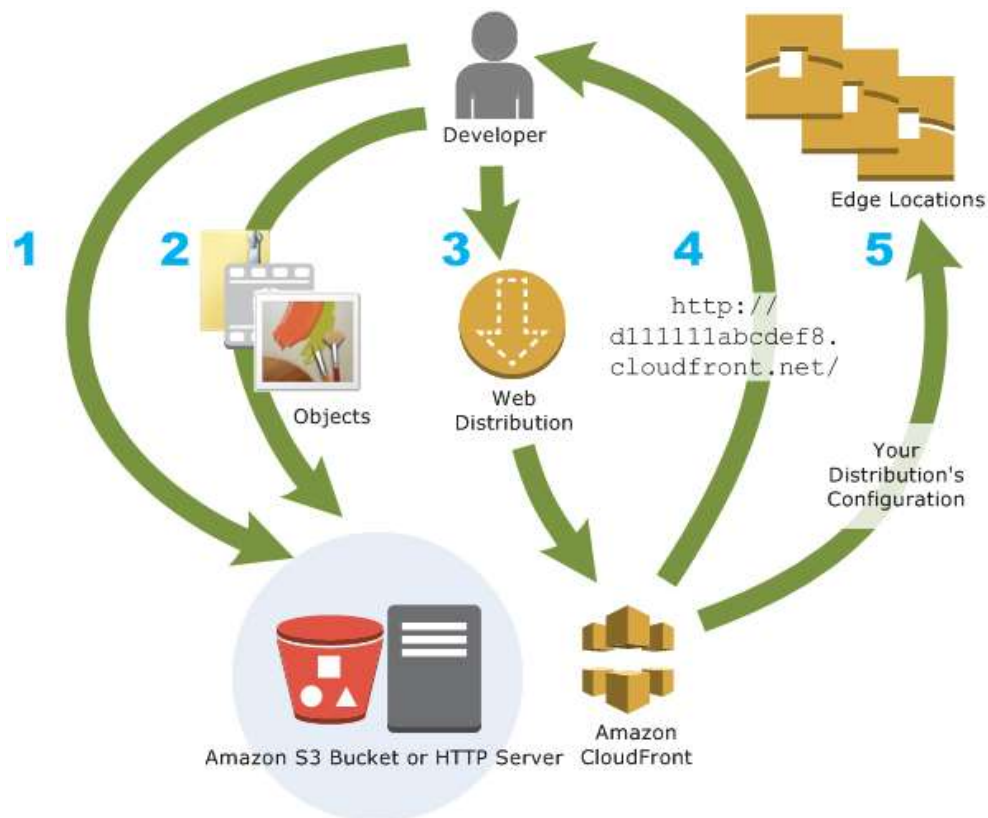
How to configure CloudFront to deliver the Content?

Setting up CloudFront involves a few simple steps:

1. You configure your **origin servers**, from which CloudFront gets your files for distribution from CloudFront edge locations all over the world.

An origin server stores the original, definitive version of your objects. If you're serving content over HTTP, your origin server is either an Amazon S3 bucket or an HTTP server, such as a web server. Your HTTP server can be running on an Amazon Elastic Compute Cloud (Amazon EC2) instance or on a server that you manage; these servers are also known as custom origins.

If you're distributing media files on demand using the Adobe Media Server RTMP protocol, your origin server is always an Amazon S3 bucket.



2. You **upload your files to your origin servers**. Your files, also known as **objects**, typically include web pages, images, and media files, but can be anything that can be served over HTTP or a supported version of Adobe RTMP, the protocol used by Adobe Flash Media Server.

If you're using an Amazon S3 bucket as an origin server, you can make the objects in your bucket publicly readable, so anyone who knows the CloudFront URLs for your objects can access them. You also have the option of keeping objects private and controlling who accesses them.

3. You create a CloudFront **distribution**, which tells CloudFront which origin servers to get your files from when users request the files through your web site or application. At the same time, you specify details such as whether you want CloudFront to log all requests and whether you want the distribution to be enabled as soon as it's created.

4. CloudFront sends your distribution's configuration (but not your content) to all of its **edge locations**—collections of servers in geographically dispersed data centers where CloudFront caches copies of your objects.

5. As you develop your website or application, you use the domain name that CloudFront provides for your URLs. For example, if CloudFront returns `d111111abcdef8.cloudfront.net` as the domain name for your distribution, the URL for `logo.jpg` in your Amazon S3 bucket (or in the root directory on an HTTP server) will be `http://d111111abcdef8.cloudfront.net/logo.jpg`.

You can also configure your CloudFront distribution so you can use your own domain name. In that case, the URL might be <http://www.example.com/logo.jpg>.

6. Optionally, you can configure your origin server to add headers to the files; the headers indicate how long you want the files to stay in the cache in CloudFront edge locations. By default, each object stays in an edge location for 24 hours before it expires. The minimum **expiration time** is 0 seconds; there isn't a maximum expiration time limit.

How CloudFront Delivers Content to Your Users?

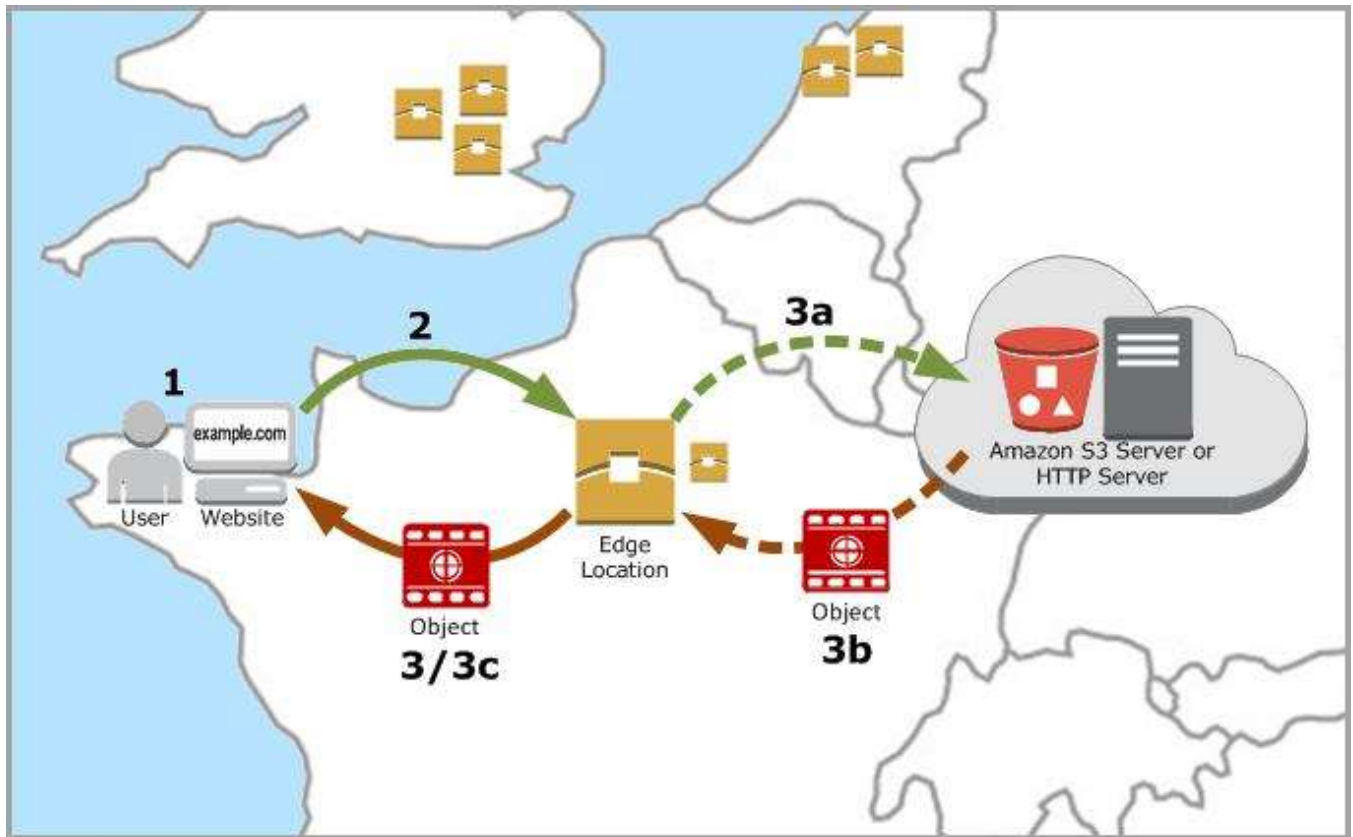
Once you configure CloudFront to deliver your content, here's what happens when users request your objects:

1. A user accesses your website or application and requests one or more objects, such as an image file and an HTML file.
2. DNS routes the request to the CloudFront edge location that can best serve the user's request, typically, the nearest CloudFront edge location in terms of latency, and routes the request to that edge location.
3. In the edge location, CloudFront checks its cache for the requested files. If the files are in the cache, CloudFront returns them to the user. If the files are not in the cache, it does the following:
 - a. CloudFront compares the request with the specifications in your distribution and forwards the request for the files to the applicable origin server for the corresponding file type—for example, to your Amazon S3 bucket for image files and to your HTTP server for the HTML files.
 - b. The origin servers send the files back to the CloudFront edge location.
 - c. As soon as the first byte arrives from the origin, CloudFront begins to forward the files to the user. CloudFront also adds the files to the cache in the edge location for the next time someone requests those files.
4. After an object has been in an edge cache for 24 hours or for the duration specified in your file headers, CloudFront does the following:
 - a. CloudFront forwards the next request for the object to your origin to determine whether the edge

location has the latest version.

b. If the version in the edge location is the latest, CloudFront delivers it to your user.

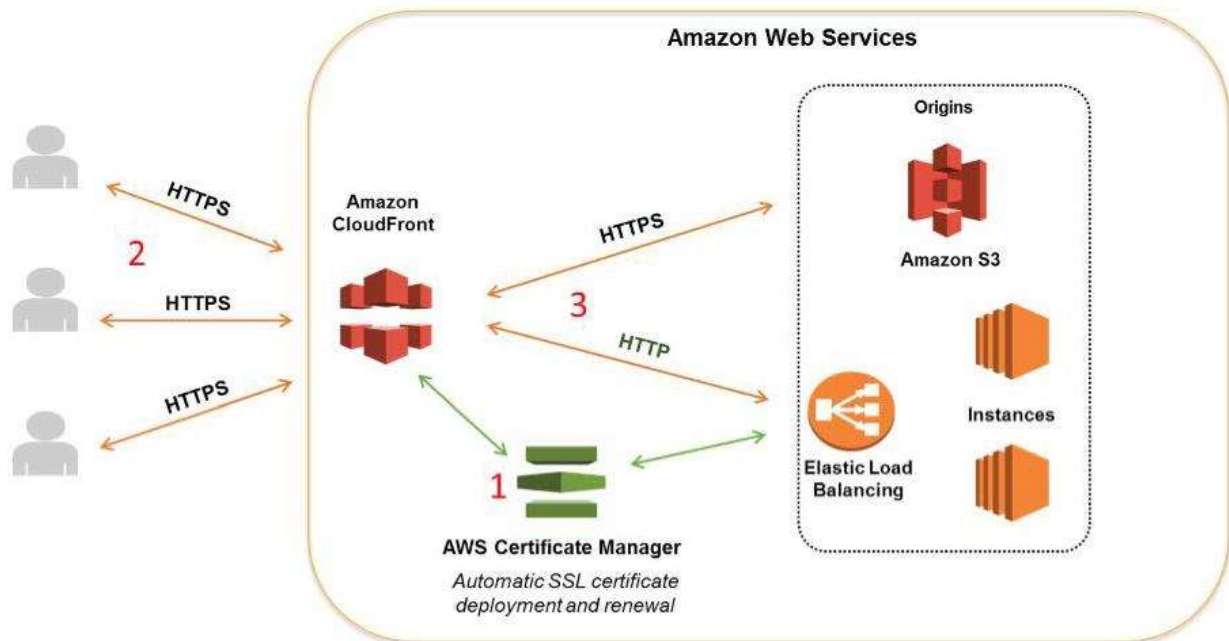
If the version in the edge location is not the latest, your origin sends the latest version to CloudFront, and CloudFront delivers the object to your user and stores the latest version in the cache at that edge location.



Share the CloudFront Configuration Step by Step?

Pre-requisites

- User should have AWS account or IAM user with CloudFront Full Access Policy

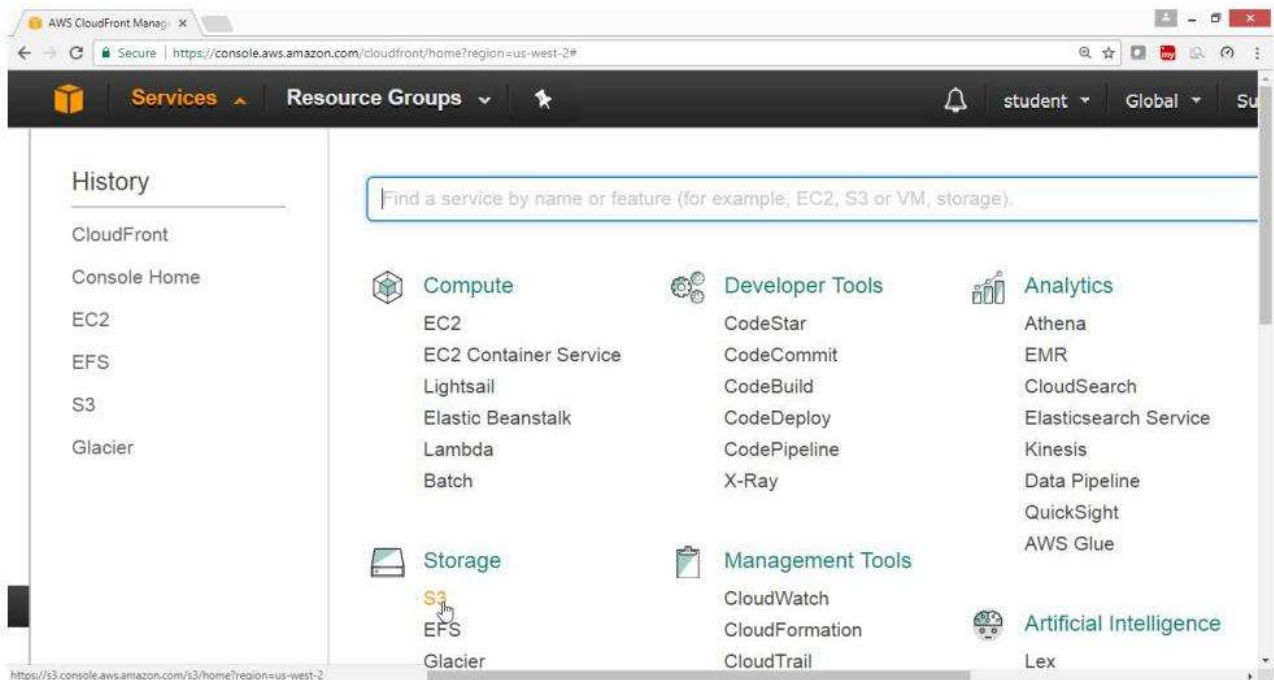


To configure CloudFront with following task

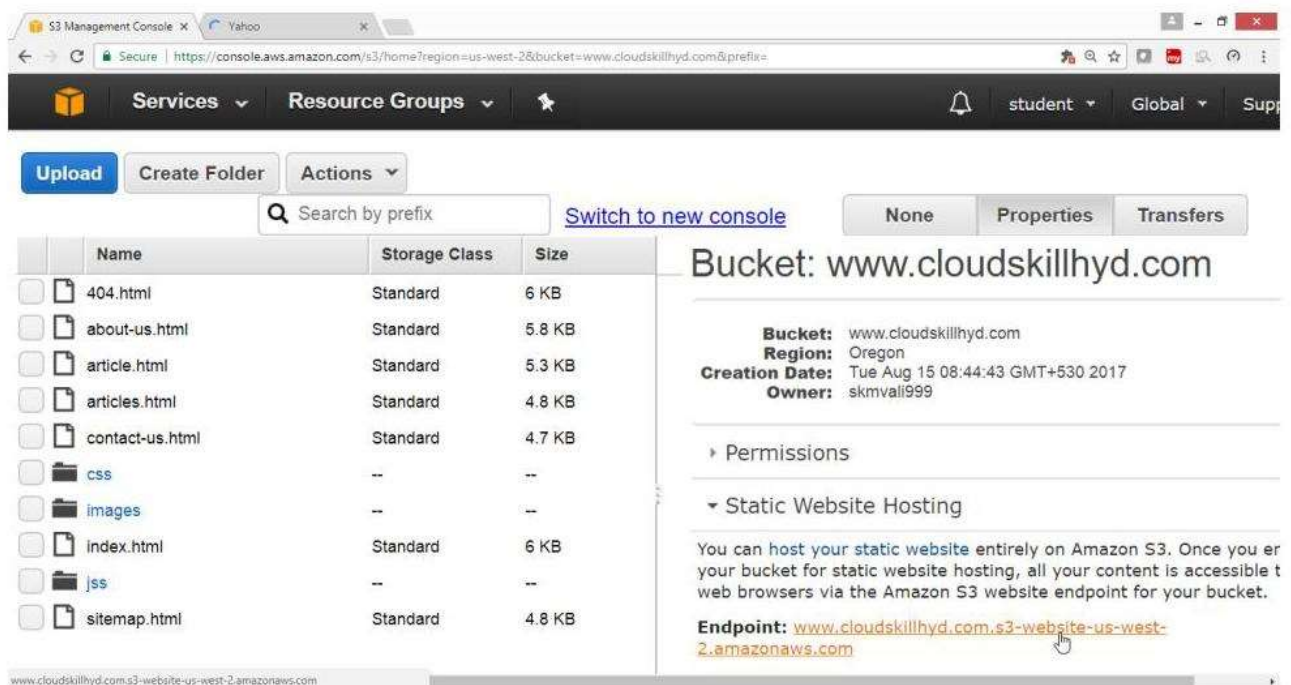
- Step – 1 Configure a website with Amazon S3 bucket by uploading your content
- Step – 2 Create a CloudFront Web Distribution
- Step – 3 Verify your site by providing CloudFront DNS link

Step - 1) Configure a website with Amazon S3 bucket by uploading your content

- Open AWS console goes for S3 Service
- Follow the law steps of Website Hosting in S3

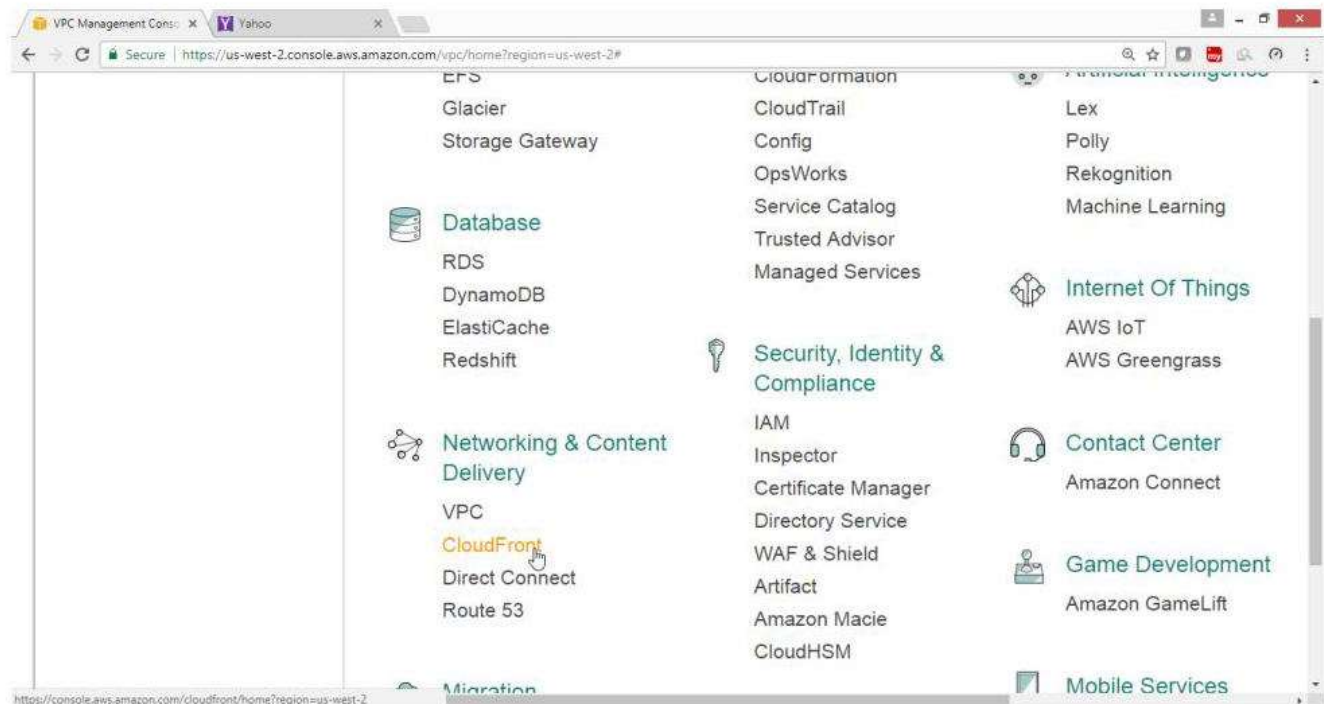


- Check the S3 bucket content

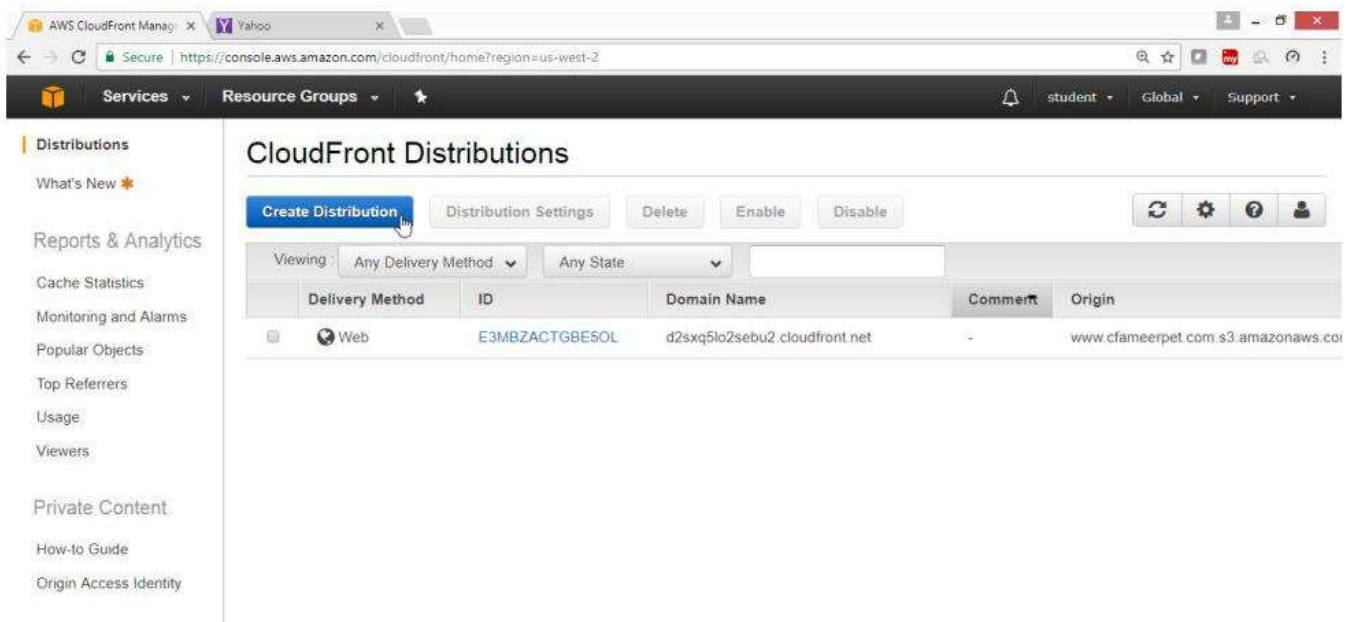


Step - 2) Create a CloudFront Web Distribution

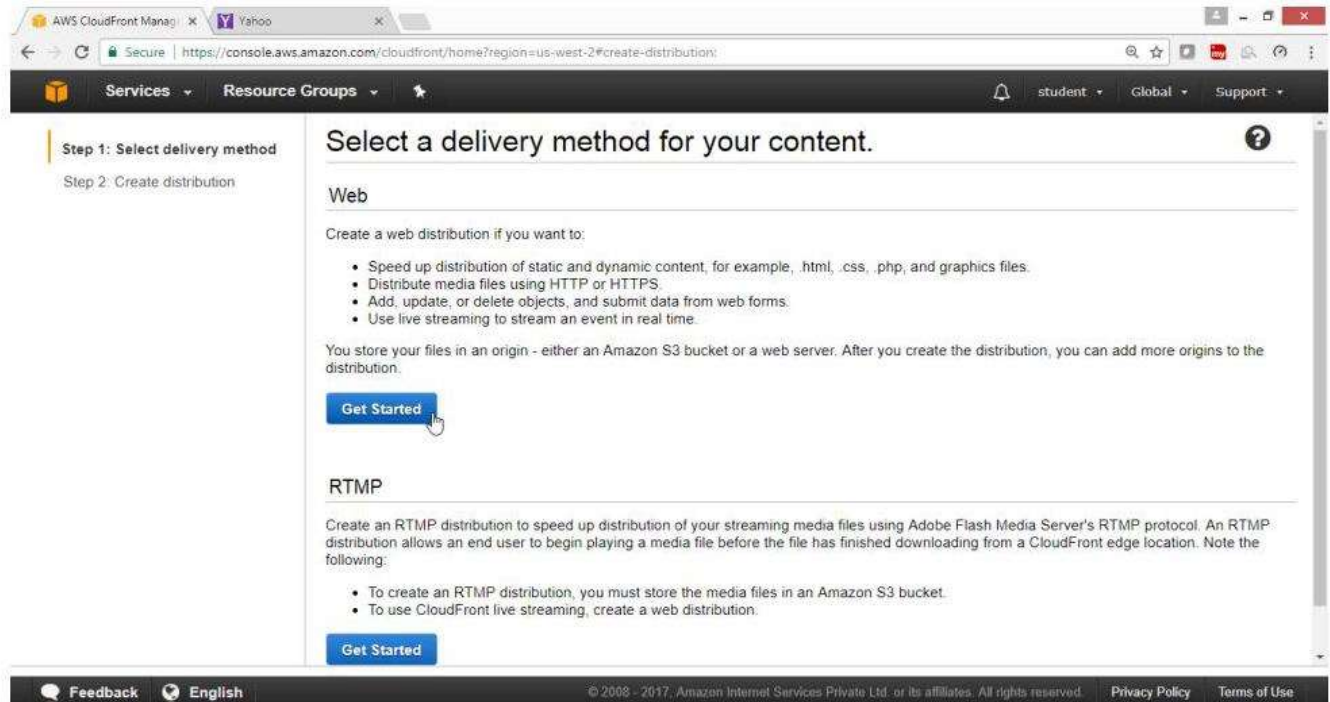
- Open AWS Console
- Select Networking and Content Delivery
- Click **CloudFront** Service



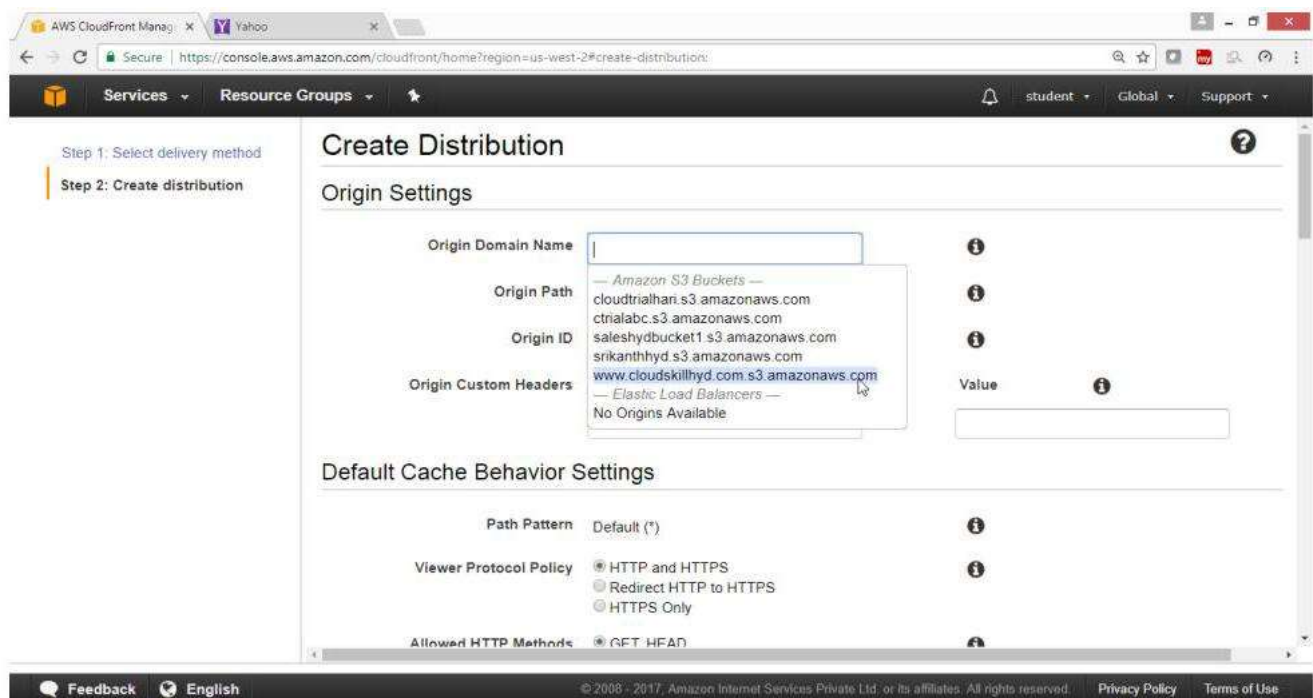
- Click on **Create Distribution** button



- Use "Select a delivery method for your content" wizard
- Under Web
- Click on **Get Started** button



- Under Create Distribution
- For Origin Domain Name -> Drop Down -> **www.cloudskill.com.s3.amazonaws.com**



Verify Origin Domain Name got selected

Price Class->Use only Canada and Europe

The screenshot shows the AWS CloudFront console's 'Distribution Settings' page. The left sidebar indicates 'Step 2: Create distribution'. The main settings area includes:

- Price Class:** A dropdown menu set to 'Use Only US, Canada and Europe'.
- AWS WAF Web ACL:** A dropdown menu set to 'None'.
- Alternate Domain Names (CNAMEs):** An empty text input field.
- SSL Certificate:** A radio button selection for 'Default CloudFront Certificate (*.cloudfront.net)'. Below this, a warning states: 'Choose this option if you want your users to use HTTPS or HTTP to access your content with the CloudFront domain name (such as https://d1111111abcde8.cloudfront.net/logo.jpg). Important: If you choose this option, CloudFront requires that browsers or devices support TLSv1 or later to access your content.'
- Below the SSL Certificate section, there is a radio button for 'Custom SSL Certificate (example.com):' with a corresponding warning: 'Choose this option if you want your users to access your content by using an alternate domain name, such as https://www.example.com/logo.jpg. You can use a certificate stored in AWS Certificate Manager (ACM) in the US East (N. Virginia) Region, or you can use a certificate stored in IAM.'
- At the bottom, there is a button labeled 'Request or Import a Certificate with ACM'.

Drag Down

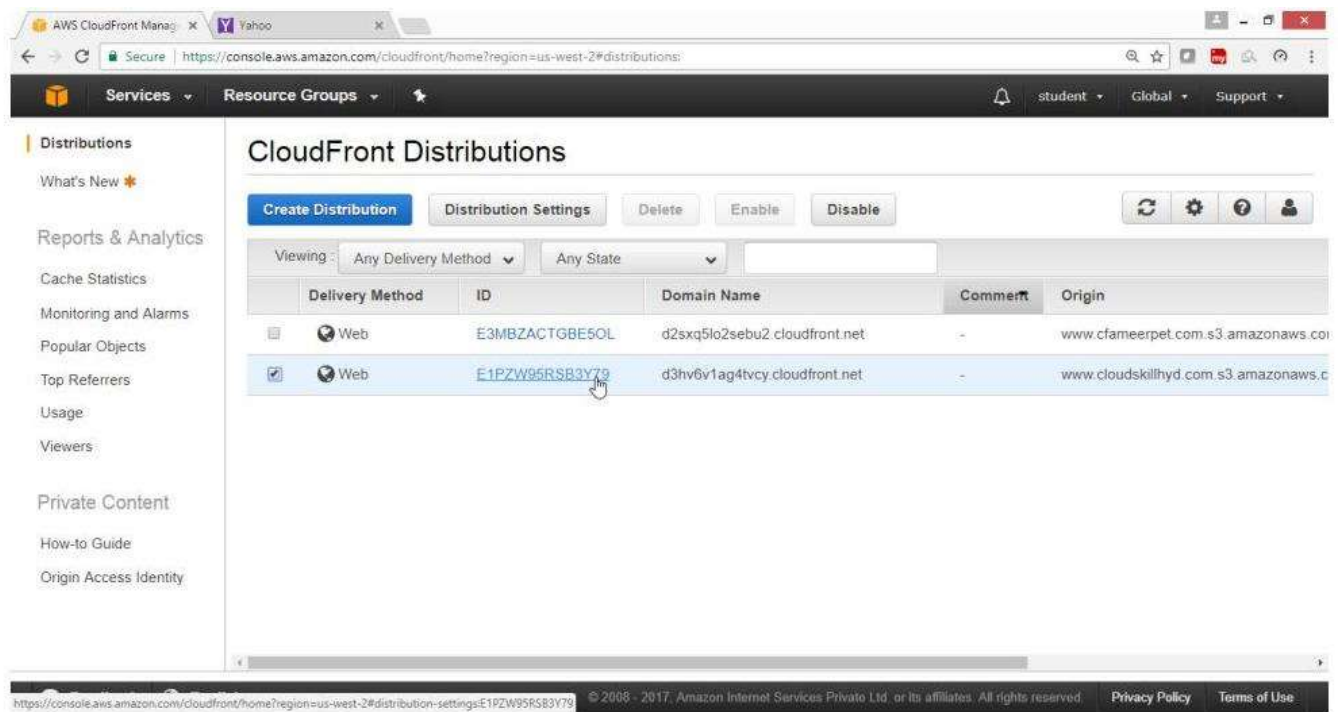
Click on **Create Distribution**

The screenshot shows the AWS CloudFront console interface for creating a new distribution. The left sidebar contains two steps: 'Step 1: Select delivery method' and 'Step 2: Create distribution', with the second step being the active one. The main content area is titled 'Create distribution' and contains several configuration fields:

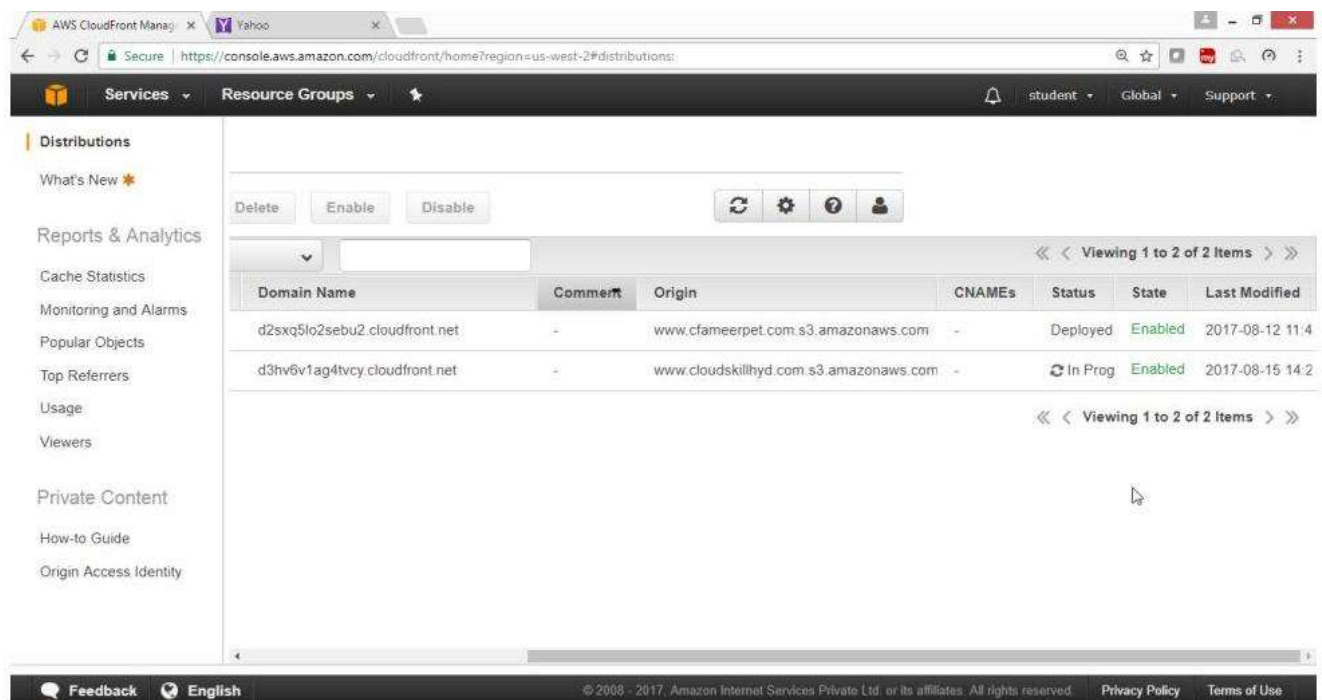
- Default Root Object:** An empty text input field.
- Logging:** Radio buttons for 'On' and 'Off', with 'Off' selected.
- Bucket for Logs:** An empty text input field.
- Log Prefix:** An empty text input field.
- Cookie Logging:** Radio buttons for 'On' and 'Off', with 'Off' selected.
- Enable IPv6:** A checked checkbox.
- Learn more:** A blue link.
- Comment:** An empty text input field.
- Distribution State:** Radio buttons for 'Enabled' and 'Disabled', with 'Enabled' selected.

At the bottom right, there are three buttons: 'Cancel', 'Back', and 'Create Distribution'. The 'Create Distribution' button is highlighted with a mouse cursor.

Verify the status



Check Column Status
Shows -> **In Progress**



Wait for status to gen **Enable**
Note: It takes around 15 minutes

AWS CloudFront Management Console screenshot showing the Distributions page. The left sidebar contains navigation links: Distributions, What's New, Reports & Analytics, Cache Statistics, Monitoring and Alarms, Popular Objects, Top Referrers, Usage, Viewers, Private Content, How-to Guide, and Origin Access Identity. The main content area displays a table of distributions with columns: Comment, Origin, CNAMEs, Status, State, and Last Modified. Two distributions are listed, both with a state of 'Enabled'.

Comment	Origin	CNAMEs	Status	State	Last Modified
lfront.net	www.cfameerpet.com.s3.amazonaws.com	-	Deployed	Enabled	2017-08-12 11:4
front.net	www.cloudskillhyd.com.s3.amazonaws.com	-	Deployed	Enabled	2017-08-15 14:2

Step - 3) Verify the site with DNS name "d3hv6v1ag4tvcy.cloudfront.net"

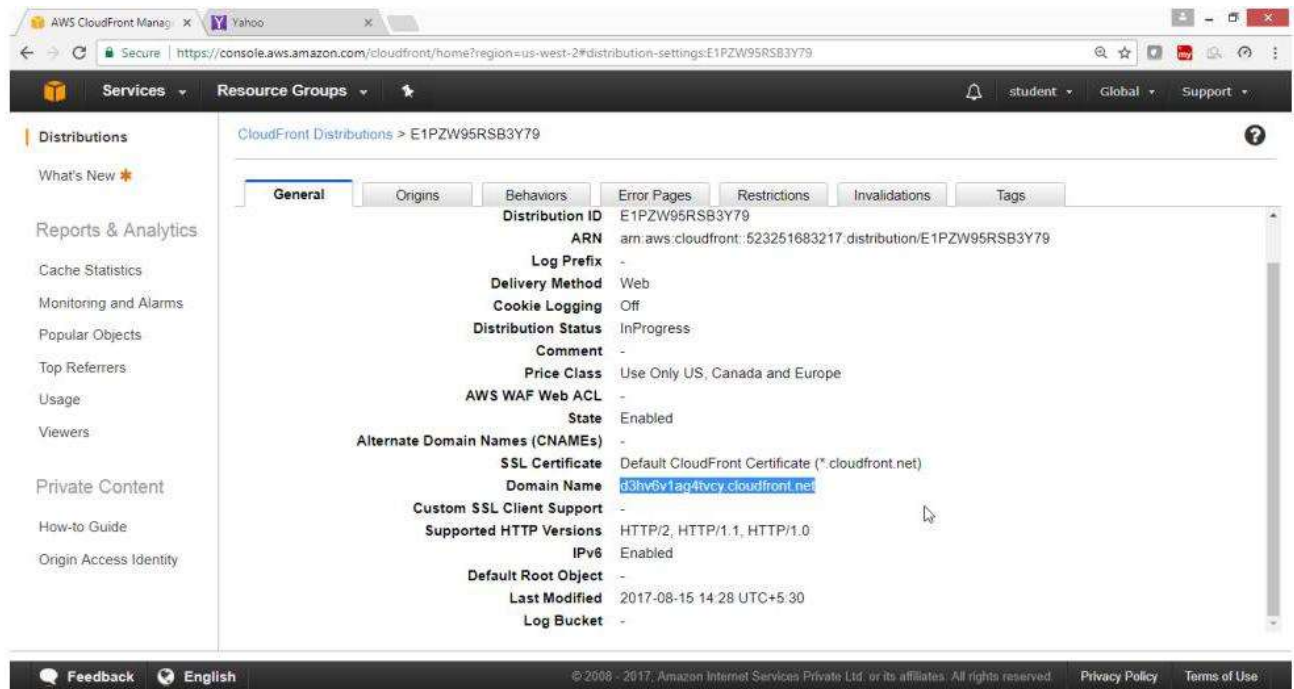
AWS CloudFront Management Console screenshot showing the CloudFront Distributions page. The left sidebar is identical to the previous screenshot. The main content area displays a table of distributions with columns: Delivery Method, ID, Domain Name, Comment, and Origin. Two distributions are listed, both with a state of 'Enabled'. The second distribution's domain name is highlighted.

Delivery Method	ID	Domain Name	Comment	Origin
Web	E3MBZACTGBE5OL	d2sxq5lo2sebu2.cloudfront.net	-	www.cfameerpet.com
Web	E1PZW95RSB3Y79	d3hv6v1ag4tvcy.cloudfront.net	-	www.cloudskillhyd.com

Verify

Now open the browser and type

<http://d3hv6v1ag4tvcy.cloudfront.net/index.html>



The website is coming from CloudFront Service



How do we get higher performance in our application by using Amazon CloudFront?

If our application is content rich and used across multiple locations, we can use Amazon CloudFront to increase its performance.

Some of the techniques used by Amazon CloudFront are as follows: -

Caching: Amazon CloudFront caches the copies of our application's content at locations closer to our viewers. By this caching our users get our content very fast. Also, due to caching the load on our main server decreases.

Edge / Regional Locations: CloudFront uses a global network of Edge and Regional edge locations to cache our content. These locations cater to almost all of the geographical areas across the world. **Persistent Connections:** In certain cases, CloudFront keeps persistent connections with the main server to fetch the content quickly.

Other Optimization: Amazon CloudFront also uses other optimization optimization techniques like TCP initial congestion window etc to deliver high performance experience.

What is the mechanism behind Regional Edge Cache in Amazon CloudFront?

A Regional Edge Cache location lies between the main webserver and the global edge location. When the popularity of an object/content decreases, the global edge location may take it out from the cache. But Regional Edge location maintains a larger cache.

Due to this the object/content can stay for long time in Regional Edge location. Due to this CloudFront does not have to go back to main webserver.

When it does not find any object in Global Edge location it just looks for in Regional Edge location. This improves the performance for serving content to our users in Amazon CloudFront.

What are the benefits of Streaming content?

We can get following benefits by Streaming content:

Control: We can provide more control to our users for what they want to watch. In a video streaming, users can select the locations in video where they want to start watching from.

Content: With streaming our entire content does not stay at a user's device. Users gets only the part they are watching. Once the session is over, content is removed from the user's device.

Cost: With streaming there is no need to download all the content to a user's device. A user can start viewing content as soon as some part is available for viewing. This saves costs since we do not have to download a large media file before starting each viewing session.

What are the different types of events triggered by Amazon CloudFront?

Different types of events triggered by Amazon CloudFront are as follows:

Viewer Request: When an end user or a client program makes an HTTP/HTTPS request to CloudFront, this event is triggered at the Edge Location closer to the end user.

Viewer Response: When a CloudFront server is ready to respond to a request, this event is triggered.

Origin Request: When CloudFront server does not have the requested object in its cache, the request is forwarded to Origin server. At this time this event is triggered.

Origin Response: When CloudFront server at an Edge location receives the response from Origin server, this event is triggered.

What is Geo Targeting in Amazon CloudFront?

In Amazon CloudFront we can detect the country from where end users are requesting our content. This information can be passed to our Origin server by Amazon CloudFront. It is sent in a new HTTP header.

Based on different countries we can generate different content for different versions of the same content. These versions can be cached at different Edge Locations that are closer to the end users of that country. In this way we are able to target our end users based on their geographic locations.

What are the main features of Amazon CloudFront?

Some of the main features of Amazon CloudFront are as follows:

Device Detection Protocol Detection Geo Targeting Cache Behavior Cross Origin Resource Sharing Multiple Origin Servers HTTP Cookies Query String Parameters Custom SSL.



Amazon Route 53

Route 53 Highlights

Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service.

It is designed to give developers and businesses an extremely reliable and cost-effective way to route end users to Internet applications by translating names like `www.example.com` into the numeric IP addresses like `192.0.2.1` that computers use to connect to each other. Amazon Route 53 is fully compliant with IPv6 as well.

The different routing policies

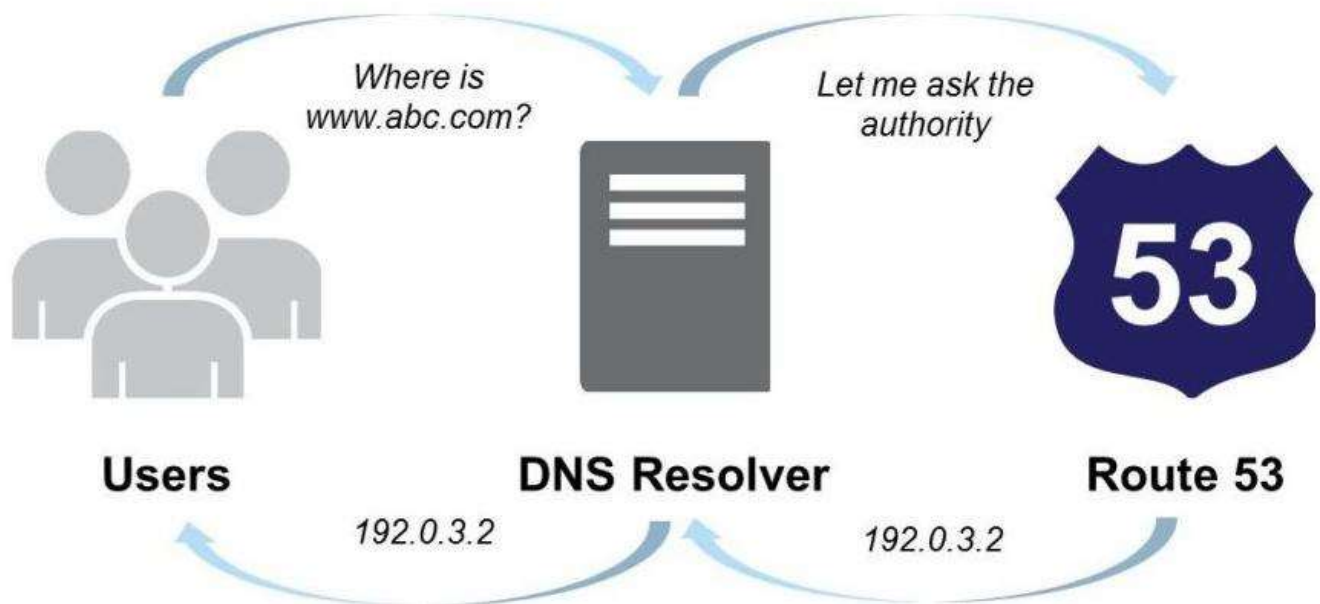
- Simple
- Weighted
- Latency
- Failover
- Geolocation

Share the Route 53 Configuration Step by Step?

Pre-requisites

To configure and use AWS Route 53 Service

Topology



Pre-requisites

- User should have AWS account, or IAM user with Amazon Route53 Full Access
- By default, AWS does not provide to Register Domain Name with AWS
- You should have a registered domain name one with your ISP

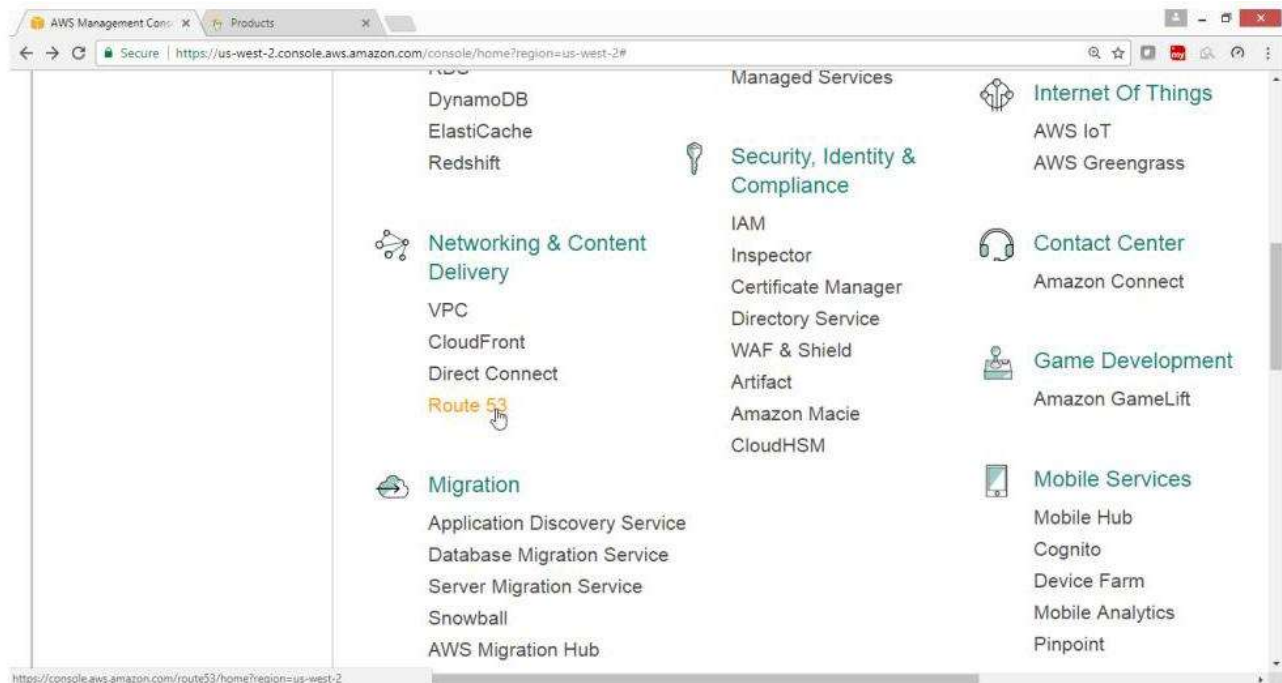
To configure Route53 with the following task: -

To Transfer existing DNS service from your ISP to Amazon Route53

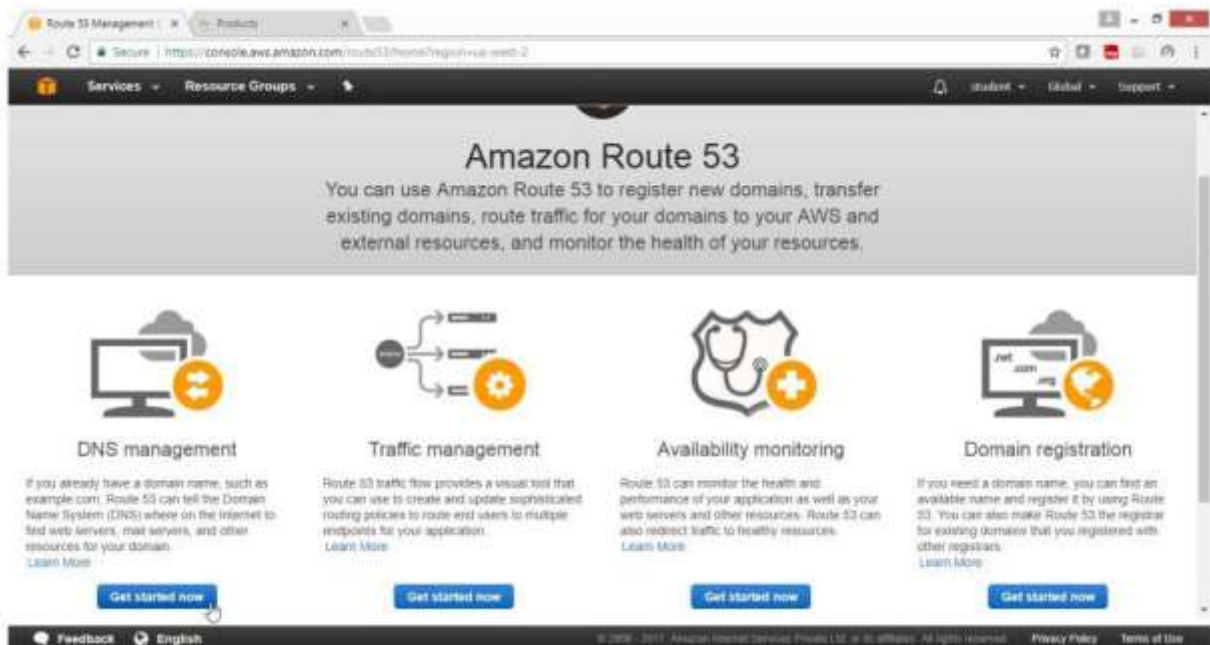
- Creating [record set](#)
- Create [CNAME](#) record set

Step-1: Configuration of Route53 for Domain Name

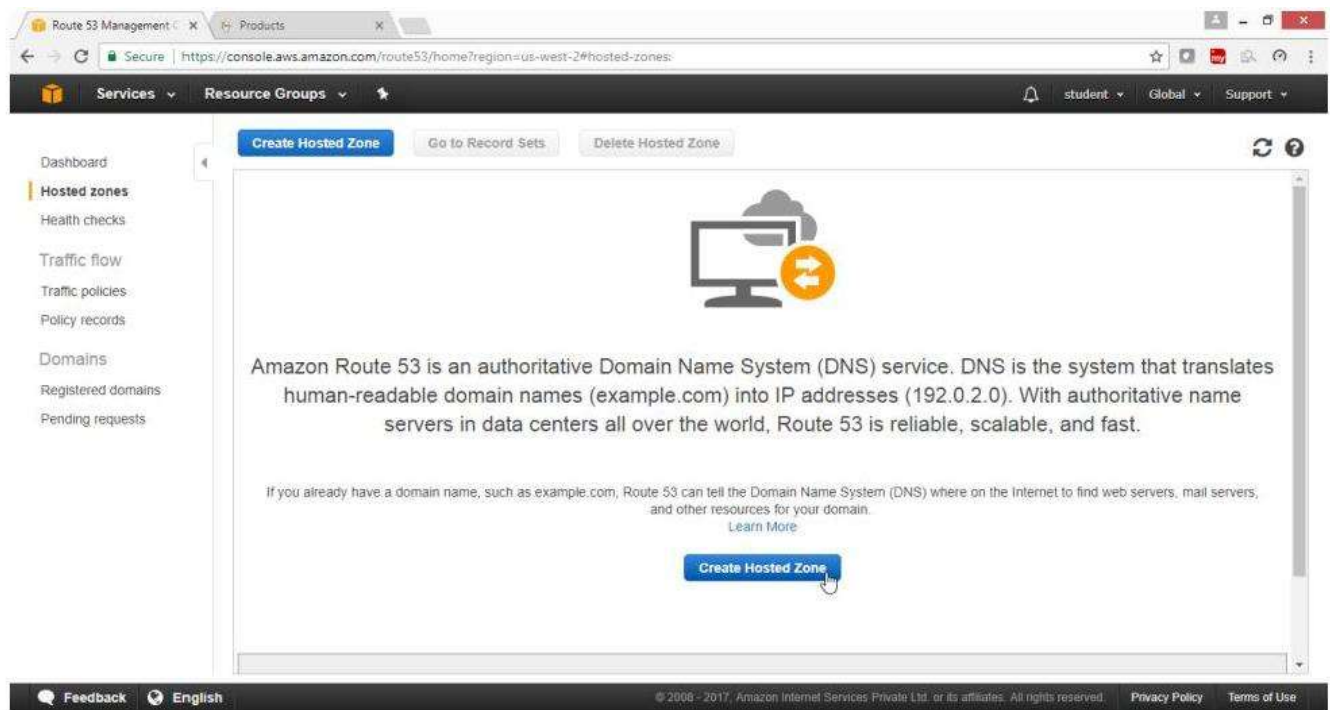
- Open **AWS console**
- Select **"Networking & Content Delivery"**
- Click on **Route53 Services**



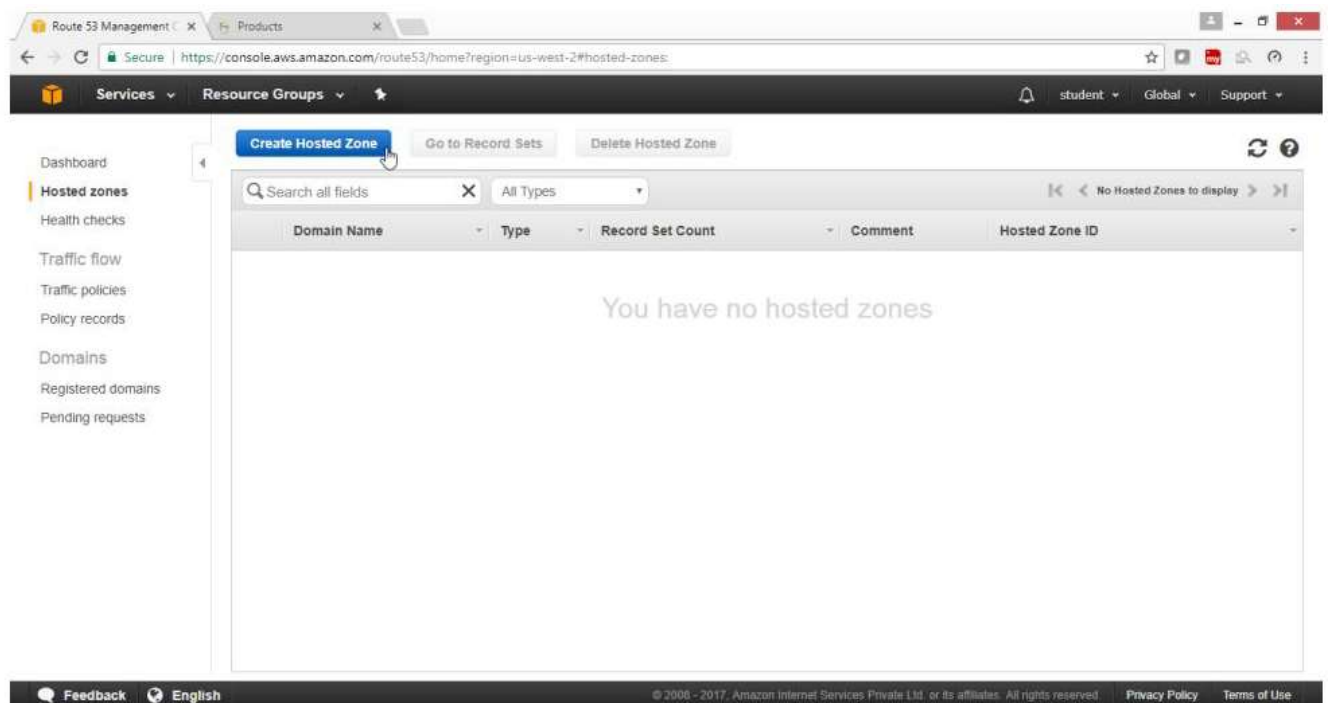
- **Route53 Dashboard wizard opens**
- **Under DNS management**
- Click on **"Get Started Now"** button



Click on **"Created Hosted Zone"** button



Again, Click on Created Hosted Zone button



Under "**Created Hosted Zone**" wizard

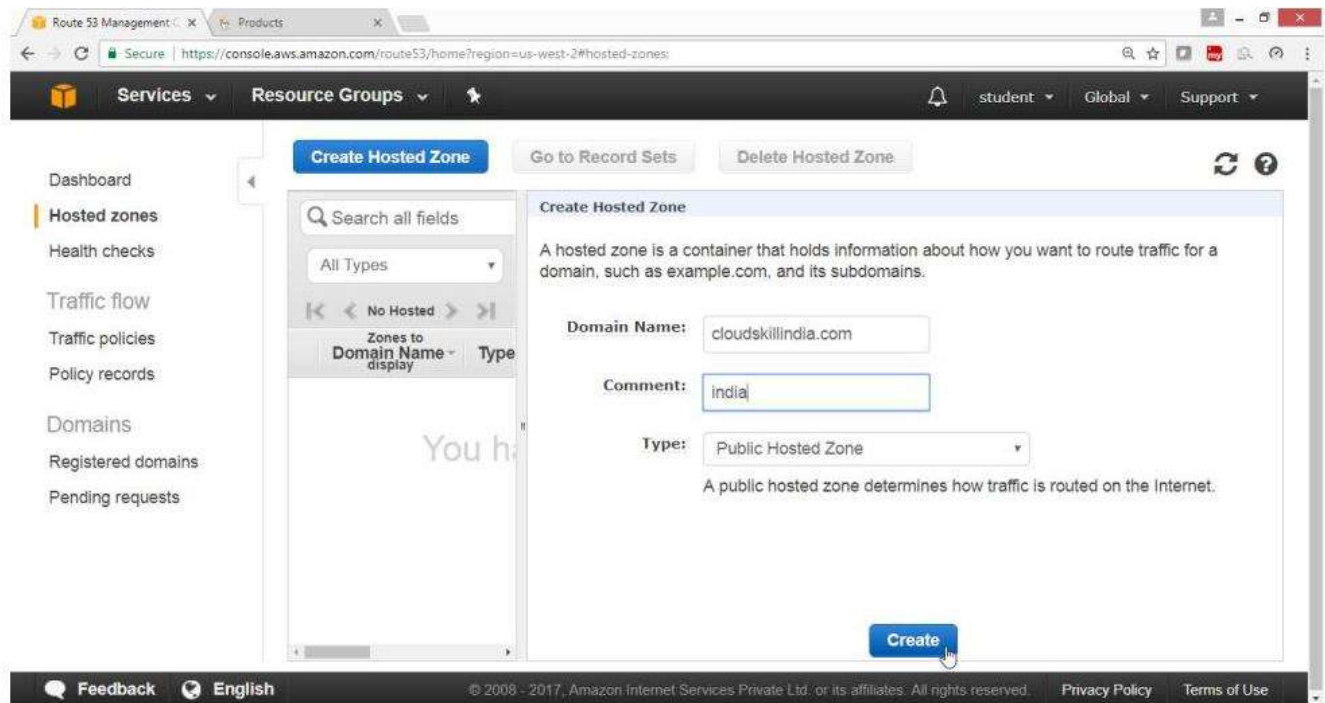
On right side panel provide the following values

For Domain Name: ->cloudskillindia.com

For Comment ->india

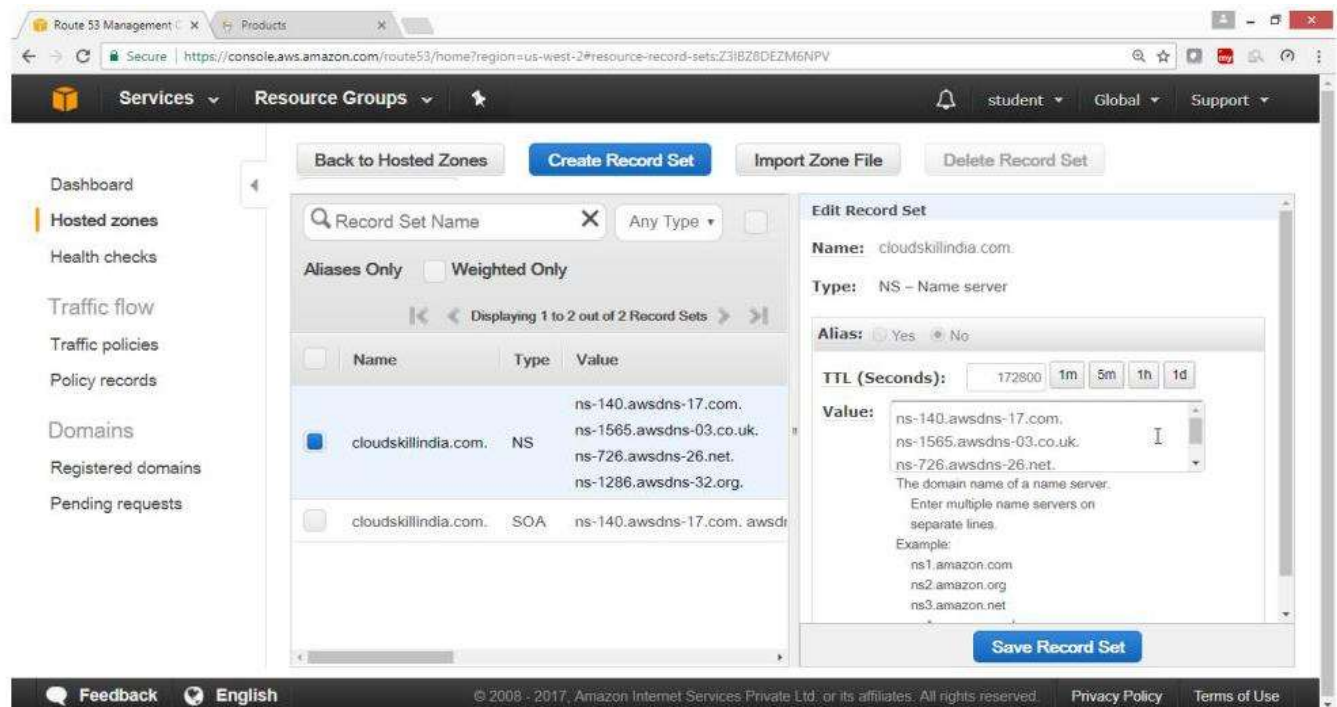
For Type -> Public Hosted Zone

Click on **Create** button



Now the list of AWS NS records will appear

Now add all AWS NS record to your DNS NS record (godaddy.com)



Step-2: Now copy these DNS NS record in godaddy.com for cloudskillindia.com domain

ns-140.awsdns-17.com

ns-1565.awsdns-03.co.uk

ns-726.awsdns-26.net

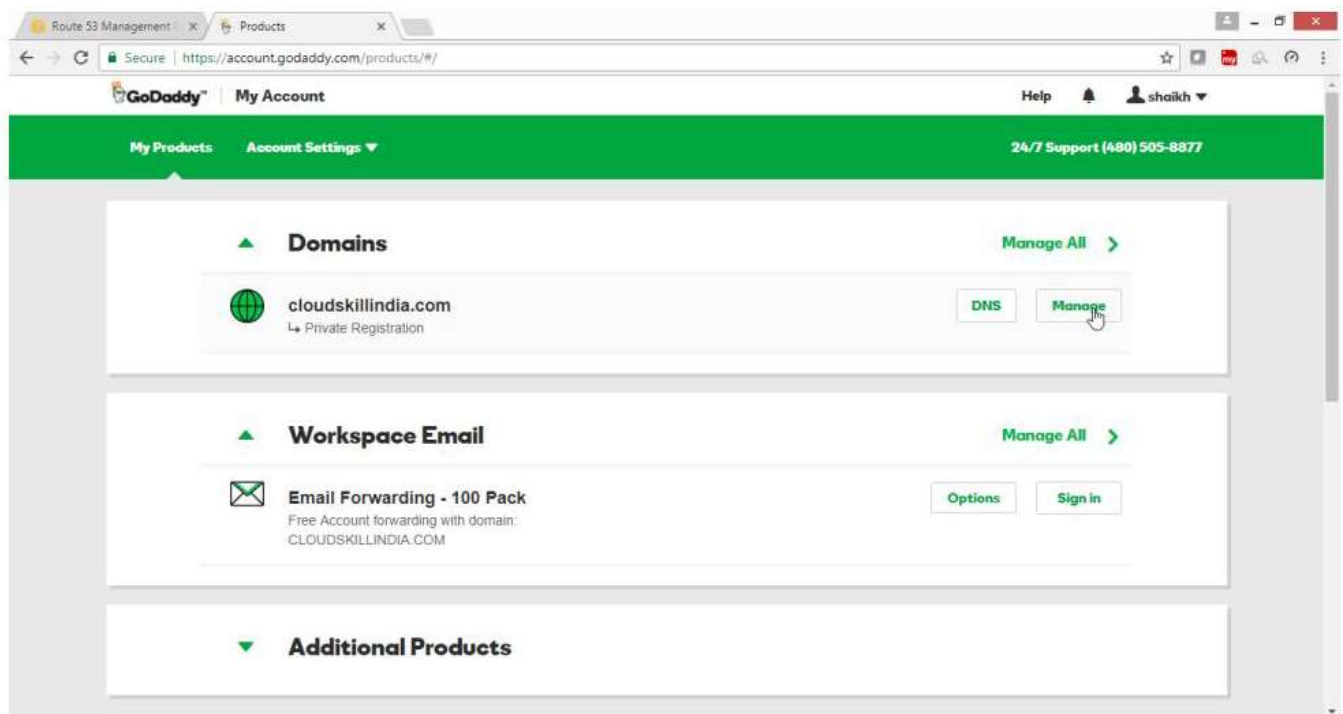
ns-1286.awsdns-32.org

Open the browser

Go to godaddy.com site

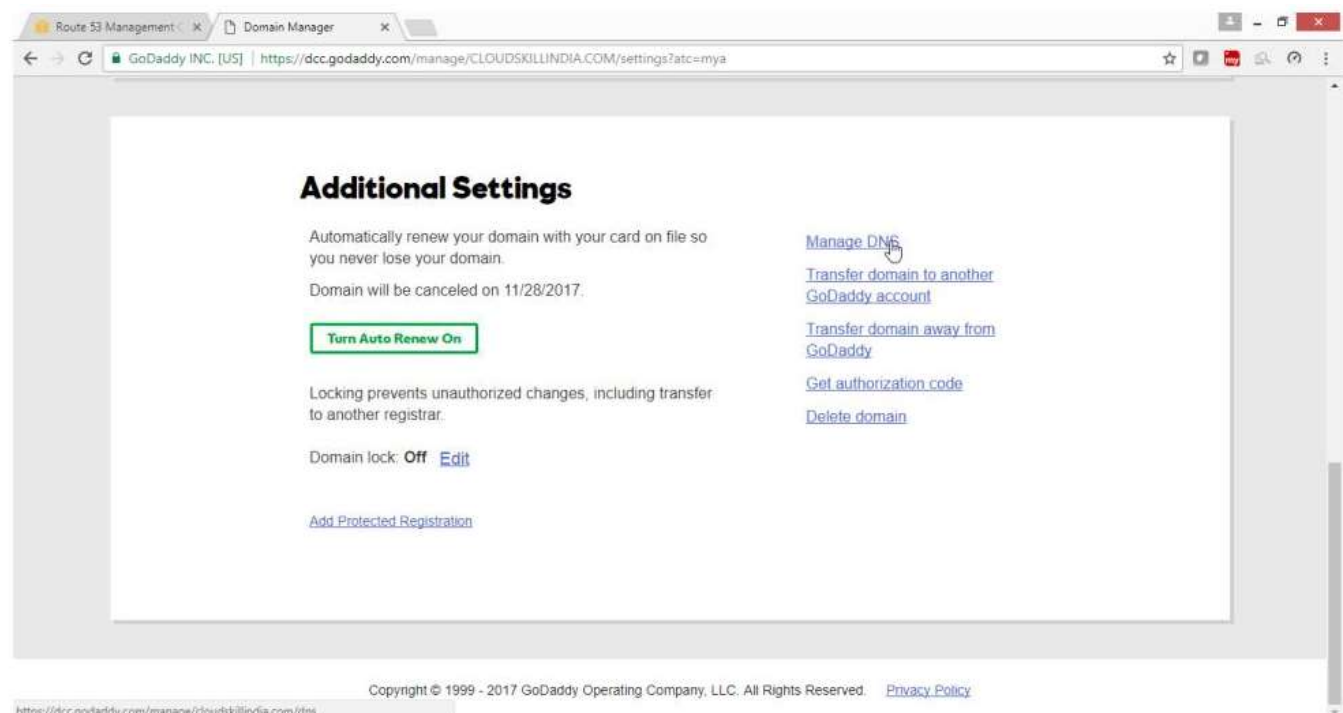
Login and select your domain name

Click on [Manage](#)



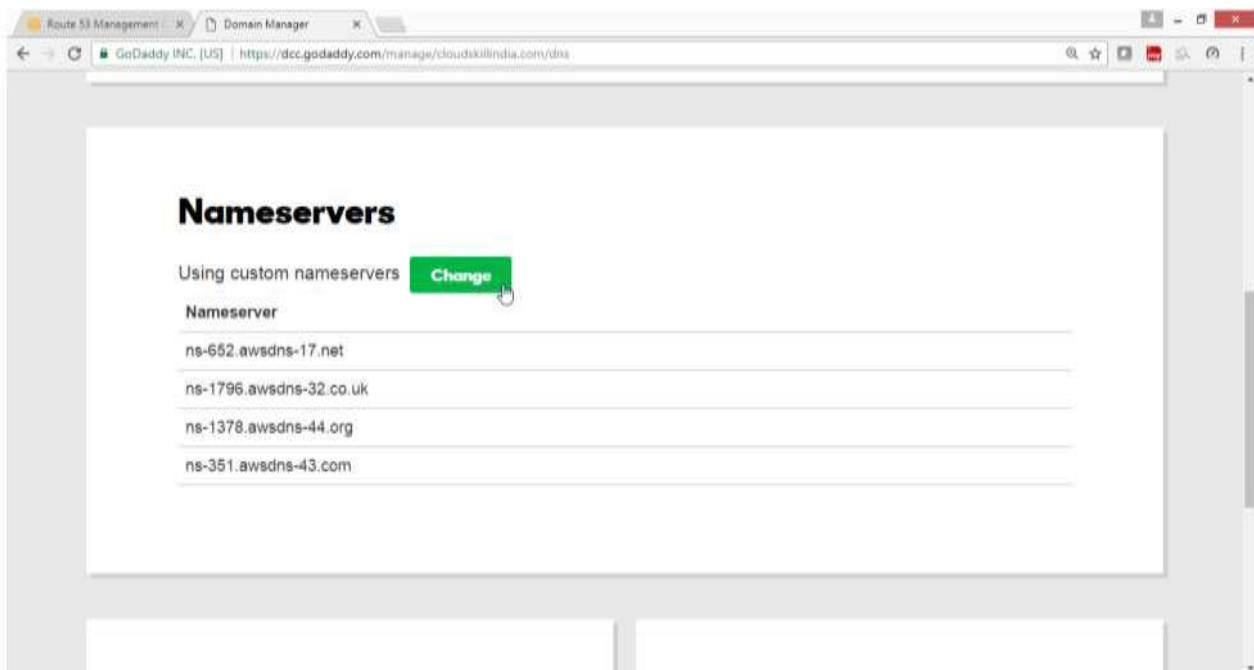
Drag Down

Click on [Manage DNS](#)



Click on [Change](#)

Add [latest entries](#) provided by Route53 NS records



The screenshot shows the 'Nameservers' section of the GoDaddy Domain Manager. It indicates that custom nameservers are being used. A green 'Change' button is visible. Below the heading, there is a list of four nameservers: ns-652.awsdns-17.net, ns-1796.awsdns-32.co.uk, ns-1378.awsdns-44.org, and ns-351.awsdns-43.com.

Nameservers

Using custom nameservers [Change](#)

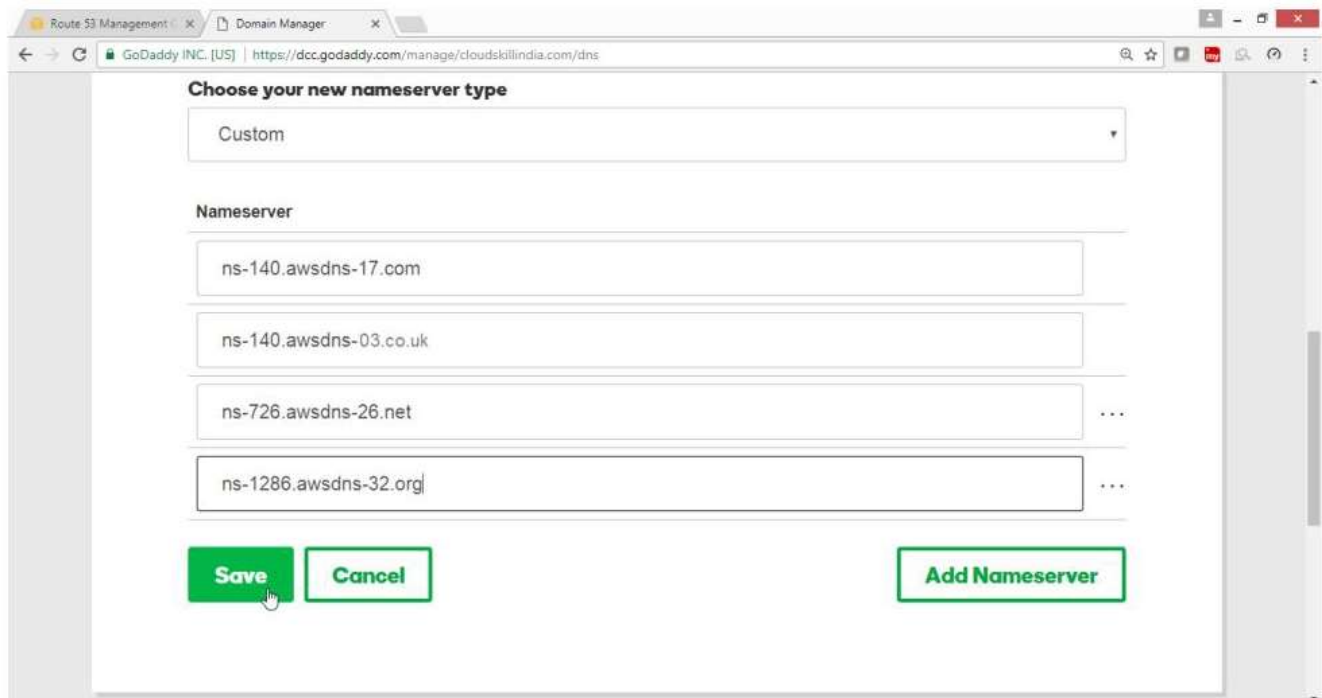
Nameserver

- ns-652.awsdns-17.net
- ns-1796.awsdns-32.co.uk
- ns-1378.awsdns-44.org
- ns-351.awsdns-43.com

For choose your new name server -> [Custom](#)

Replace old NS records with latest NS records

Click on [Save](#) button



The screenshot shows the 'Choose your new nameserver type' form in the GoDaddy Domain Manager. The 'Custom' option is selected in the dropdown menu. Below, there are four text input fields for nameservers, each followed by an ellipsis (...) indicating more options. The nameservers entered are ns-140.awsdns-17.com, ns-140.awsdns-03.co.uk, ns-726.awsdns-26.net, and ns-1286.awsdns-32.org. At the bottom, there are three buttons: 'Save', 'Cancel', and 'Add Nameserver'. A mouse cursor is hovering over the 'Save' button.

Choose your new nameserver type

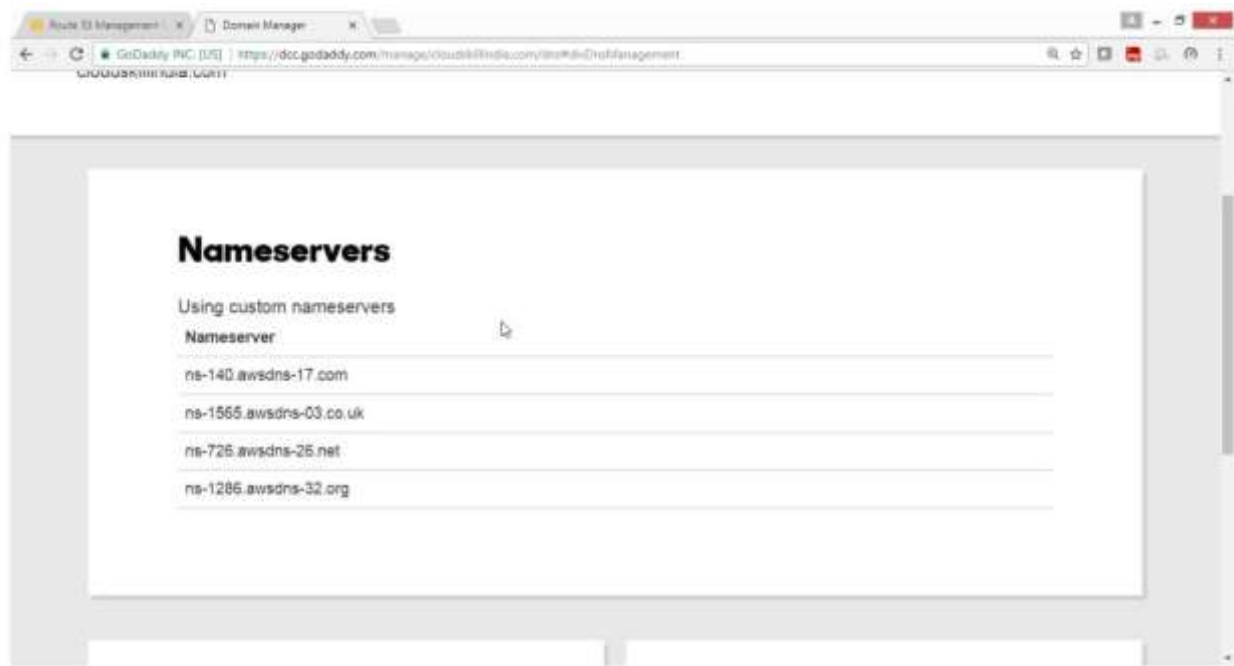
Custom

Nameserver

- ns-140.awsdns-17.com
- ns-140.awsdns-03.co.uk
- ns-726.awsdns-26.net
- ns-1286.awsdns-32.org

[Save](#) [Cancel](#) [Add Nameserver](#)

[Verify New names](#) got updated

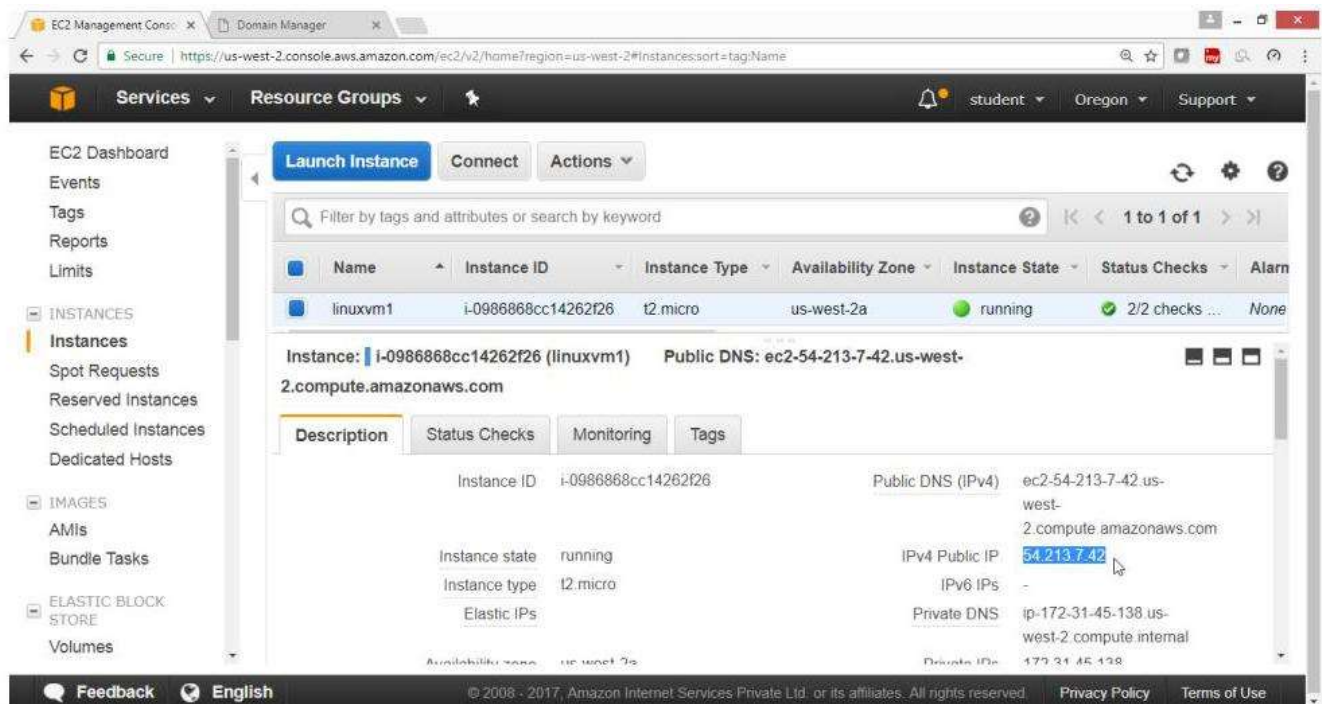


Step-3 Launch an instance configure it as a Webserver

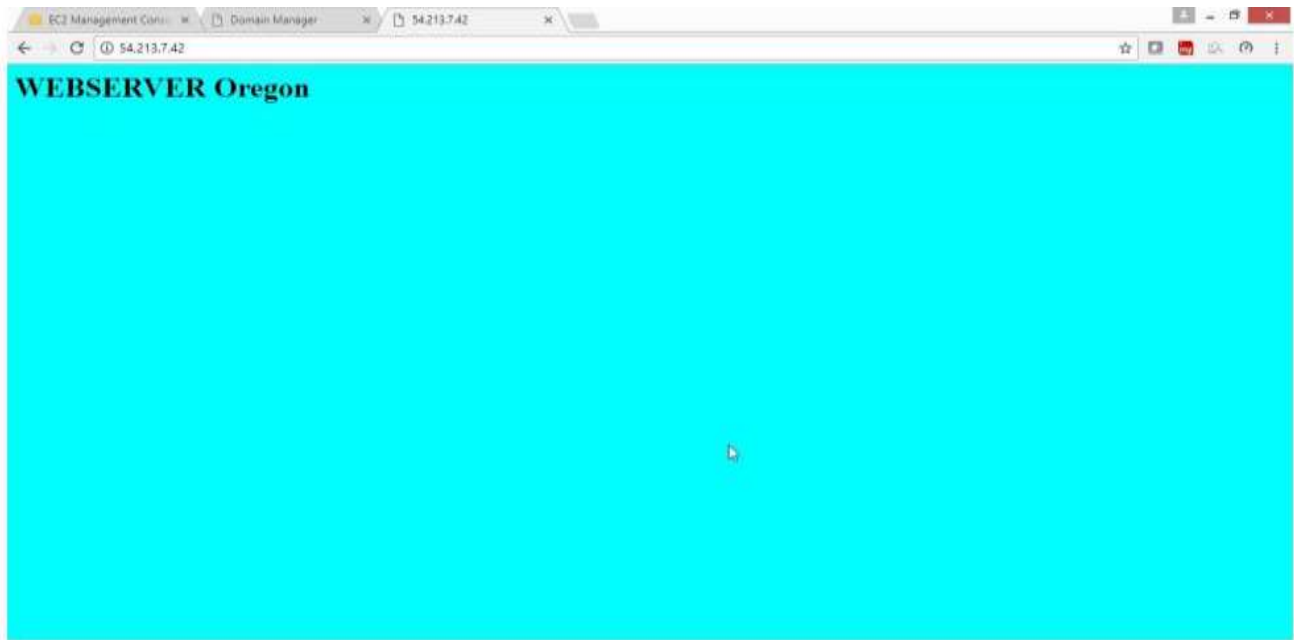
- Launch an Amazon Linux Instance
- Configure it as a Web Server

Note: Repeat LAB Hosting webserver on Linux

Copy the public IP and type in browser



Verify Website is accessible

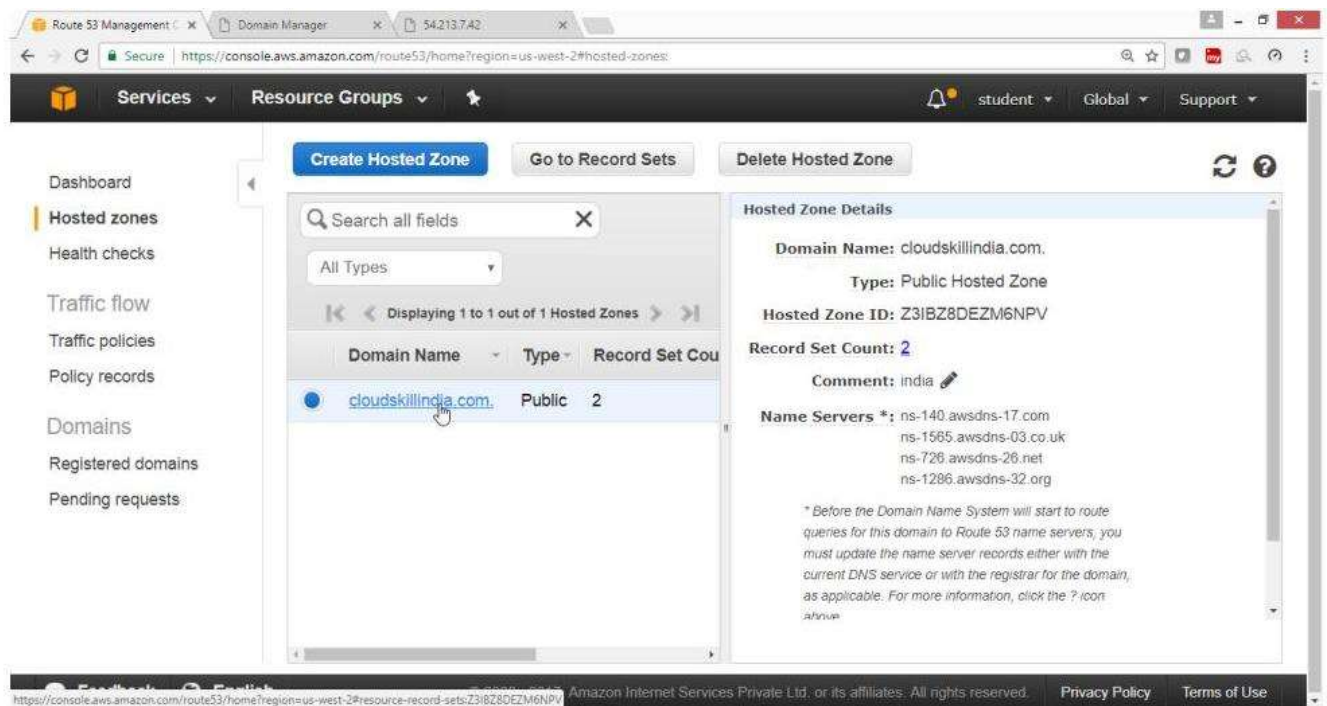


Step-4 To add a "A" record and CNAME record in Route53 From Route53 Dashboard

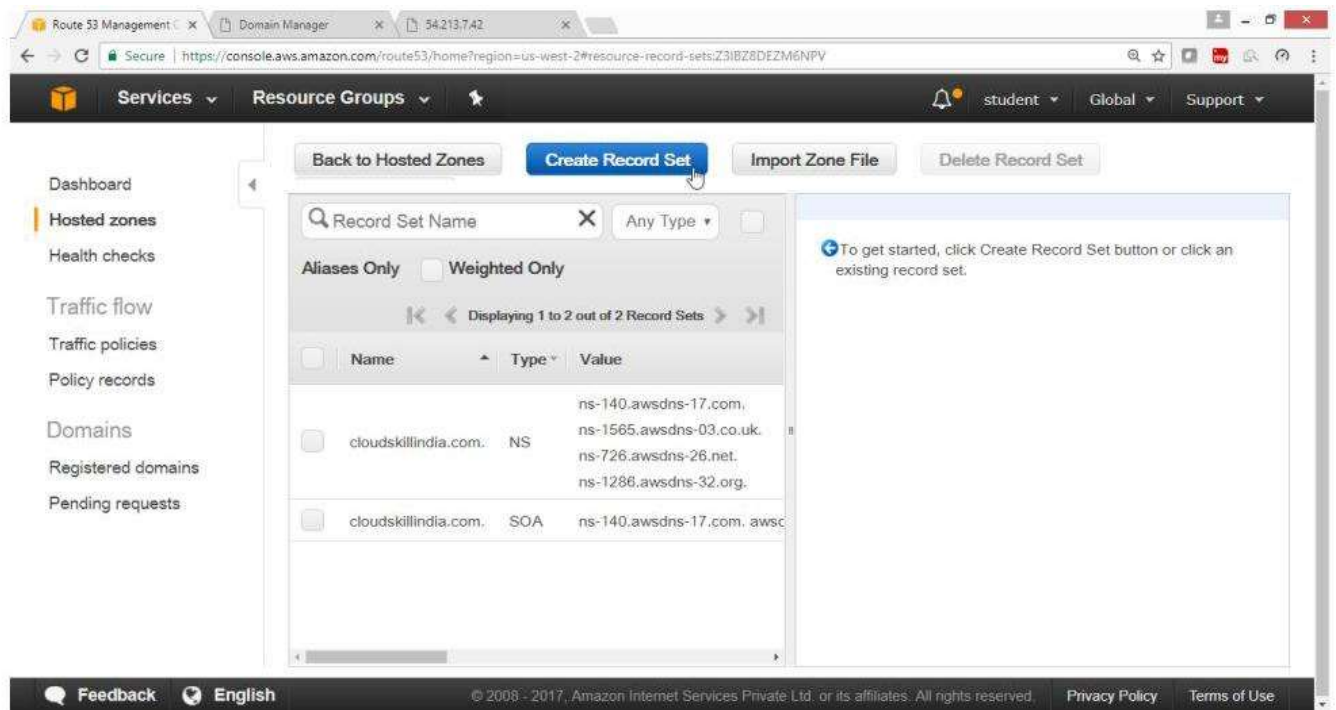
Click on "Hosted Zones"

Select Domain Name

Click on "cloudskillindia.com"



Click on Create Record set button



To add A record

On right side under Create Record Set

Provide the following values

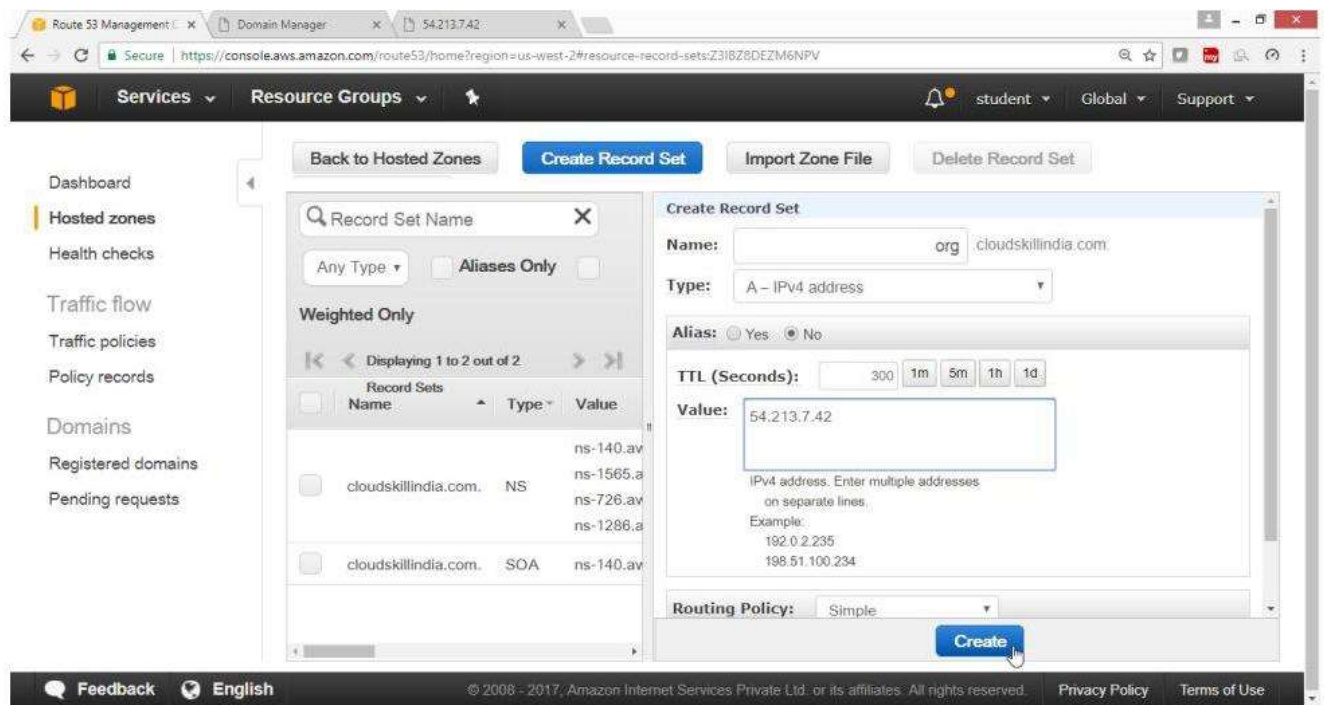
NAME-> org.cloudskillindia.com

Type-> A-IPV4 address

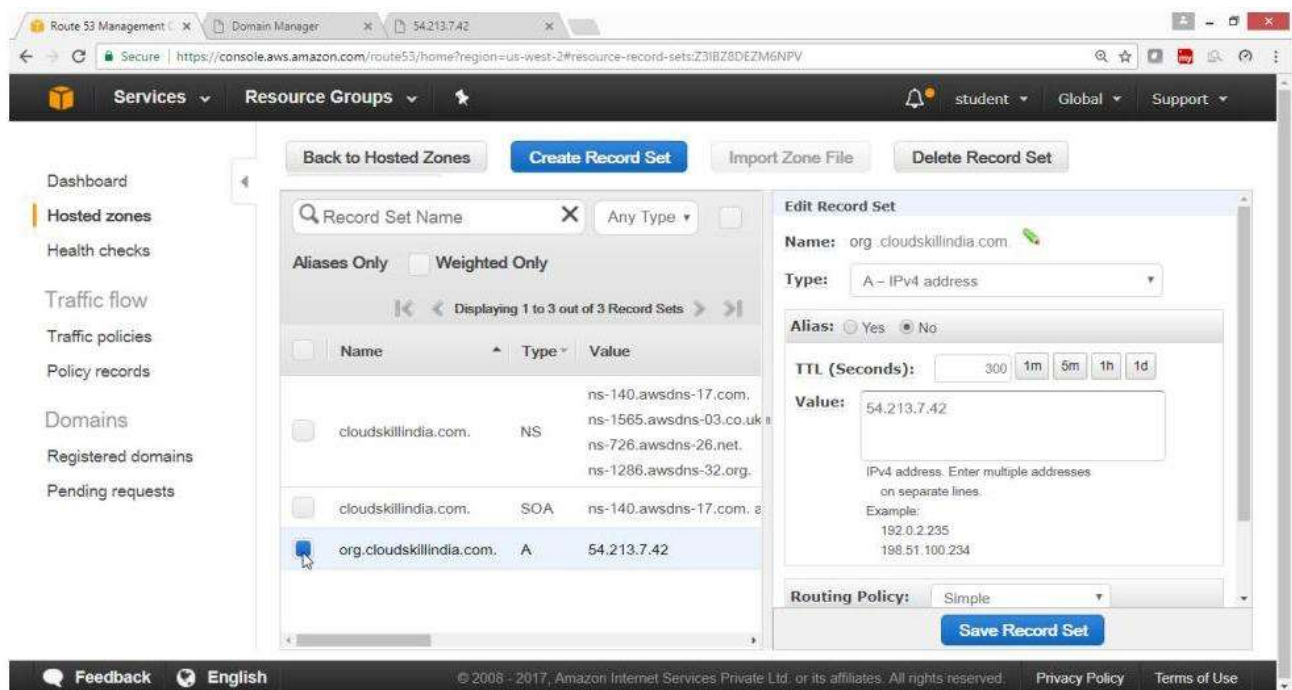
Alias ->No

Value=>54.213.7.42 [Give your Instance Public IP]

Click on "Create" button



Verify the A record got created



Create Alias record

The screenshot shows the AWS Route 53 console with the 'Create Record Set' dialog open. The dialog is for creating an Alias record for the domain 'www.cloudskillindia.com'. The 'Name' field is 'www.cloudskillindia.com', the 'Type' is 'CNAME - Canonical name', and the 'Value' is 'org.cloudskillindia.com'. The 'Alias' checkbox is checked, and the 'TTL (Seconds)' is set to 300. The 'Routing Policy' is set to 'Simple'. The 'Create' button is highlighted.

Record Set Name: Any Type

Aliases Only ☐ Weighted Only ☐

Displaying 1 to 3 out of 3 Record Sets

Name	Type	Value
cloudskillindia.com.	NS	ns-140.awsdns-17.com. ns-1565.awsdns-03.co.uk. ns-726.awsdns-26.net. ns-1286.awsdns-32.org.
cloudskillindia.com.	SOA	ns-140.awsdns-17.com.
org.cloudskillindia.com.	A	54.213.7.42

Create Record Set

Name: www.cloudskillindia.com

Type: CNAME - Canonical name

Alias: ☒ Yes ☐ No

TTL (Seconds): 300 1m 5m 1h 1d

Value: org.cloudskillindia.com

The domain name that you want to resolve to instead of the value in the Name field.
Example: www.example.com

Routing Policy: Simple

Create

Verify the CNAME record got created

The screenshot shows the AWS Route 53 console with the 'Record Set Name' search bar. The 'Create Record Set' button is highlighted. The table below shows the list of record sets, including the newly created CNAME record for 'www.cloudskillindia.com' pointing to 'org.cloudskillindia.com'.

Record Set Name: Any Type

Aliases Only ☐ Weighted Only ☐

Displaying 1 to 4 out of 4 Record Sets

Name	Type	Value
cloudskillindia.com.	NS	ns-140.awsdns-17.com. ns-1565.awsdns-03.co.uk. ns-726.awsdns-26.net. ns-1286.awsdns-32.org.
cloudskillindia.com.	SOA	ns-140.awsdns-17.com.
org.cloudskillindia.com.	A	54.213.7.42
www.cloudskillindia.com.	CNAME	org.cloudskillindia.com.

To get started, click Create Record Set button or click an existing record set.

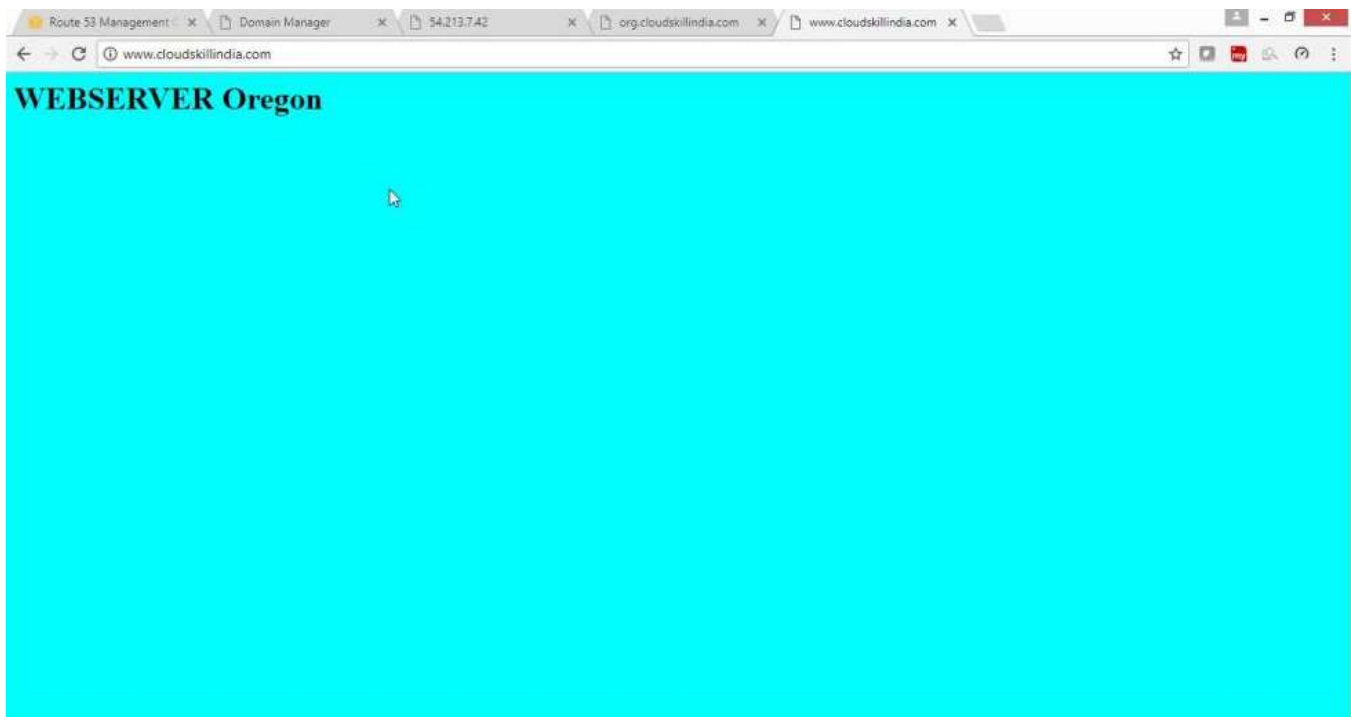
Verification

Now access the website with A record -> org.cloudskillindia.com



Verification

Now access the website with CNAME record -> org.cloudskillindia.com



What is Route 53? What are its Features?

Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service.

It is designed to give developers and businesses an extremely reliable and cost-effective way to route end users to Internet applications by translating names like `www.example.com` into the numeric IP addresses like `192.0.2.1` that computers use to connect to each other. Amazon Route 53 is fully compliant with IPv6 as well.

Amazon Route 53 effectively connects user requests to infrastructure running in AWS – such as Amazon EC2 instances, Elastic Load Balancing load balancers, or Amazon S3 buckets

Features:

Traffic Flow

Easy-to-use and cost-effective global traffic management: route end users to the best endpoint for your application based on latency, health, and other considerations.

Latency Based Routing

Route end users to the AWS region that provides the lowest possible latency.

Geo DNS

Route end users to a particular endpoint that you specify based on the end user's geographic location.

Private DNS for Amazon VPC

Manage custom domain names for your internal AWS resources without exposing DNS data to the public Internet.

DNS Failover

Automatically route your website visitors to an alternate location to avoid site outages.

Health Checks and Monitoring

Amazon Route 53 can monitor the health and performance of your application as well as your web servers and other resources.

Domain Registration

Amazon Route 53 offers domain name registration services, where you can search for and register available domain names or transfer in existing domain names to be managed by Route 53.

Weighted Round Robin

Amazon Route 53 offers Weighted Round Robin (WRR) functionality.

Amazon ELB Integration

Amazon Route 53 is integrated with Elastic Load Balancing (ELB).

Management Console

Amazon Route 53 works with the AWS Management Console. This web-based, point-and-click, graphical user interface lets you manage Amazon Route 53 without writing any code at all.

What are the different routing policies available in Route 53?

Route 53 provides multiple options for creating a Routing policy. Some of these options are as follows:

Simple Routing: In this option, Route 53 will respond to DNS queries based on the values in resource record set.

Weighted Routing: In this policy, we can specify the weightage according to which multiple resources will handle the load. E.g. If we have two web servers, we can divide load in 40/60 ratio on these servers.

Latency Routing: In this option, Route 53 will respond to DNS queries with the resources that provide the best latency.

Failover Routing: We can configure active/passive failover by using this policy. One resource will get all the traffic when it is up. Once first resource is down, all the traffic will be routed to second resource that is active during failover.

Geolocation Routing: As the name suggests, this policy works on the basis of location of end users from where requests originate.

You have an EC2 Security Group with several running EC2 instances. You changed the Security Group rules to allow inbound traffic on a new port and protocol, and then launched several new instances in the same Security Group. The new rules apply:

- A. Immediately to all instances in the security group.
- B. Immediately to the new instances only.
- C. Immediately to the new instances, but old instances must be stopped and restarted before the new rules apply.
- D. To all instances, but it may take several minutes for old instances to see the changes.

Answer A.

Explanation: Any rule specified in an EC2 Security Group applies immediately to all the instances, irrespective of when they are launched before or after adding a rule.

To create a mirror image of your environment in another region for disaster recovery, which of the following AWS resources do not need to be recreated in the second region? (Choose 2 answers)

- A. Route 53 Record Sets
- B. Elastic IP Addresses (EIP)
- C. EC2 Key Pairs
- D. Launch configurations
- E. Security Groups

Answer A,B.

Explanation: Elastic IPs and Route 53 record sets are common assets therefore there is no need to replicate them, since Elastic IPs and Route 53 are valid across regions

A customer wants to capture all client connection information from his load balancer at an interval of 5 minutes, which of the following options should he choose for his application?

- A. Enable AWS CloudTrail for the loadbalancer.
- B. Enable access logs on the load balancer.
- C. Install the Amazon CloudWatch Logs agent on the load balancer.
- D. Enable Amazon CloudWatch metrics on the load balancer.

Answer A

Explanation: AWS CloudTrail provides inexpensive logging information for load balancer and other AWS resources This logging information can be used for analyses and other administrative work, therefore is perfect for this use case.

A customer wants to track access to their Amazon Simple Storage Service (S3) buckets and also use this information for their internal security and access audits. Which of the following will meet the Customer requirement?

- A. Enable AWS CloudTrail to audit all Amazon S3 bucket access.
- B. Enable server access logging for all required Amazon S3 buckets.
- C. Enable the Requester Pays option to track access via AWS Billing
- D. Enable Amazon S3 event notifications for Put and Post.

Answer A

Explanation: AWS CloudTrail has been designed for logging and tracking API calls. Also this service is available for storage, therefore should be used in this use case.

Which of the following are true regarding AWS CloudTrail? (Choose 2 answers)

- A. CloudTrail is enabled globally

- B. CloudTrail is enabled on a per-region and service basis
- C. Logs can be delivered to a single Amazon S3 bucket for aggregation.
- D. CloudTrail is enabled for all available services within a region.

Answer B, C

Explanation: Cloudtrail is not enabled for all the services and is also not available for all the regions. Therefore, option B is correct, also the logs can be delivered to your S3 bucket, hence C is also correct.

What happens if CloudTrail is turned on for my account but my Amazon S3 bucket is not configured with the correct policy?

CloudTrail files are delivered according to S3 bucket policies. If the bucket is not configured or is misconfigured, CloudTrail might not be able to deliver the log files.

How do I transfer my existing domain name registration to Amazon Route 53 without disrupting my existing web traffic?

You will need to get a list of the DNS record data for your domain name first, it is generally available in the form of a “zone file” that you can get from your existing DNS provider.

Once you receive the DNS record data, you can use Route 53’s Management Console or simple web-services interface to create a hosted zone that will store your DNS records for your domain name and follow its transfer process.

It also includes steps such as updating the nameservers for your domain name to the ones associated with your hosted zone. For completing the process, you have to contact the registrar with whom you registered your domain name and follow the transfer process. As soon as your registrar propagates the new name server delegations, your DNS queries will start to get answered.



AWS Load Balancing

Elastic Load Balancing Highlights

Elastic Load Balancing distributes incoming application traffic across multiple EC2 instances, in multiple Availability Zones. This increases the fault tolerance of your applications.

The load balancer serves as a single point of contact for clients, which increases the availability of your application. You can add and remove instances from your load balancer as your needs change, without disrupting the overall flow of requests to your application. Elastic Load Balancing scales your load balancer as traffic to your application changes over time and can scale to the vast majority of workloads automatically.

You can configure health checks, which are used to monitor the health of the registered instances so that the load balancer can send requests only to the healthy instances. You can also offload the work of encryption and decryption to your load balancer so that your instances can focus on their main work.

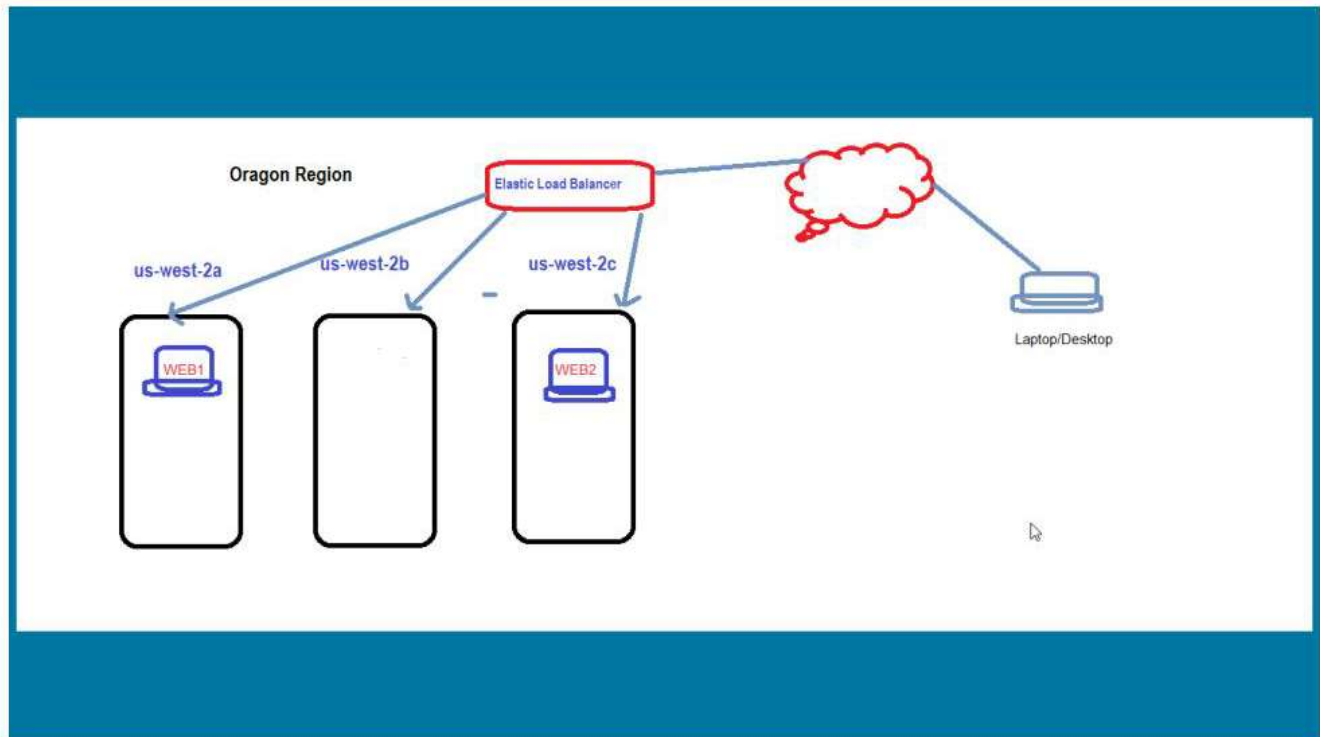
When should we use a Classic Load Balancer vs. an Application load balancer?

A Classic Load Balancer is used for simple load balancing of traffic across multiple EC2 instances.

An Application Load Balancer is more suited for Microservices based architecture or container-based architecture. Mainly in these architectures there is a need to do load balancing as well as there is need to route traffic to multiple services on same EC2 instance.

Share the Load Balancer Configuration Step by Step?

To Configure Elastic Load Balancer in AWS



Step 1: Select a Load Balancer Type

Elastic Load Balancing supports two types of load balancers: Application Load Balancers and Classic Load Balancers. For this tutorial, you create an Application Load Balancer.

To create an Application Load Balancer

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. On the navigation bar, choose a region for your load balancer. Be sure to select the same region that you used for your EC2 instances.
3. On the navigation pane, under **LOAD BALANCING**, choose **Load Balancers**.
4. Choose **Create Load Balancer**.
5. Choose **Application Load Balancer**, and then choose **Continue**.

Step 2: Configure Your Load Balancer and Listener

On the **Configure Load Balancer** page, complete the following procedure.

To configure your load balancer and listener

1. For **Name**, type a name for your load balancer.

The name of your Application Load Balancer must be unique within your set of Application Load Balancers for the region, can have a maximum of 32 characters, can contain only alphanumeric characters and hyphens, and must not begin or end with a hyphen.

2. For **Scheme**, keep the default value, **internet-facing**.

Basic Configuration

To configure your load balancer, provide a name, select a scheme, specify one or more listeners, and select a network. The default configuration is an Internet-facing load balancer in the selected network with a listener that receives HTTP traffic on port 80.

Name ⓘ	<input type="text" value="my-load-balancer"/>
Scheme ⓘ	<input checked="" type="radio"/> internet-facing <input type="radio"/> internal

3. For **IP address type**, select **ipv4** if your instances support IPv4 addresses or **dualstack** if they support IPv4 and IPv6 addresses.

4. For **Listeners**, keep the default, which is a listener that accepts HTTP traffic on port 80.

Listeners

A listener is a process that checks for connection requests, using the protocol and port that you configured.

Load Balancer Protocol	Load Balancer Port	
<input type="text" value="HTTP"/>	<input type="text" value="80"/>	<input type="button" value="X"/>
<input type="button" value="Add listener"/>		

5. For **Availability Zones**, select the VPC that you used for your EC2 instances. For each of the two Availability Zones that contain your EC2 instances, select the Availability Zone and then select the public subnet for that Availability Zone.

6. Choose **Next: Configure Security Settings**.

7. For this tutorial, you are not using a secure listener. Choose **Next: Configure Security Groups**.

Step 3: Configure a Security Group for Your Load Balancer

The security group for your load balancer must allow it to communicate with registered targets on both the listener port and the health check port. The console can create security groups for your load balancer on your behalf, with rules that specify the correct protocols and ports.

Note:

If you prefer, you can create and select your own security group instead. For more information, see [Recommended Rules](#) in the *Application Load Balancer Guide*.

On the [Configure Security Groups](#) page, complete the following procedure to have Elastic Load Balancing create a security group for your load balancer on your behalf.

To configure a security group for your load balancer

1. Choose [Create a new security group](#).

2. Type a name and description for the security group, or keep the default name and description. This new security group contains a rule that allows traffic to the load balancer listener port that you selected on the [Configure Load Balancer](#) page.

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

Security group name:

Description:

Type <small>i</small>	Protocol <small>i</small>	Port Range <small>i</small>	Source <small>i</small>	
HTTP ▾	TCP	80	Anywhere ▾	0.0.0.0/0 ✕

3. Choose [Next: Configure Routing](#).

Step 4: Configure Your Target Group

Create a target group, which is used in request routing. The default rule for your listener routes requests to the registered targets in this target group. The load balancer checks the health of targets in this target group using the health check settings defined for the target group. On the [Configure Routing](#) page, complete the following procedure.

To configure your target group

1. For [Target group](#), keep the default, [New target group](#).

2. For [Name](#), type a name for the new target group.



3. Keep [Protocol](#) as HTTP and [Port](#) as 80.

Target group

Target group		<input type="text" value="New target group"/>
Name		<input type="text" value="my-targets"/>
Protocol		<input type="text" value="HTTP"/>
Port		<input type="text" value="80"/>

4. For [Health checks](#), keep the default protocol and ping path.

Health checks

Protocol		<input type="text" value="HTTP"/>
Path		<input type="text" value="/"/>

5. Choose [Next: Register Targets](#).

Step 5: Register Targets with Your Target Group




On the **Register Targets** page, complete the following procedure.

To register targets with the target group

1. For [Instances](#), select one or more instances.
2. Keep the default port, 80, and choose [Add to registered](#).

Instances

To register additional instances, select one or more running instances, specify a port, and then click Add. The default port is the port specified for the target group. If the instance is already registered on the specified port, you must specify a different port.

<input type="button" value="Add to registered"/>	on port <input type="text" value="80"/>						
<input type="text" value="Search Instances"/> 							
<input type="checkbox"/>	Instance	Name	State	Security groups	Zone	Subnet ID	Subnet CIDR
<input checked="" type="checkbox"/>	i-23a490a6	Server1	 running	my-security-group	us-west-2a	subnet-65ea5f08	10.0.0.0/24
<input checked="" type="checkbox"/>	i-ee7fe276	Server2	 running	my-security-group	us-west-2b	subnet-7ad90a22	10.0.2.0/24

3. If you need to remove an instance that you selected, for [Registered instances](#), select the instance and then choose [Remove](#).
4. When you have finished selecting instances, choose [Next: Review](#).

Step 6: Create and Test Your Load Balancer

Before creating the load balancer, review the settings that you selected. After creating the load balancer, verify that it's sending traffic to your EC2 instances.

To create and test your load balancer

1. On the [Review](#) page, choose [Create](#).
2. After you are notified that your load balancer was created successfully, choose [Close](#).
3. On the navigation pane, under [LOAD BALANCING](#), choose [Target Groups](#).
4. Select the newly created target group.
5. On the [Targets](#) tab, verify that your instances are ready. If the status of an instance is initial, it's probably because the instance is still in the process of being registered, or it has not passed the minimum number of health checks to be considered healthy. After the status of at least one instance is healthy, you can test your load balancer.
6. On the navigation pane, under [LOAD BALANCING](#), choose [Load Balancers](#).
7. On the **Description** tab, copy the DNS name of the load balancer (for example, my-loadbalancer-1234567890.us-west-2.elb.amazonaws.com). Paste the DNS name into the address field of an Internet-connected web browser. If everything is working, the browser displays the default page of your server.

Step 7: Delete Your Load Balancer (Optional)

As soon as your load balancer becomes available, you are billed for each hour or partial hour that you keep it running. When you no longer need a load balancer, you can delete it. As soon as the load balancer is deleted, you stop incurring charges for it. Note that deleting a load balancer does not affect the targets registered with the load balancer. For example, your EC2 instances continue to run.

To delete your load balancer

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. On the navigation pane, under [LOAD BALANCING](#), choose [Load Balancers](#).
3. Select the load balancer, and then choose [Actions, Delete](#).
4. When prompted for confirmation, choose [Yes, Delete](#).

How many subnets are needed for the Application Load Balancers?

You will need at least 2 public subnets in order to deploy an application load balancer

Explain how the buffer is used in Amazon web services?

The buffer is used to make the system more robust to manage traffic or load by synchronizing different component. Usually, components receive and process the requests in an unbalanced way, With the help of buffer, the components will be balanced and will work at the same speed to provide faster services.

Suppose you have an application where you have to render images and also do some general computing. From the following services which service will best fit your need?

- A. Classic Load Balancer
- B. Application Load Balancer**
- C. Both of them
- D. None of these

Answer B

Explanation: You will choose an application load balancer, since it supports path-based routing, which means it can take decisions based on the URL, therefore if your task needs image rendering it will route it to a different instance, and for general computing it will route it to a different instance.

What is the difference between Scalability and Elasticity?

Scalability is the ability of a system to increase its hardware resources to handle the increase in demand. It can be done by increasing the hardware specifications or increasing the processing nodes.

Elasticity is the ability of a system to handle increase in the workload by adding additional hardware resources when the demand increases (same as scaling) but also rolling back the scaled resources, when the resources are no longer needed. This is particularly helpful in Cloud environments, where a pay per use model is followed.

How will you change the instance type for instances which are running in your application tier and are using Auto Scaling. Where will you change it from the following areas?

- A. Auto Scaling policy configuration
- B. Auto Scaling group
- C. Auto Scaling tags configuration
- D. Auto Scaling launch configuration

Answer D

Explanation: Auto scaling tags configuration, is used to attach metadata to your instances, to change the instance type you have to use auto scaling launch configuration.

You have a content management system running on an Amazon EC2 instance that is approaching 100% CPU utilization. Which option will reduce load on the Amazon EC2 instance?

- A. Create a load balancer, and register the Amazon EC2 instance with it
- B. Create a CloudFront distribution, and configure the Amazon EC2 instance as the origin
- C. Create an Auto Scaling group from the instance using the CreateAutoScalingGroup action
- D. Create a launch configuration from the instance using the CreateLaunchConfigurationAction

Answer A

Explanation: Creating alone an autoscaling group will not solve the issue, until you attach a load balancer to it. Once you attach a load balancer to an autoscaling group, it will efficiently distribute the load among all the instances. Option B – CloudFront is a CDN, it is a data transfer tool therefore will not help reduce load on the EC2 instance. Similarly, the other option – Launch configuration is a template for configuration which has no connection with reducing loads.

When should I use a Classic Load Balancer and when should I use an Application load balancer?

A Classic Load Balancer is ideal for simple load balancing of traffic across multiple EC2 instances, while an Application Load Balancer is ideal for microservices or container-based architectures where there is a need to route traffic to multiple services or load balance across multiple ports on the same EC2 instance.

What does Connection draining do?

- A. Terminates instances which are not in use.
 - B. Re-routes traffic from instances which are to be updated or failed a health check.
 - C. Re-routes traffic from instances which have more workload to instances which have less workload.
- Drains all the connections from an instance, with one click.

Answer B

Explanation: Connection draining is a service under ELB which constantly monitors the health of the instances. If any instance fails a health check or if any instance has to be patched with a software update, it pulls all the traffic from that instance and re-routes them to other instances.

When an instance is unhealthy, it is terminated and replaced with a new one, which of the following services does that?

- A. Sticky Sessions
- B. Fault Tolerance
- C. Connection Draining
- D. Monitoring

Answer B

Explanation: When ELB detects that an instance is unhealthy, it starts routing incoming traffic to other healthy instances in the region. If all the instances in a region becomes unhealthy, and if you have instances in some other availability zone/region, your traffic is directed to them. Once your instances become healthy again, they are re-routed back to the original instances.



Management Tools

AWS CloudWatch Monitor Resources and Applications	AWS Auto Scaling Scale Multiple Resources to Meet Demand	Amazon CloudFormation Create and Manage Resources with Templates
AWS CloudTrail Track User Activity and API Usage	AWS Config Track Resource Inventory & Changes	AWS OpsWorks Automate Operations with Chef & Puppet
AWS Service Catalog Create and Use Standardized Products	AWS System Manager Gain Operational Insights and Take Action	AWS Trusted Advisor Optimize Performance and Security
AWS Personal Health Dashboard Personalized View of AWS Service Health		



Management Tools

AWS CloudWatch

CloudWatch Highlights

Amazon CloudWatch is a monitoring service by Amazon for cloud-based AWS resources. Some of the main options in Amazon CloudWatch are as follows:

Logs: We can monitor and store logs generated by EC2 instances and our application in CloudWatch. We can store the log data for time period convenient for our use.

Dashboard: We can create visual Dashboards in the form of graphs to monitor our AWS resource in CloudWatch.

Alarms: We can set alarms in CloudWatch. These alarms can notify us by email or text when a specific metric crosses a threshold. These alarms can also detect the event when an Instance starts or shuts down.

Events: In CloudWatch we can also set up events that are triggered by an Alarm. These events can take an automated action when a specific Alarm is triggered.

CloudWatch is for performance Monitoring, CloudTrail is for auditing

Standard Monitoring = 5 Minutes | Detailed Monitoring = 1 Minute

Share the CloudWatch Configuration Step by Step?

To configure AWS CloudWatch to monitor CPU utilization

Topology



Pre-requisites

- User should have AWS account or IAM user with EC2 Full Access Policy

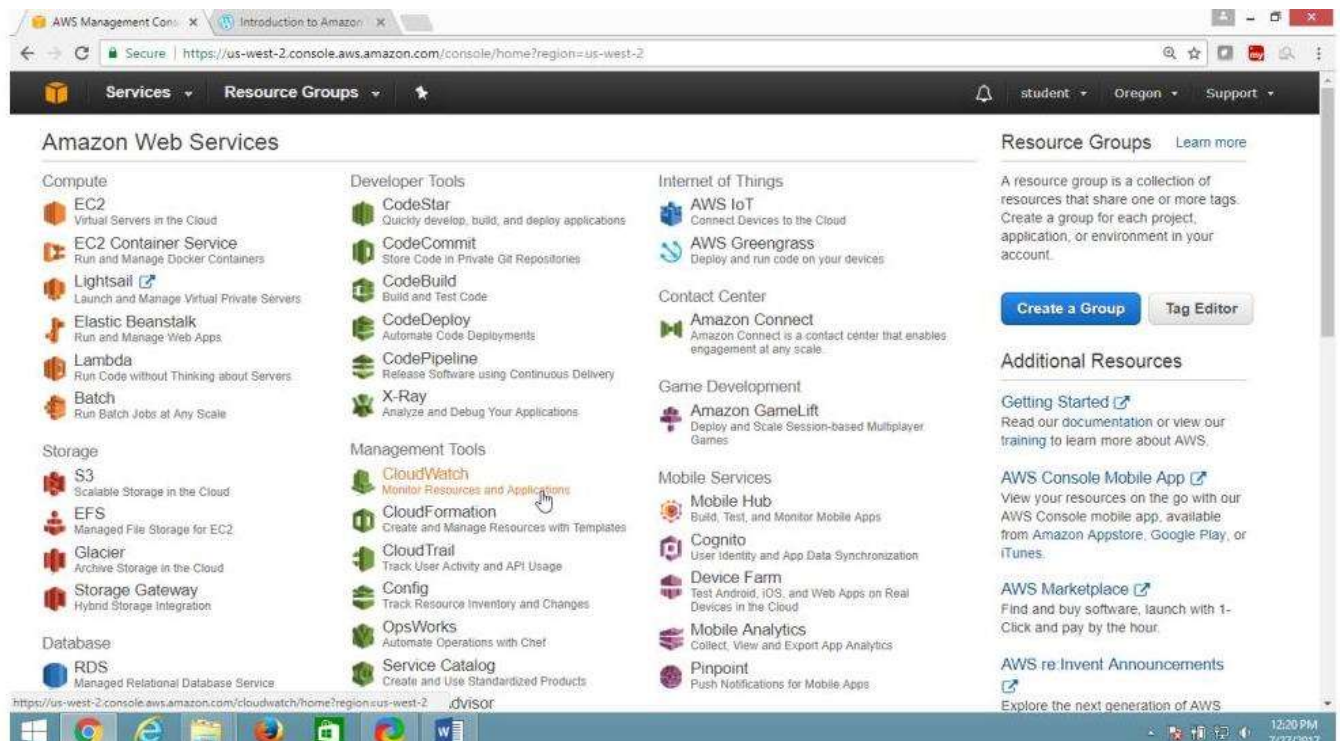
Task

- Creating Alarm
- Select Notification
- Check mail to Verify

Step-1) To Configure Amazon CloudWatch Service

Launch an Amazon Linux Instance, then

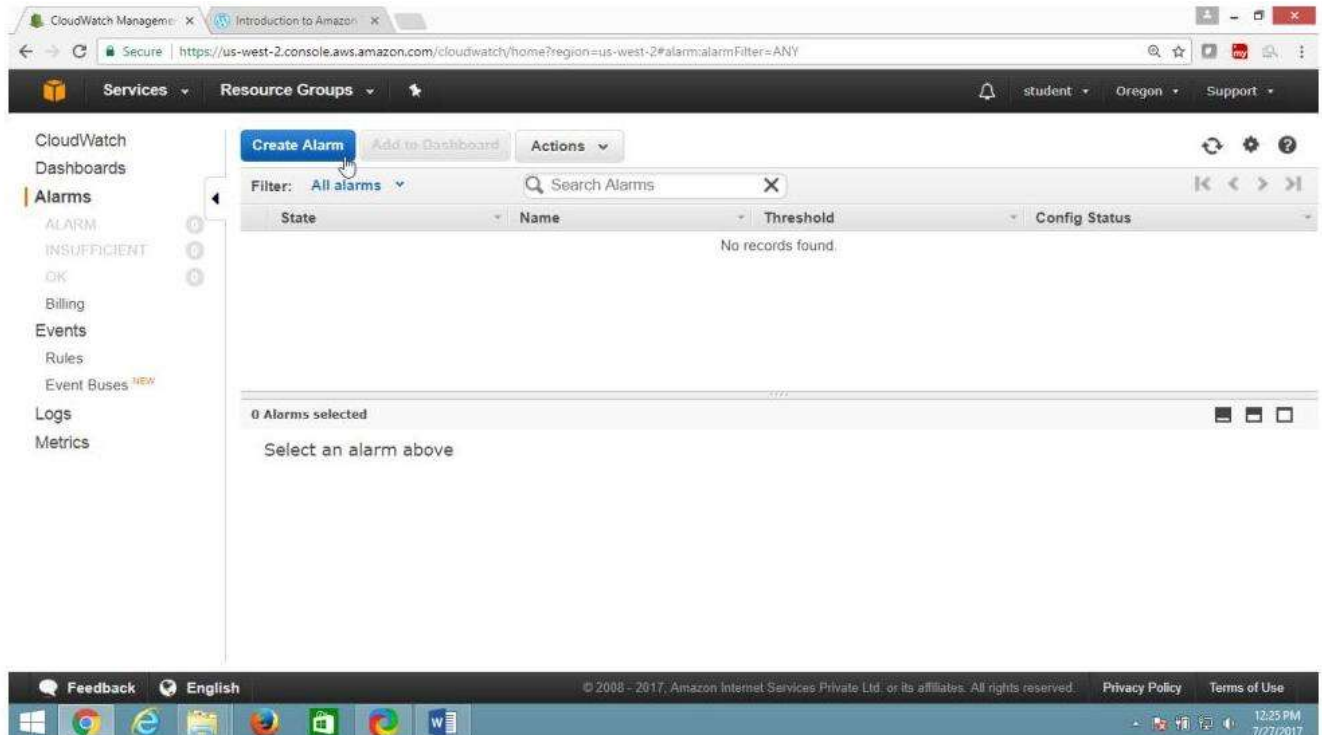
- Open AWS Console
- Click on Services
- In the Management Tools section
- Click on CloudWatch



On "CloudWatch", panel

Select **Alarms**

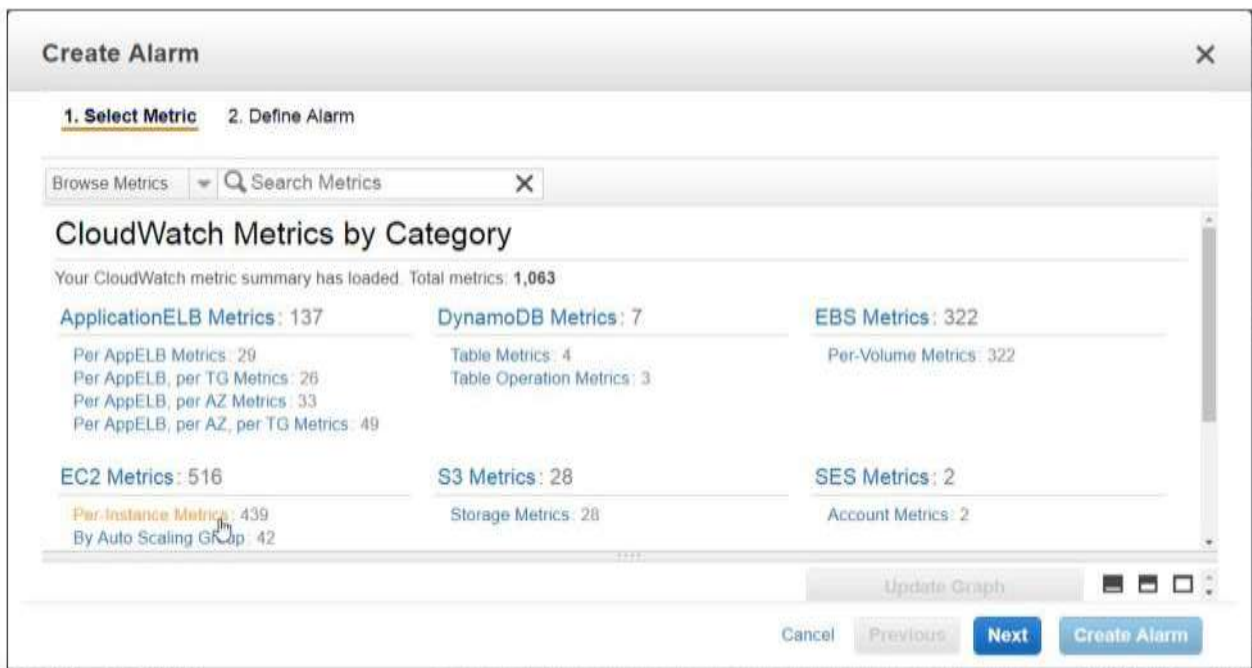
Click on "Create Alarm" button



In "Create Alarm" page

Select **"EC2 Metrics"**

Click on **"Per-instance Metrics"**



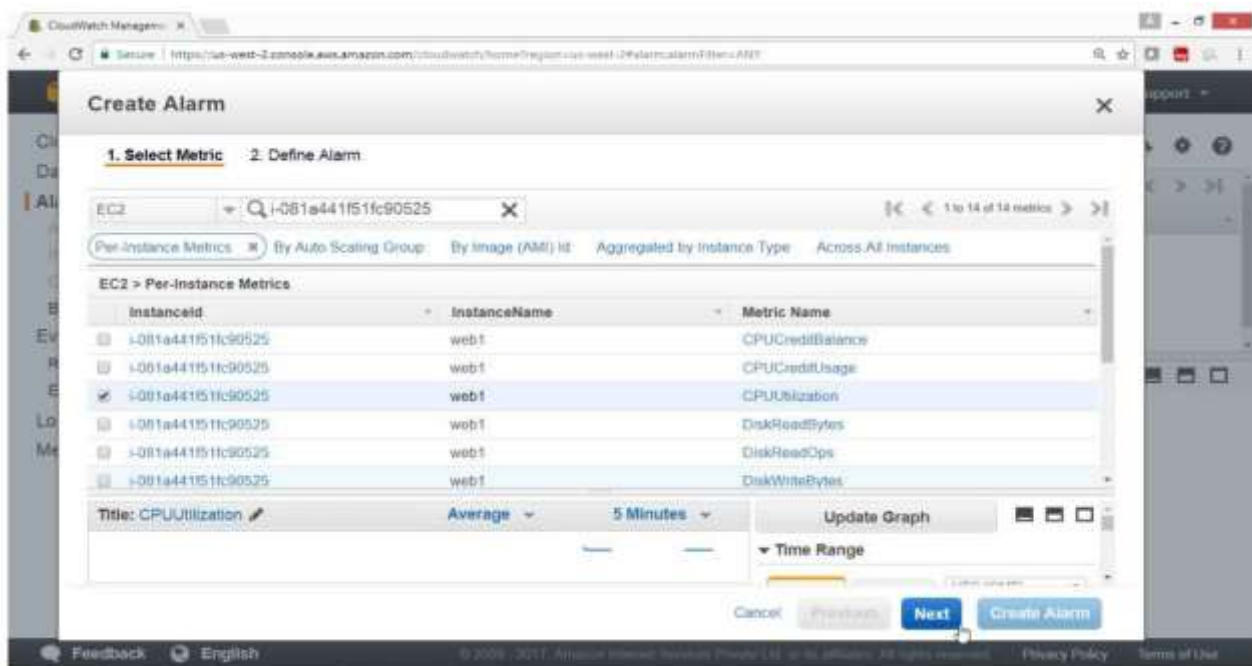
From "Create Alarm" page

Select "1. Select Metric"

In search box provide instance ID or Name

Under Metric Name, Select **CPU Utilization** checkbox

Click on **Next Button**



On **Create Alarm** Page

Select "2. Define Alarm"

Under Alarm Threshold

Name->testcpuutilization

Description->cputest

Under whenever CPUUtilization
is ≥ 30
for 1 consecutive periods

Drag Down

The screenshot shows the 'Create Alarm' wizard in the AWS CloudWatch console, specifically the '2. Define Alarm' step. The 'Alarm Threshold' section is active, showing the following configuration:

- Name:** testcpuutilization
- Description:** cputest1
- Whenever:** CPUUtilization
- is:** \geq 30
- for:** 1 consecutive period(s)

The 'Alarm Preview' section on the right shows a line graph titled 'CPUUtilization ≥ 0 '. The graph displays a blue line representing the metric value over time, with a red horizontal threshold line at 30. The text states: 'This alarm will trigger when the blue line goes up to or above the red line for a duration of 5 minutes'.

Below the graph, the following details are shown:

- Namespace:** AWS/EC2
- InstanceId:** i-081a441f51fc90525
- InstanceName:** web1
- Metric Name:** CPU Utilization

At the bottom of the form, there are buttons for 'Cancel', 'Previous', 'Next', and 'Create Alarm'. The footer of the console shows 'Feedback', 'English', and copyright information for Amazon Internet Services Private Ltd.

Under **Actions**

Whenever this alarm -> **State is Alarm**

Send notification to -> **Click on New list**

The screenshot shows the 'Create Alarm' wizard in the AWS CloudWatch console, specifically the '2. Define Alarm' step. The 'Additional settings' section on the right shows the alarm is configured for the 'AWS/EC2' namespace, instance 'web1', monitoring 'CPUUtilization' with a '5 Minutes' period and 'Standard' statistic. In the 'Actions' section, the trigger is set to 'Whenever this alarm: State is ALARM'. The 'Send notification to:' dropdown is open, showing 'Select a notification list'. A 'New list' link is highlighted next to the dropdown. At the bottom, there are buttons for '+ Notification', '+ AutoScaling Action', '+ EC2 Action', 'Cancel', 'Previous', 'Next', and 'Create Alarm'.

Send notification to -> **CPUTopicabc**

Email -> **adminabc@abc.com**

Click on "**Create Alarm**" button

This screenshot shows the same 'Create Alarm' wizard, but now the 'Send notification to:' dropdown is set to 'CPUTopicabc'. The 'Email list:' field below it is populated with 'adminabc@abc.com'. The 'Create Alarm' button at the bottom right is now highlighted in blue, indicating it is the next step in the process.

Click on "**I will do it Later**" button

Confirm new email addresses



Check your email inbox for a message with the subject "AWS Notification - Subscription Confirmation" and click the included link to confirm that you are willing to receive alerts to that address. AWS can only send notifications to confirmed addresses

Waiting for confirmation of 1 new email address

 adminabc@abc.com [Resend confirmation link](#)

Note: You have 72 hours to confirm these email addresses

I will do it later

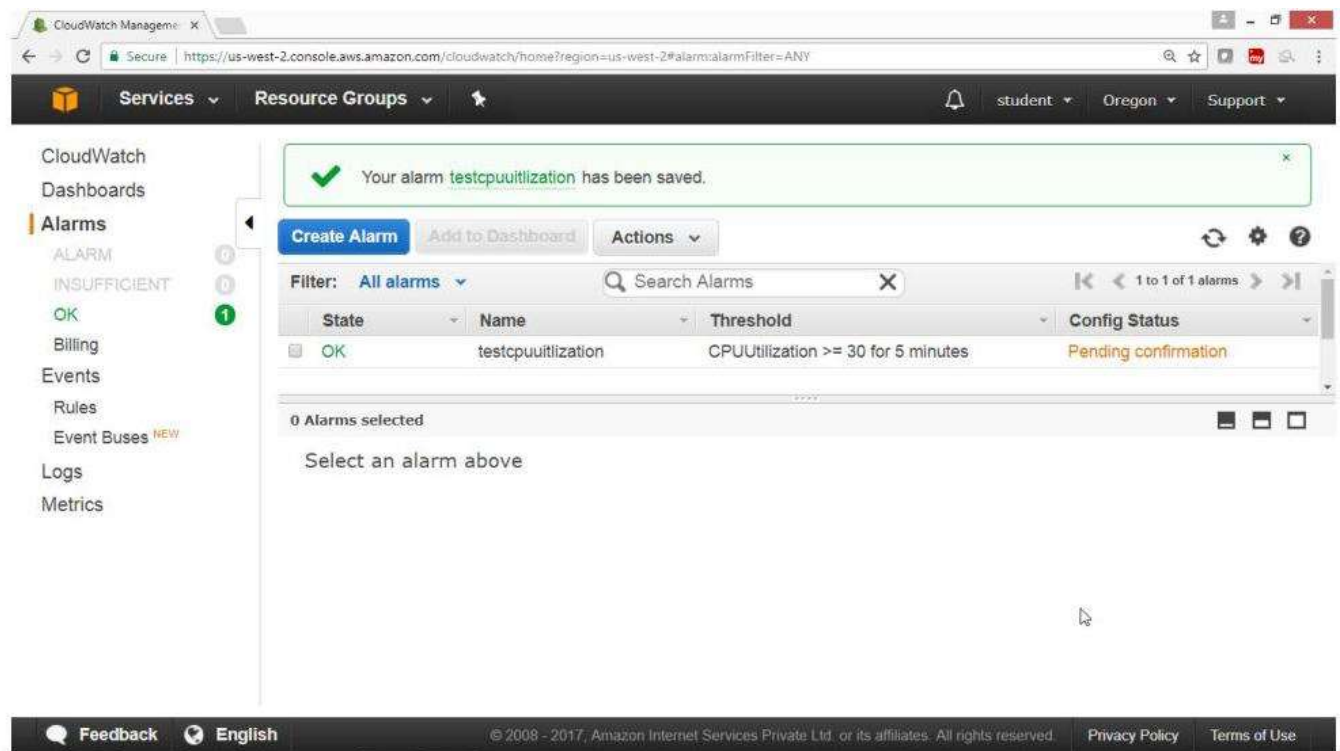
View Alarm

Go to your Email Account and check the Mail

Once mail is being checked

Config status -> [Pending Confirmation](#)

Verify the link from your Email

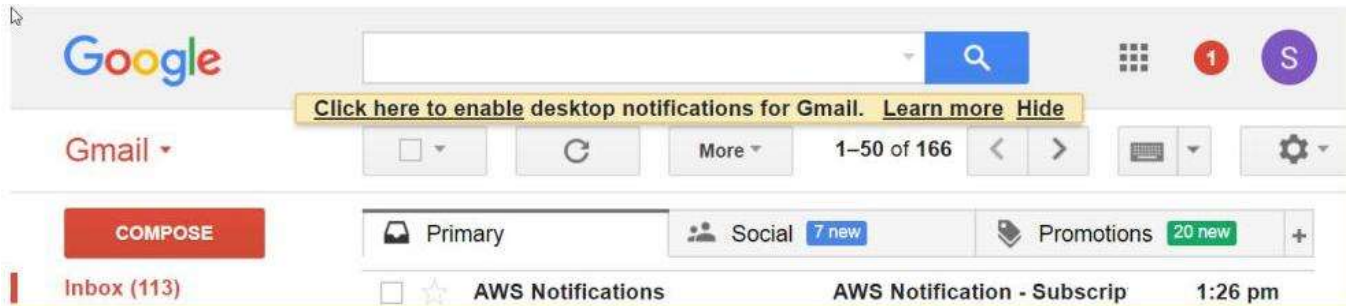


The screenshot shows the AWS CloudWatch console interface. At the top, a green notification banner states: "Your alarm testcpuutilization has been saved." Below this, the "Alarms" section is active, displaying a table with one alarm:

State	Name	Threshold	Config Status
OK	testcpuutilization	CPUUtilization >= 30 for 5 minutes	Pending confirmation

The "Config Status" for the alarm is "Pending confirmation". The left sidebar shows the navigation menu with "Alarms" selected. The bottom of the console includes a footer with "Feedback", "English", copyright information, and links to "Privacy Policy" and "Terms of Use".

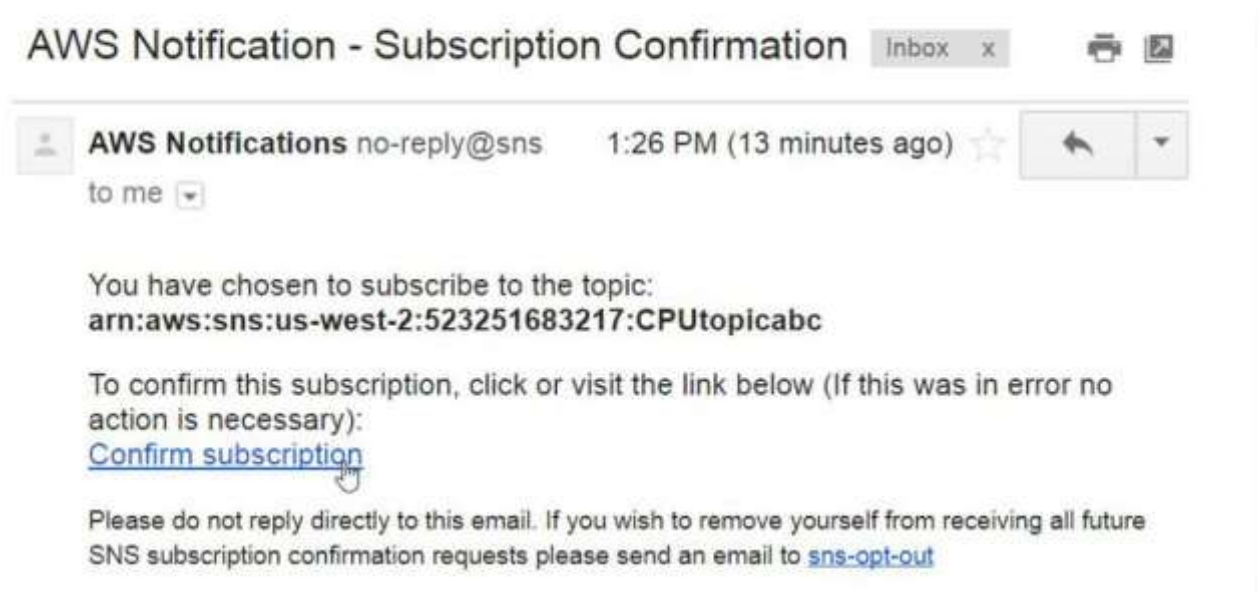
Open your email



Click on "Confirm Subscription"

Click on "Confirm subscription"

=====



=====

Verified by this output

Verified by this output

=====



Simple Notification Service

Subscription confirmed!

You have subscribed **adminabc@abc.com** to the topic:
CPUtopicabc.

Your subscription's id is:

arn:aws:sns:us-west-2:523251683217:CPUtopicabc:8e548f92-5474-4587-8105-64022c49ebf6

If it was not your intention to subscribe, [click here to unsubscribe](#).

After confirmation from email "Config Status" has become blank

=====

After confirmation from email Config status has become blank

The screenshot shows the AWS CloudWatch console interface. At the top, a green notification bar states: "Your alarm testcpuutilization has been saved." Below this, the "Alarms" section is active, displaying a table with one alarm:

State	Name	Threshold	Config Status
OK	testcpuutilization	CPUUtilization >= 30 for 5 minutes	

Below the table, it indicates "0 Alarms selected" and "Select an alarm above". The left sidebar shows navigation options: CloudWatch, Dashboards, Alarms (selected), Billing, Events, Rules, Event Buses, Logs, and Metrics. The bottom of the console features a footer with "Feedback", "English", copyright information, and links to "Privacy Policy" and "Terms of Use".

Now login to instance using mobaxterm


```
[2017-07-27 14:19.15] ~  
[shaikh.pc_mas] > cd e:awskeys
```

```
[2017-07-27 14:19.55] /drives/e/awskeys  
[shaikh.pc_mas] > ssh -i "25july2017masorg.pem" ec2-user@ec2-54-191-150-199.us-w  
est-2.compute.amazonaws.com
```

Switch to root user and install stress command

```
[ec2-user@ip-172-31-40-129 ~]$ sudo su  
[root@ip-172-31-40-129 ec2-user]# yum install stress -y
```

Login to another terminal-2

Run top command

```
[root@ip-172-31-40-129 ec2-user]# top
```

Verify Output

CPU status is 100% idle

```
top - 08:56:26 up 1:53, 2 users, load average: 0.00, 0.00, 0.00  
Tasks: 94 total, 1 running, 93 sleeping, 0 stopped, 0 zombie  
Cpu(s): 0.0%us, 0.0%sy, 0.0%ni, 100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st  
Mem: 1017372k total, 166080k used, 851292k free, 9224k buffers  
Swap: 0k total, 0k used, 0k free, 90380k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	root	20	0	19628	2420	2108	S	0.0	0.2	0:00.90	init
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	ksoftirqd/0
4	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0H
6	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kworker/u30:0
7	root	20	0	0	0	0	S	0.0	0.0	0:00.03	rcu_sched
8	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_bh
9	root	RT	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
10	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	lru-add-drain
11	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
12	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
13	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	netns
16	root	20	0	0	0	0	S	0.0	0.0	0:00.01	xenwatch
17	root	20	0	0	0	0	S	0.0	0.0	0:00.02	kworker/u30:2
21	root	20	0	0	0	0	S	0.0	0.0	0:00.00	xenbus
139	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khungtaskd
140	root	20	0	0	0	0	S	0.0	0.0	0:00.00	oom_reaper
141	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	writeback
143	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kcompactd0
144	root	25	5	0	0	0	S	0.0	0.0	0:00.00	ksmd
145	root	39	19	0	0	0	S	0.0	0.0	0:00.00	khugepaged
146	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	crypto
147	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kintegrityd

Run this command in terminal -1 which will increase the load

#stress --cpu40 --timeout 1000

```
[root@ip-172-31-40-129 ec2-user]# stress --cpu 40 --timeout 1000  
stress: info: [3095] dispatching hogs: 40 cpu, 0 io, 0 vm, 0 hdd
```


Now check the status in another terminal -2 by running top command

#top

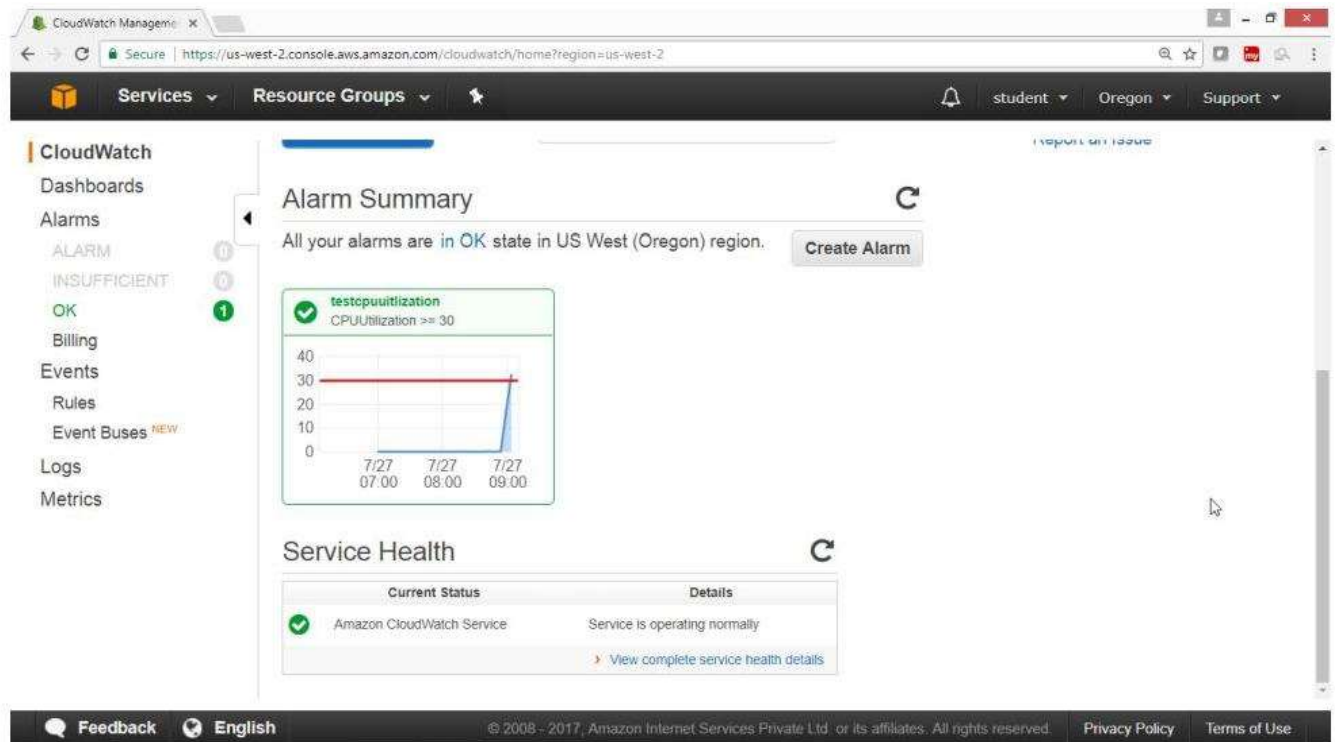
Verify the output

CPU load is 100%

```
top - 09:07:11 up 2:04, 3 users, load average: 16.16, 6.55, 2.88
Tasks: 144 total, 41 running, 103 sleeping, 0 stopped, 0 zombie
Cpu(s):100.0%us, 0.0%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 1017372k total, 179324k used, 838048k free, 9460k buffers
Swap: 0k total, 0k used, 0k free, 90760k cached
```

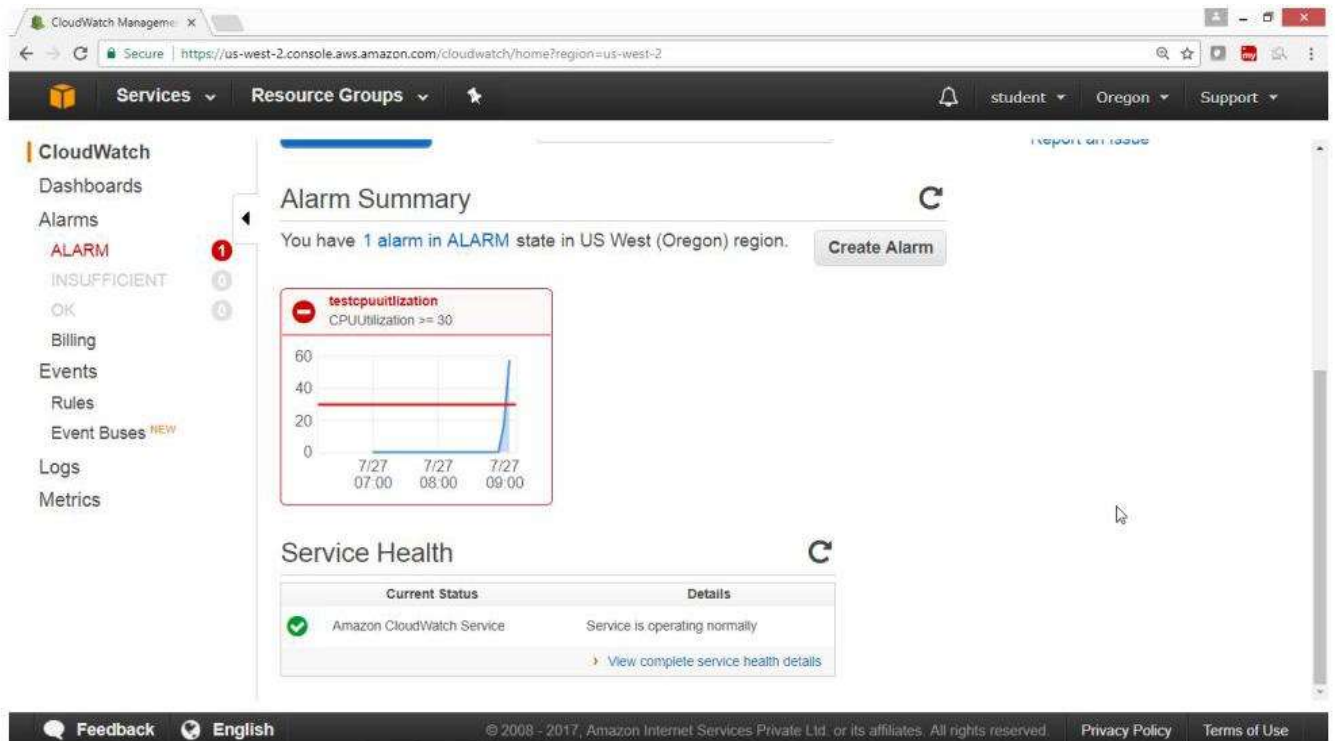
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3143	root	20	0	7260	96	0	R	2.7	0.0	0:00.73	stress
3147	root	20	0	7260	96	0	R	2.7	0.0	0:00.73	stress
3179	root	20	0	7260	96	0	R	2.7	0.0	0:00.73	stress
3141	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3142	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3144	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3145	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3146	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3148	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3149	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3150	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3151	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3152	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3153	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3154	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3155	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3156	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3157	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3158	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3159	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3160	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3161	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3162	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress
3163	root	20	0	7260	96	0	R	2.3	0.0	0:00.72	stress

Go to CloudWatch Service and check the status



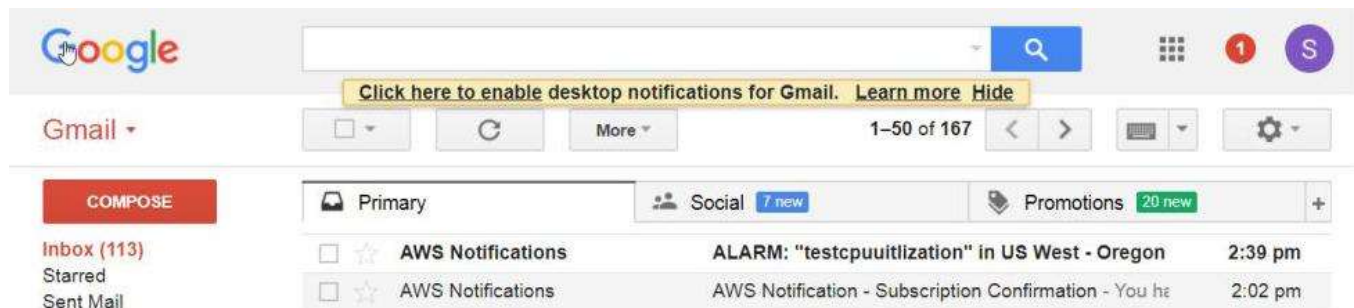
The screenshot shows the AWS CloudWatch console in the US West (Oregon) region. The left sidebar contains navigation links: CloudWatch, Dashboards, Alarms, ALARM, INSUFFICIENT, OK, Billing, Events, Rules, Event Buses (NEW), Logs, and Metrics. The main content area is titled "Alarm Summary" and displays the message: "All your alarms are in OK state in US West (Oregon) region." A "Create Alarm" button is visible. Below this, a graph for the "testcpuutilization" alarm (threshold: CPUUtilization >= 30) shows a line graph with a red threshold line at 30. The graph shows a spike in CPU utilization starting around 08:00 on 7/27, reaching approximately 35 by 09:00. The "Service Health" section below the graph shows the "Amazon CloudWatch Service" with a "Current Status" of "OK" and a message: "Service is operating normally." A link to "View complete service health details" is provided.

After 5 minutes Alarm is generated



The screenshot shows the AWS CloudWatch console in the US West (Oregon) region. The left sidebar contains navigation links: CloudWatch, Dashboards, Alarms, ALARM, INSUFFICIENT, OK, Billing, Events, Rules, Event Buses (NEW), Logs, and Metrics. The main content area is titled "Alarm Summary" and displays the message: "You have 1 alarm in ALARM state in US West (Oregon) region." A "Create Alarm" button is visible. Below this, a graph for the "testcpuutilization" alarm (threshold: CPUUtilization >= 30) shows a line graph with a red threshold line at 30. The graph shows a spike in CPU utilization starting around 08:00 on 7/27, reaching approximately 35 by 09:00. The "Service Health" section below the graph shows the "Amazon CloudWatch Service" with a "Current Status" of "OK" and a message: "Service is operating normally." A link to "View complete service health details" is provided.




Go to email and check mail



Check on mail & Verify the Output

Verify output

=====

 **AWS Notifications** no-reply@sns.ε 2:39 PM (2 minutes ago) ☆  
to me 

You are receiving this email because your Amazon CloudWatch Alarm "testcpuutilization" in the US West - Oregon region has entered the ALARM state, because "Threshold Crossed: 1 datapoint [46.236000000000004 (27/07/17 09:04:00)] was greater than or equal to the threshold (30.0)." at "Thursday 27 July, 2017 09:09:58 UTC".

View this alarm in the AWS Management Console:

<https://console.aws.amazon.com/cloudwatch/home?region=us-west-2#s=Alarms&alarm=testcpuutilization>

Alarm Details:

- Name: testcpuutilization
- Description: cputest
- State Change: OK -> ALARM
- Reason for State Change: Threshold Crossed: 1 datapoint [46.236000000000004 (27/07/17 09:04:00)] was greater than or equal to the threshold (30.0).
- Timestamp: Thursday 27 July, 2017 09:09:58 UTC

🖱 - Timestamp: Thursday 27 July, 2017 09:09:58 UTC
- AWS Account: 523251683217

Threshold:

- The alarm is in the ALARM state when the metric is
GreaterThanOrEqualToThreshold 30.0 for 300 seconds.

Monitored Metric:

- MetricNamespace: AWS/EC2
- MetricName: CPUUtilization
- Dimensions: [InstanceId = i-081a441f51fc90525]
- Period: 300 seconds
- Statistic: Average
- Unit: not specified

State Change Actions:

- OK:
- ALARM: [arn:aws:sns:us-west-2:523251683217:CPUtopicabe]
- INSUFFICIENT_DATA:

🖱 State Change Actions:

- OK:
- ALARM: [arn:aws:sns:us-west-2:523251683217:CPUtopicabe]
- INSUFFICIENT_DATA:

--

If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:

[https://sns.us-west-2.amazonaws.com/unsubscribe.html?
SubscriptionArn=arn:aws:sns:us-west-2:523251683217:
CPUtopicabe:e8d238f8-8e77-46ec-8b2f-609f9ba26876&
Endpoint=: adminabc@abc.com](https://sns.us-west-2.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:us-west-2:523251683217:CPUtopicabe:e8d238f8-8e77-46ec-8b2f-609f9ba26876&Endpoint=:adminabc@abc.com)

Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at
<https://aws.amazon.com/support>

===== END OF OUTPUT =====



Amazon CloudFormation

CloudFormation Highlights

AWS CloudFormation is a service that helps you model and set up your AWS resources. So that you can spend less time managing those resources and more time focusing on your applications that run in AWS.

Codifies creation of stack of resources stack could be: -

- ELB
- Autoscaling group
- EC2
- RDS [Database]
- All Connections between them.

The benefits of Cloud Formation are: -

- Your Infrastructure as CODE.
- Can be version Controlled
- No more guessing. Who did. what. where.
- Modularization
- Enforce "One way to deploy"
- CF costs nothing but for the resources created using it.

The 7 Sections in the Templates of Cloud Formation: -

1. Version [of CF] - Which Year it was build, in date format.
2. Description - It gives details about the template.
3. Parameters - Runtime variables, like subnet, VPCID, InstanceTypes, DB Name, etc.
4. Mappings - Available types and allocated.
5. Resources [AWS to create] - ELB, WebServers, AppSvrs, etc.
6. Properties - Specific to Resources
7. OutPuts - Final outputs upon creation.

The key point in Cloud Formation are: -

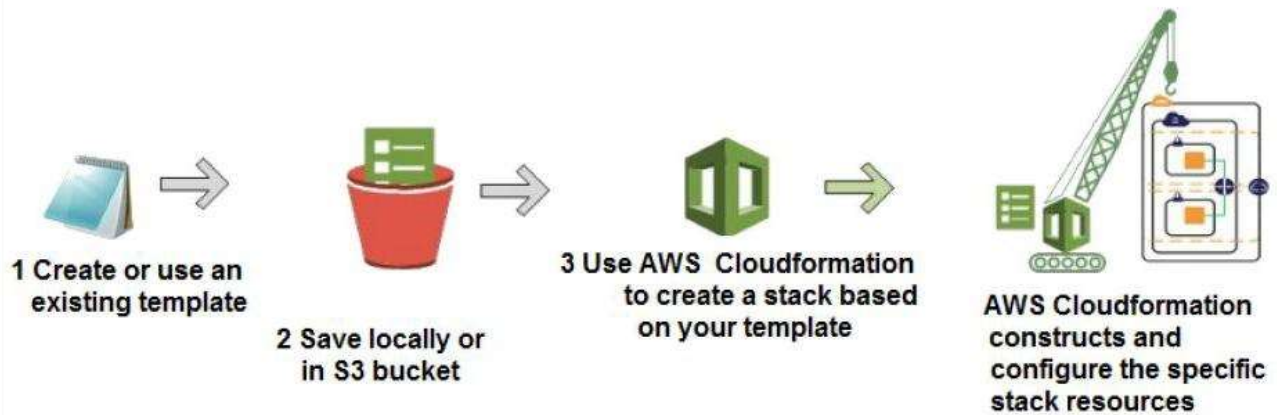
- If you sign up with Cloud formation means, you are signing up with ALL AWS SERVICES, that CF can create,
- Setting up Alarms!
- 200-300 templates are available to choose.
- Templates are JSON based
- Templates can accept "RunTime" parameters [Instance type / Key Pair].

Share the CloudFront Configuration Step by Step?

To configure AWS CloudFormation

Topology

AWS CloudFormation



Pre-requisites

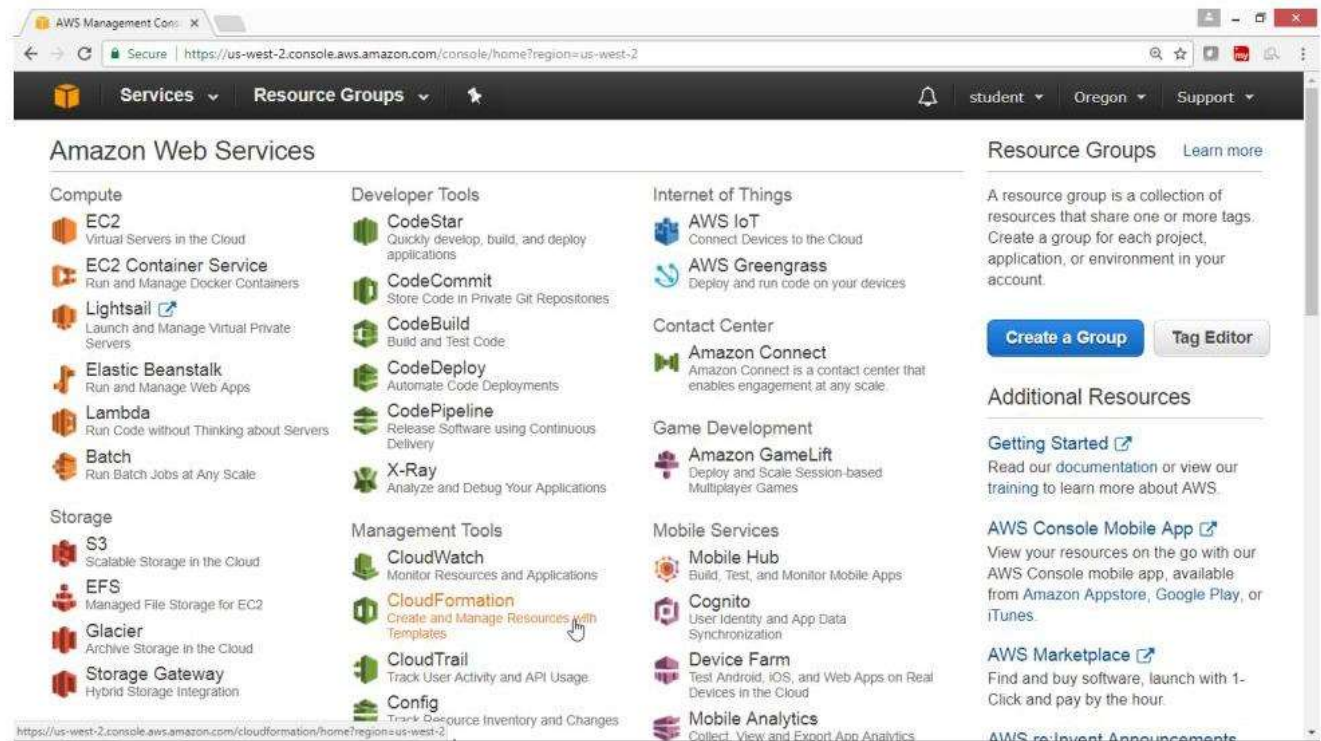
- User should have AWS account or IAM user with CloudFormation Full Access Policy

Task

- Creating EC2 instance using CloudFormation
- Deleting all resources from CloudFormation

Step-1) To launch Amazon EC2 instance in a security group using CloudFormation

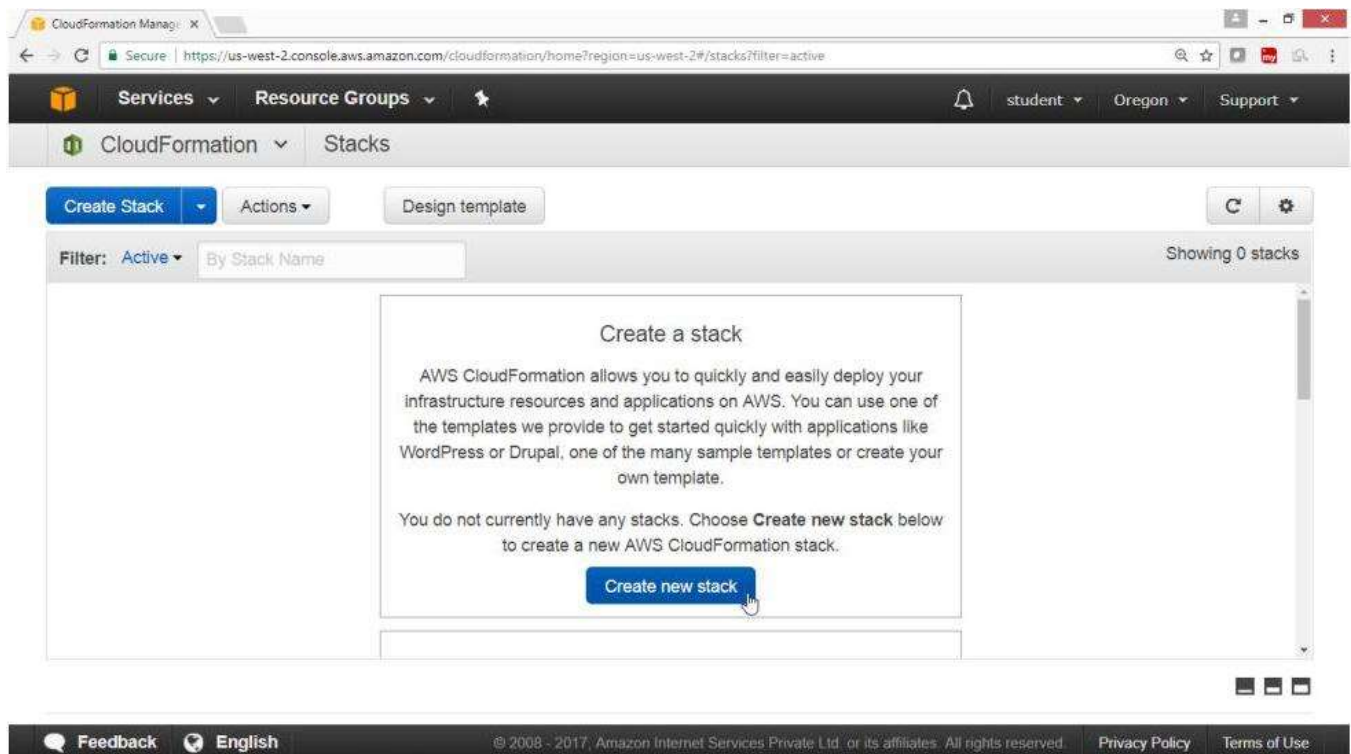
- Open [AWS Console](#)
- Click on [Services](#)
- In [Management Tools - Services](#)
- Click [CloudFormation Service](#)



Step-2) To create a new stack

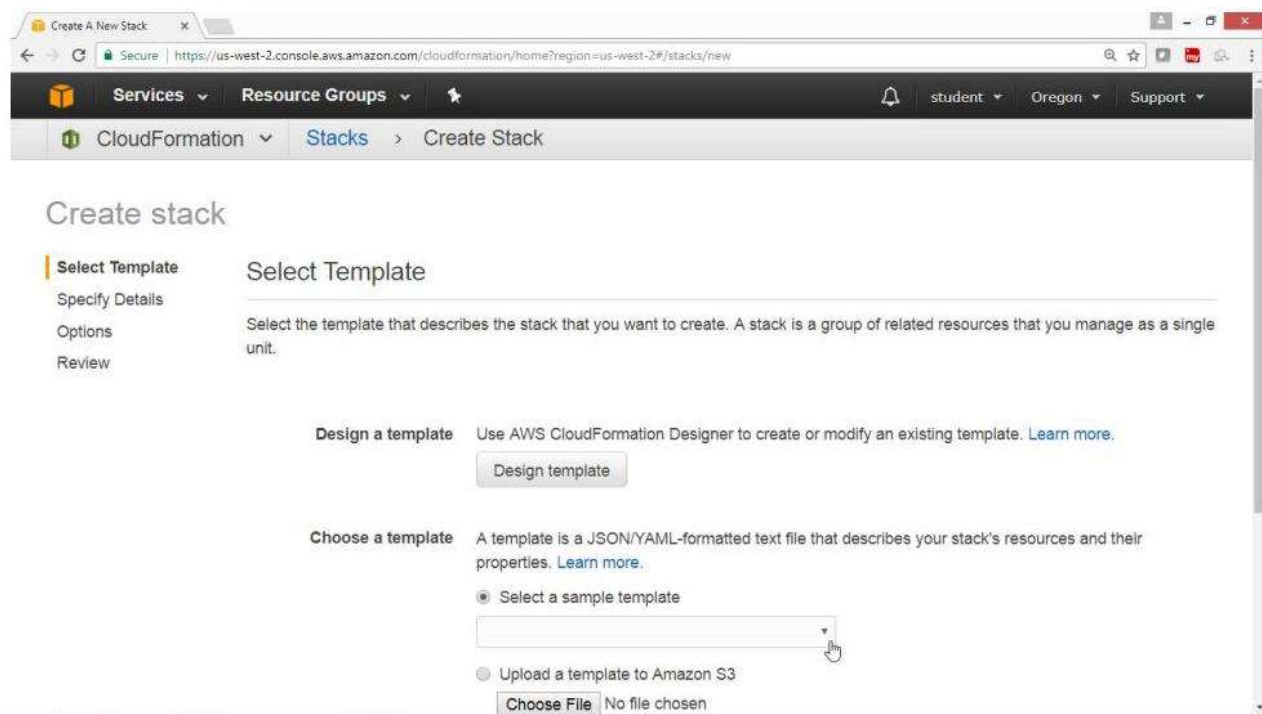
On "Create Stack", page

Click on "Create New Stack" button



Under "Choose a template"

Select "Select a sample template"

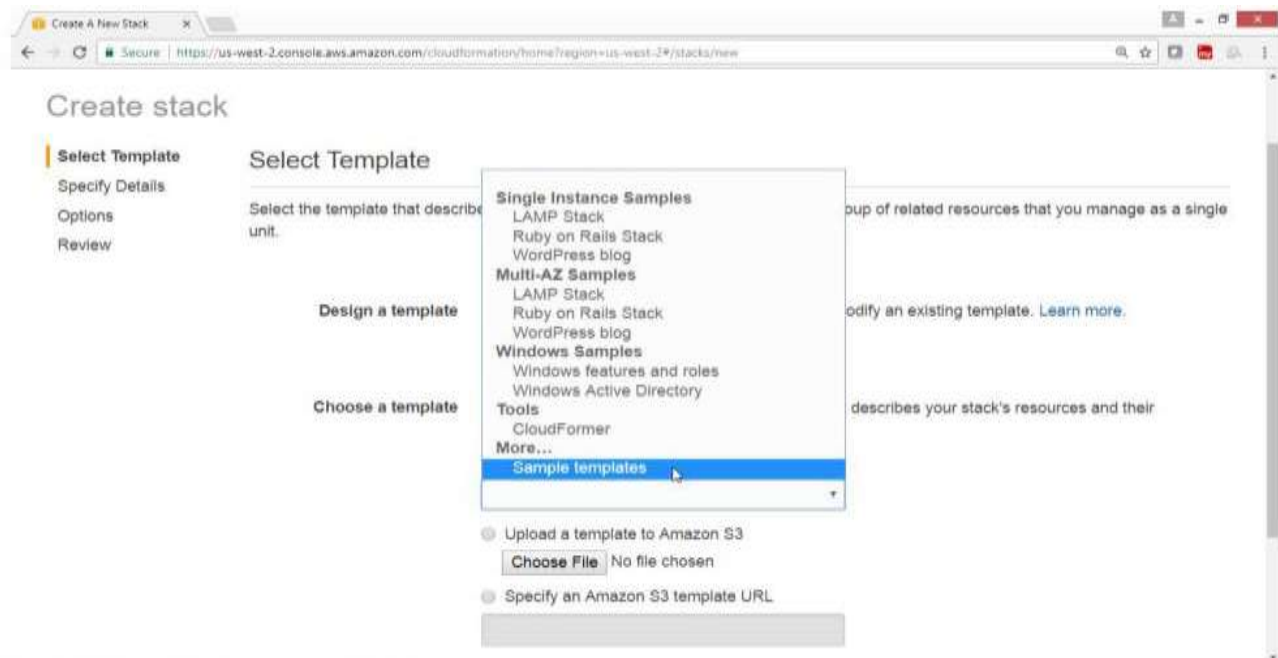


On Create stack page

Select the "Sample template"

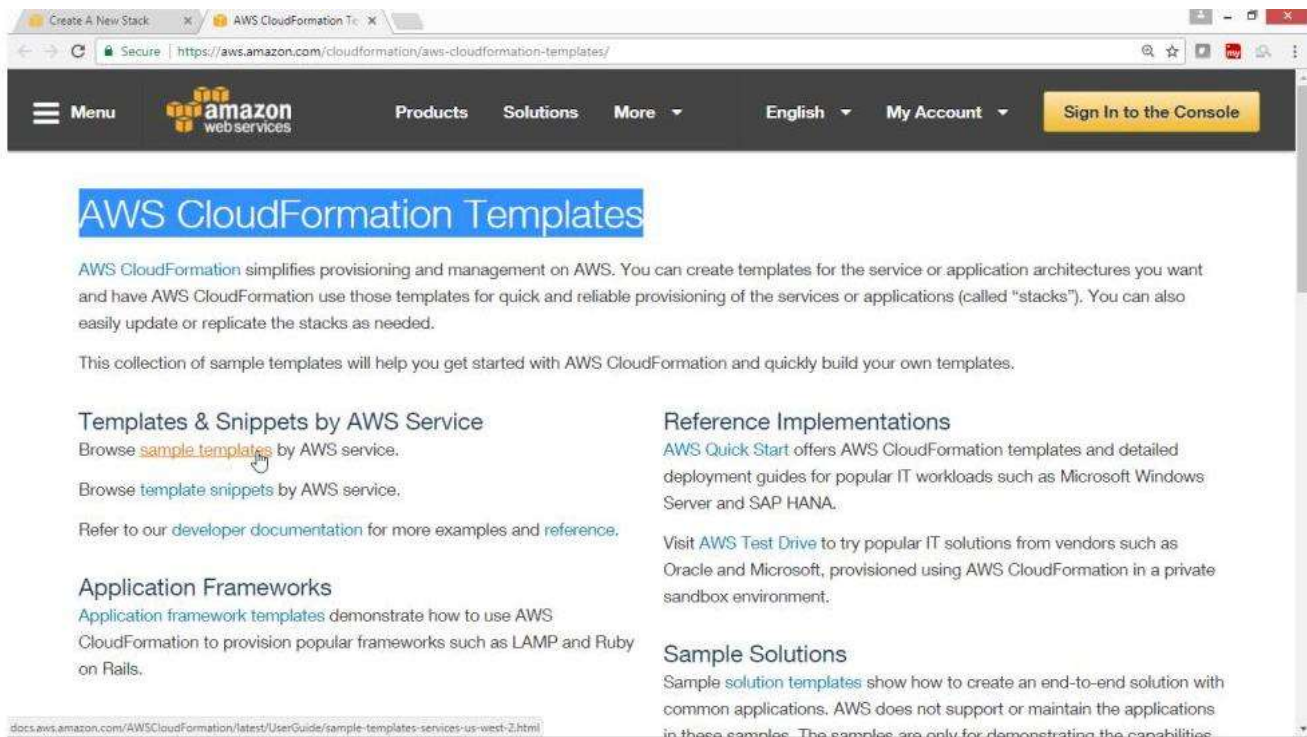
In the Drop Down, box

Choose "Sample templates" option



On "AWS CloudFormation Templates" page


Click on "sample templates"



The screenshot shows the AWS CloudFormation Templates page. The header includes the AWS logo, navigation links (Menu, Products, Solutions, More), language (English), account (My Account), and a 'Sign In to the Console' button. The main heading is 'AWS CloudFormation Templates'. Below it, a paragraph explains that AWS CloudFormation simplifies provisioning and management on AWS, allowing users to create templates for service or application architectures and have AWS CloudFormation provision those templates for quick and reliable provisioning of the services or applications (called "stacks"). It also mentions that users can easily update or replicate the stacks as needed. A sub-paragraph states that this collection of sample templates will help users get started with AWS CloudFormation and quickly build their own templates. The page is divided into four sections: 'Templates & Snippets by AWS Service' (with links to 'sample templates' and 'template snippets'), 'Reference Implementations' (with links to 'AWS Quick Start' and 'AWS Test Drive'), 'Application Frameworks' (with a link to 'application framework templates'), and 'Sample Solutions' (with a link to 'sample solution templates'). A URL bar at the bottom shows the path: docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/sample-templates-services-us-west-2.html.

Create A New Stack x AWS CloudFormation T x

Secure | https://aws.amazon.com/cloudformation/aws-cloudformation-templates/

Menu  Products Solutions More English My Account Sign In to the Console

AWS CloudFormation Templates

AWS CloudFormation simplifies provisioning and management on AWS. You can create templates for the service or application architectures you want and have AWS CloudFormation use those templates for quick and reliable provisioning of the services or applications (called "stacks"). You can also easily update or replicate the stacks as needed.

This collection of sample templates will help you get started with AWS CloudFormation and quickly build your own templates.

Templates & Snippets by AWS Service

Browse [sample templates](#) by AWS service.

Browse [template snippets](#) by AWS service.

Refer to our [developer documentation](#) for more examples and [reference](#).

Application Frameworks

[Application framework templates](#) demonstrate how to use AWS CloudFormation to provision popular frameworks such as LAMP and Ruby on Rails.

Reference Implementations

[AWS Quick Start](#) offers AWS CloudFormation templates and detailed deployment guides for popular IT workloads such as Microsoft Windows Server and SAP HANA.

Visit [AWS Test Drive](#) to try popular IT solutions from vendors such as Oracle and Microsoft, provisioned using AWS CloudFormation in a private sandbox environment.

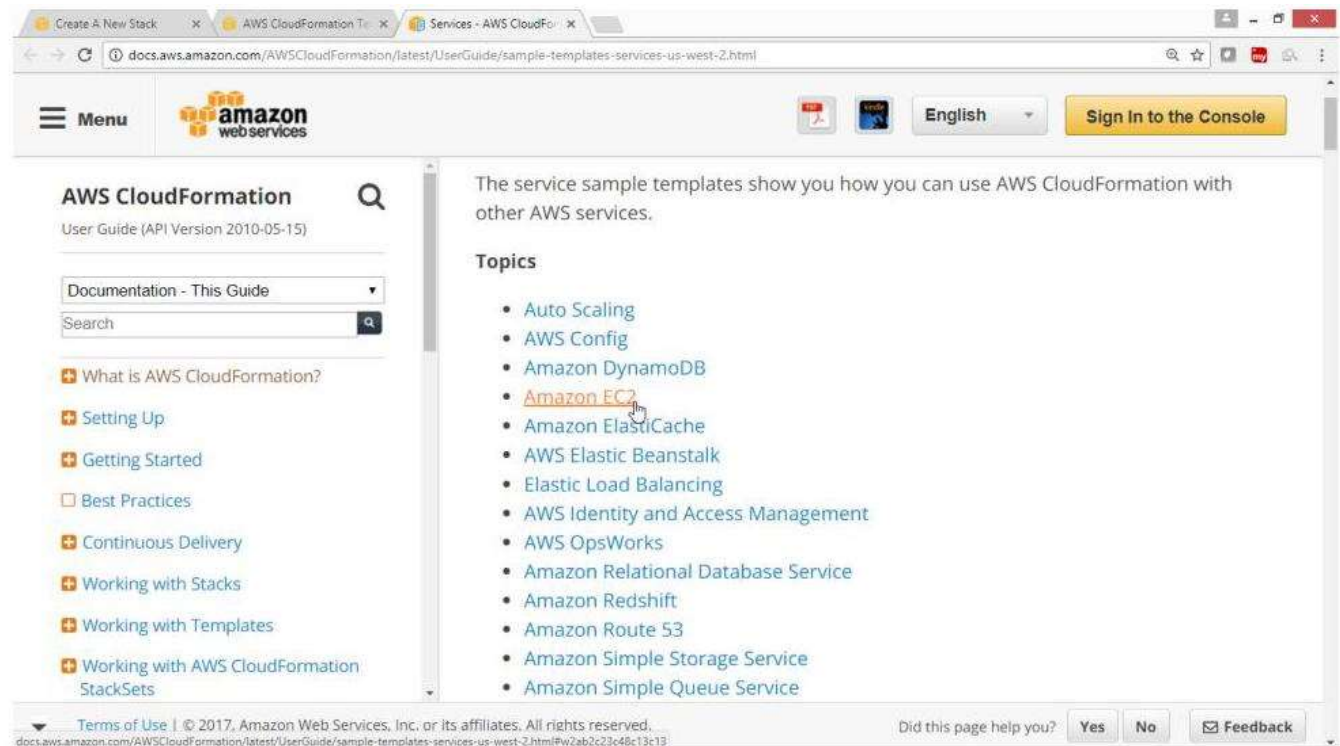
Sample Solutions

[Sample solution templates](#) show how to create an end-to-end solution with common applications. AWS does not support or maintain the applications in these samples. The samples are only for demonstrating the capabilities.

docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/sample-templates-services-us-west-2.html

Under Topics


Select [Amazon EC2](#)



The screenshot shows the AWS CloudFormation User Guide page. The header includes the AWS logo, navigation links (Menu, Services - AWS CloudFormation), language (English), and a 'Sign In to the Console' button. The main heading is 'AWS CloudFormation User Guide (API Version 2010-05-15)'. Below it, there is a search bar and a list of topics. The 'Topics' section lists various AWS services and features, with 'Amazon EC2' highlighted. A paragraph states that the service sample templates show how users can use AWS CloudFormation with other AWS services. A footer at the bottom contains the 'Terms of Use' and a 'Did this page help you?' feedback section with 'Yes', 'No', and 'Feedback' buttons. A URL bar at the bottom shows the path: docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/sample-templates-services-us-west-2.html#w2ab2c23c48c13c13.

Create A New Stack x AWS CloudFormation T x Services - AWS CloudFormation x

docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/sample-templates-services-us-west-2.html

Menu  English Sign In to the Console

AWS CloudFormation

User Guide (API Version 2010-05-15)

Documentation - This Guide

Search

- What is AWS CloudFormation?
- Setting Up
- Getting Started
- Best Practices
- Continuous Delivery
- Working with Stacks
- Working with Templates
- Working with AWS CloudFormation StackSets

The service sample templates show you how you can use AWS CloudFormation with other AWS services.

Topics

- Auto Scaling
- AWS Config
- Amazon DynamoDB
- Amazon EC2**
- Amazon ElastiCache
- AWS Elastic Beanstalk
- Elastic Load Balancing
- AWS Identity and Access Management
- AWS OpsWorks
- Amazon Relational Database Service
- Amazon Redshift
- Amazon Route 53
- Amazon Simple Storage Service
- Amazon Simple Queue Service

Terms of Use | © 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Did this page help you? Yes No Feedback

docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/sample-templates-services-us-west-2.html#w2ab2c23c48c13c13

Select ["Amazon Ec2 instance in a security group"](#)

Click on ["Launch stack"](#)

Amazon EC2

Template Name	Description	View	View in Designer	Launch
Amazon EC2 instance in a security group	Creates an Amazon EC2 instance in an Amazon EC2 security group.	View	View in Designer	Launch Stack
Amazon EC2 instance with an Elastic IP address	Creates an Amazon EC2 instance and associates an Elastic IP address with the instance.	View	View in Designer	Launch Stack
Amazon EC2 instance with an ephemeral drive	Creates an Amazon EC2 instance with an ephemeral drive by using a block device mapping.	View	View in Designer	Launch Stack

Amazon ElastiCache

Template Name	Description	View	View in Designer	Launch
ElastiCache	Creates an ElastiCache cache cluster with the Memcached	View	View in Designer	Launch Stack

Terms of Use | © 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Did this page help you? [Yes](#) [No](#) [Feedback](#)

In option "**Specify an Amazon S3 template URL**"

Verify template is loaded in S3

Click on **Next Button**

Review

Design a template Use AWS CloudFormation Designer to create or modify an existing template. [Learn more.](#)
[Design template](#)

Choose a template A template is a JSON/YAML-formatted text file that describes your stack's resources and their properties. [Learn more.](#)

☐ Select a sample template
☐ Upload a template to Amazon S3
☒ Specify an Amazon S3 template URL

[Choose File](#) No file chosen
 [View/Edit template in Designer](#)

[Cancel](#) [Next](#)

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

On Specific Details page

Key Name-> "key*.pem"

Click on Next button

Create a New Stack

Select Template

Specify Details

Options

Review

Specify Details

Specify a stack name and parameter values. You can use or change the default parameter values, which are defined in the AWS CloudFormation template. [Learn more.](#)

Stack name: EC2SecurityGroupSample

Parameters

InstanceType: t2.small (WebServer EC2 instance type)

KeyName: 25july2017masorg (Name of an existing EC2 KeyPair to enable SSH access to the instance)

SSHLocation: 0.0.0.0/0 (The IP address range that can be used to SSH to the EC2 instances)

Cancel Previous **Next**

Under **Options Tag**, provide values for

Key -> Nameweb

Value -> Web

Drag Down

Create stack

Select Template

Specify Details

Options

Review

Options

Tags

You can specify tags (key-value pairs) for resources in your stack. You can add up to 50 unique key-value pairs for each stack. [Learn more.](#)

	Key (127 characters maximum)	Value (255 characters maximum)	
1	Nameweb	web	+

Permissions

You can choose an IAM role that CloudFormation uses to create, modify, or delete resources in the stack. If you don't choose a role, CloudFormation uses the permissions defined in your account. [Learn more.](#)

IAM Role: Choose a role (optional)

Enter role arn

Previous **Next**

Click on **Next**

Create A New Stack

Securehttps://us-west-2.console.aws.amazon.com/cloudformation/home?region=us-west-2#/stacks/new?stackName=EC2SecurityGroupSample&templateURL=https://s3-us-west-2.amazonaws.com/cloudformation-templates-us-west-2/EC2InstanceWithSecurityGroupSample.template

Key (127 characters maximum)	Value (255 characters maximum)
1 Nameweb	web

Permissions

You can choose an IAM role that CloudFormation uses to create, modify, or delete resources in the stack. If you don't choose a role, CloudFormation uses the permissions defined in your account. [Learn more.](#)

IAM Role

Choose a role (optional)

Enter role arn

Advanced

You can set additional options for your stack, like notification options and a stack policy. [Learn more.](#)

Cancel

Previous

Next

Feedback

English

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

Privacy Policy

Terms of Use

Review, check the summary

Create A New Stack

Securehttps://us-west-2.console.aws.amazon.com/cloudformation/home?region=us-west-2#/stacks/new?stackName=EC2SecurityGroupSample&templateURL=https://s3-us-west-2.amazonaws.com/cloudformation-templates-us-west-2/EC2InstanceWithSecurityGroupSample.template

Create stack

Select Template

Specify Details

Options

Review

Review

Template

Template URL

https://s3-us-west-2.amazonaws.com/cloudformation-templates-us-west-2/EC2InstanceWithSecurityGroupSample.template

Description

AWS CloudFormation Sample Template EC2InstanceWithSecurityGroupSample: Create an Amazon EC2 instance running the Amazon Linux AMI. The AMI is chosen based on the region in which the stack is run. This example creates an EC2 security group for the instance to give you SSH access.
WARNING This template creates an Amazon EC2 instance. You will be billed for the AWS resources used if you create a stack from this template.

Estimate cost

Cost

Details

Stack name:

EC2SecurityGroupSample

InstanceType

t2.small

KeyName

25july2017masorg

SSHLocation

0.0.0.0/0

Click **Create Button**

Stack name: EC2SecurityGroupSample

InstanceType: t2.small
KeyName: 25july2017masorg
SSHLocation: 0.0.0.0/0

Options

Tags

Name: web

Advanced

Notification Timeout: none
Rollback on failure: Yes

Cancel Previous **Create**

Check the status

CloudFormation is in progress state

Introducing StackSets

AWS StackSet is a container for a set of AWS CloudFormation stacks and allows you to create stacks across multiple AWS Accounts and AWS Regions. Open the StackSets console to get started.

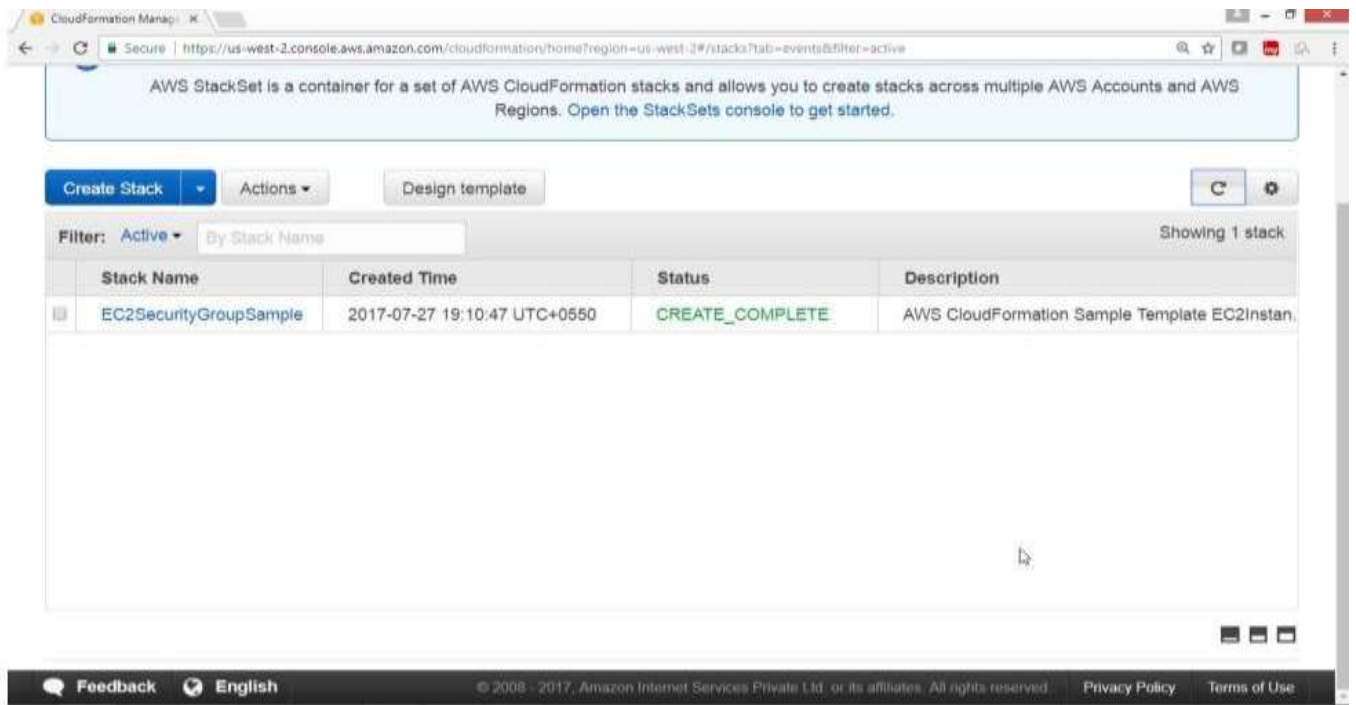
Create Stack Actions Design template

Filter: Active By Stack Name Showing 1 stack

	Stack Name	Created Time	Status	Description
	EC2SecurityGroupSample	2017-07-27 19:10:47 UTC+0550	CREATE_IN_PROGRESS	AWS CloudFormation Sample Template EC2Instan.

Verify

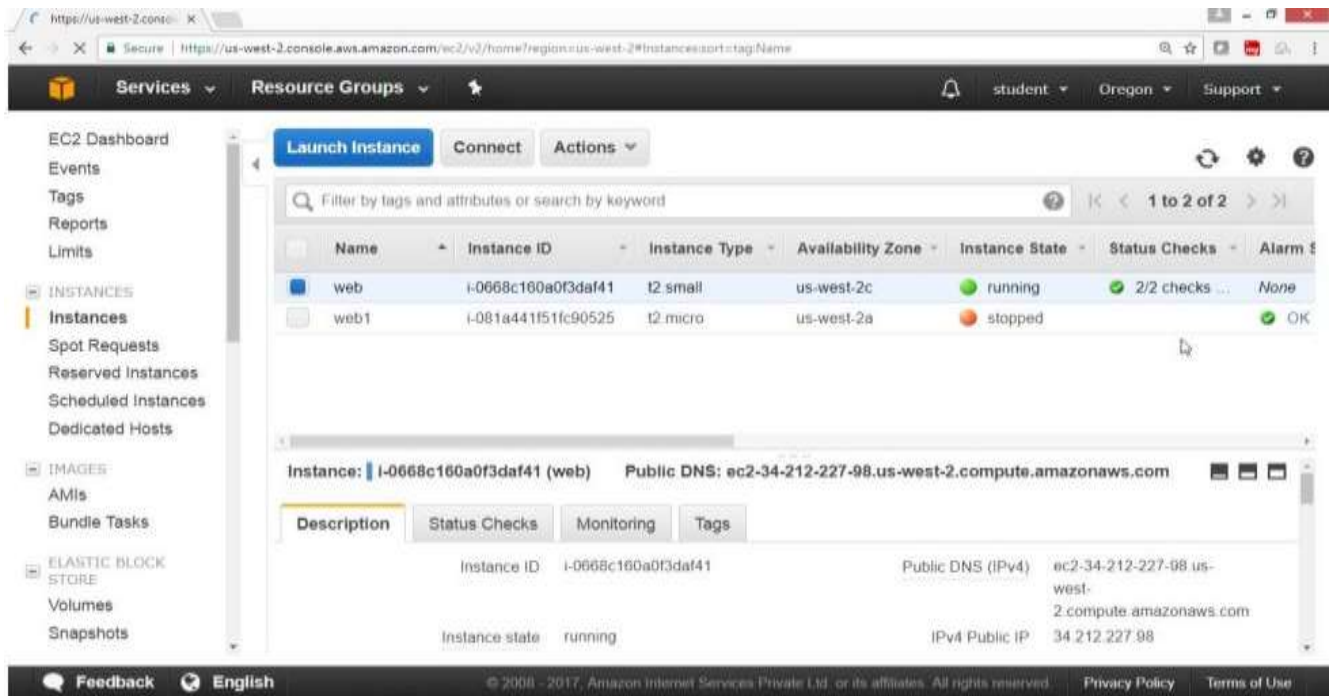
Status is **Create Complete**



Go to EC2 Service

Check the instances

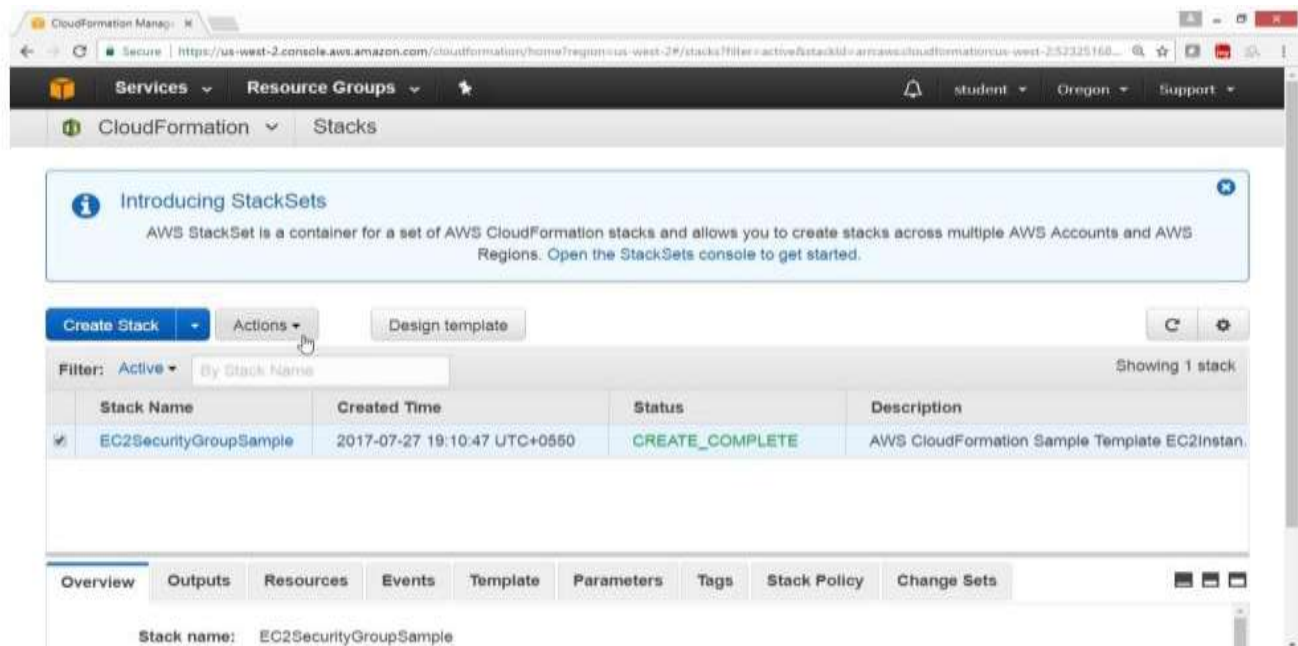
An instance with the Name "web" is launched



Step-3) To remove the Instances created by CloudFormation

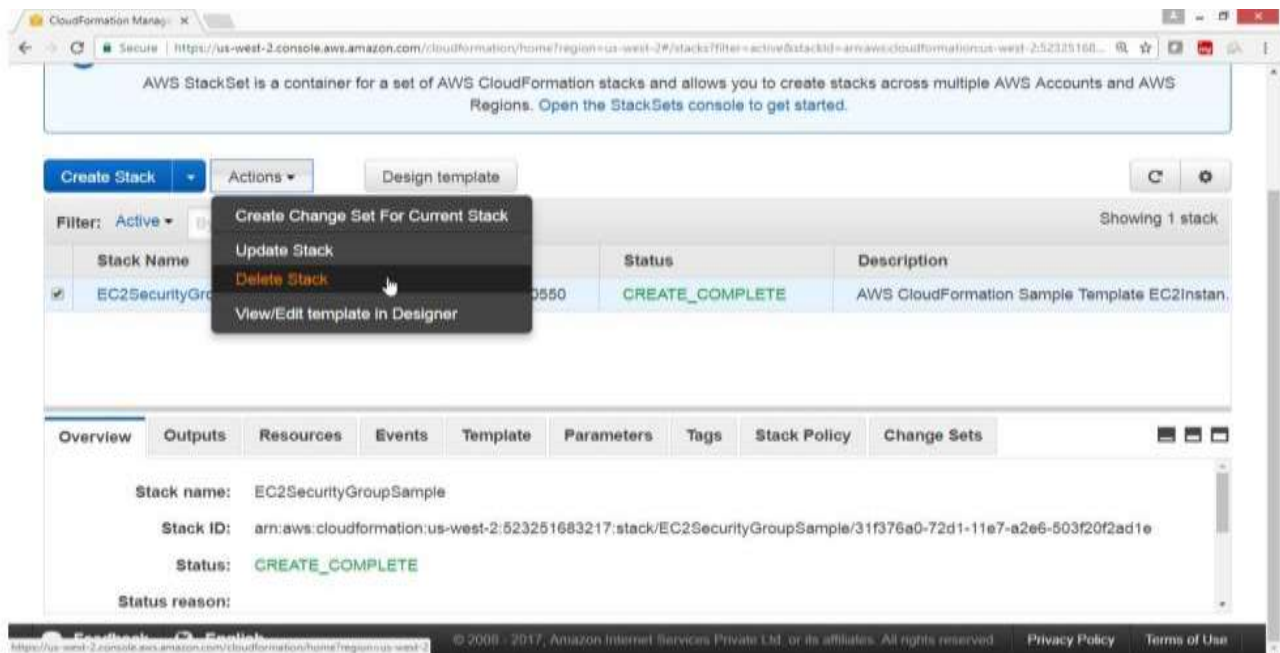
From AWS Console

- Select Services **Management Tools**
- Select **CloudFormation**
- Select the Stack Name checkbox

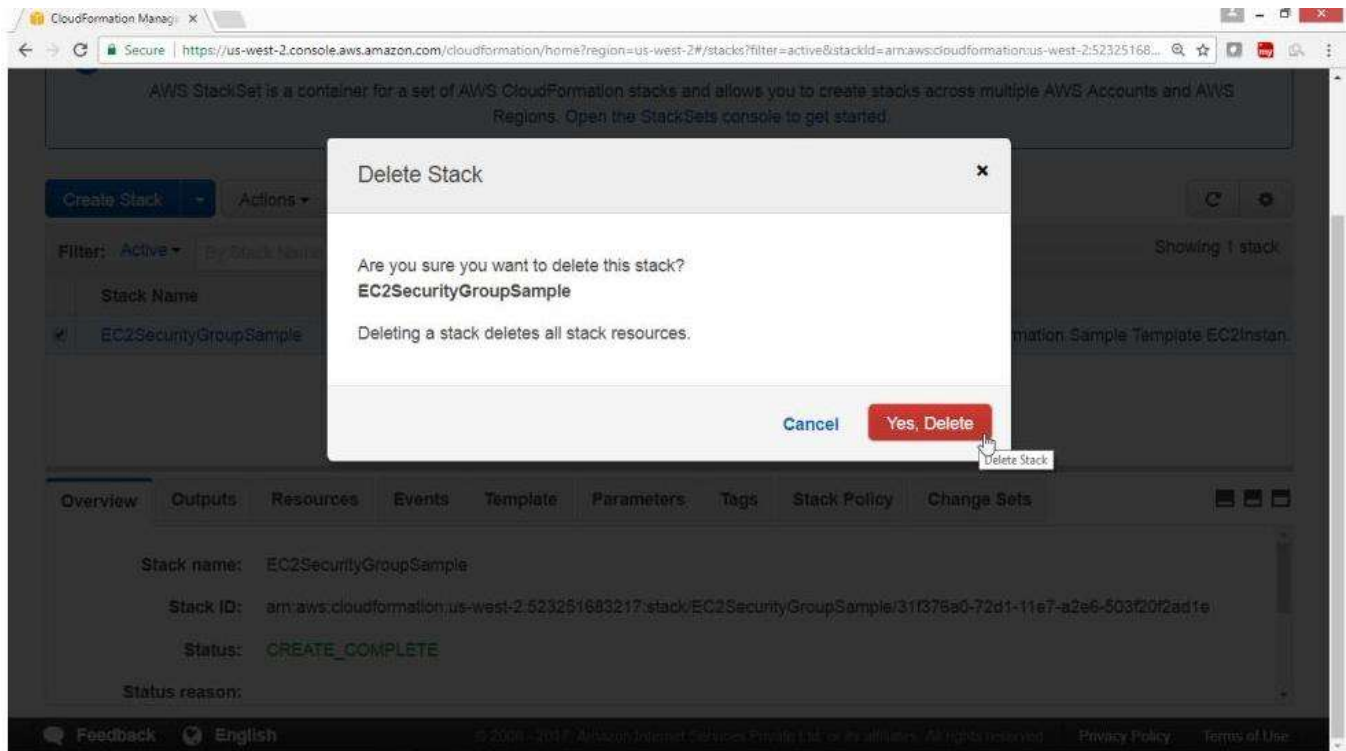


Click on **Actions** button

Select "**Delete stack**"

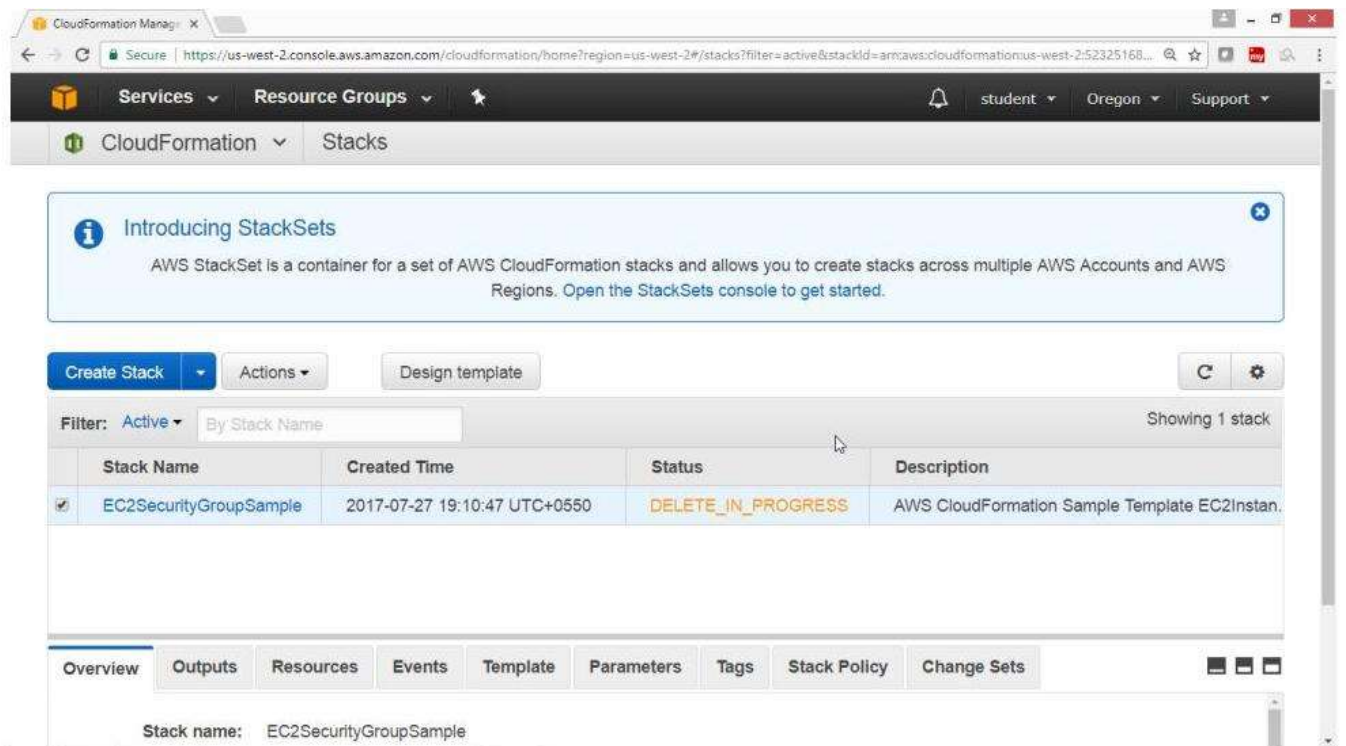


Click on "Yes, Delete"



Verify

Delete is in progress



Verification

After deletion again starting screen of CloudFormation is displayed

CloudFormation Manager

Secure | https://us-west-2.console.aws.amazon.com/cloudformation/home?region=us-west-2#/stacks?filter=active

Services

Resource Groups

student Oregon Support

CloudFormation

Stacks

Create Stack

Actions

Design template

Filter: Active By Stack Name Showing 0 stacks

Create a stack

AWS CloudFormation allows you to quickly and easily deploy your infrastructure resources and applications on AWS. You can use one of the templates we provide to get started quickly with applications like WordPress or Drupal, one of the many sample templates or create your own template.

You do not currently have any stacks. Choose **Create new stack** below to create a new AWS CloudFormation stack.

Create new stack

Feedback

English

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

Privacy Policy

Terms of Use



AWS OpsWorks

How is AWS OpsWorks different than AWS CloudFormation?

OpsWorks and CloudFormation both support application modelling, deployment, configuration, management and related activities. Both support a wide variety of architectural patterns, from simple web applications to highly complex applications. AWS OpsWorks and AWS CloudFormation differ in abstraction level and areas of focus.

AWS CloudFormation is a building block service which enables customer to manage almost any AWS resource via JSON-based domain specific language. It provides foundational capabilities for the full breadth of AWS, without prescribing a particular model for development and operations. Customers define templates and use them to provision and manage AWS resources, operating systems and application code.

In contrast, AWS OpsWorks is a higher-level service that focuses on providing highly productive and reliable DevOps experiences for IT administrators and ops-minded developers. To do this, AWS OpsWorks employs a configuration management model based on concepts such as stacks and layers, and provides integrated experiences for key activities like deployment, monitoring, auto-scaling, and automation. Compared to AWS CloudFormation, AWS OpsWorks supports a narrower range of application-oriented AWS resource types including Amazon EC2 instances, Amazon EBS volumes, Elastic IPs, and Amazon CloudWatch metrics.

A company needs to monitor the read and write IOPS for their AWS MySQL RDS instance and send real-time alerts to their operations team. Which AWS services can accomplish this?

- A. Amazon Simple Email Service
- B. Amazon CloudWatch**
- C. Amazon Simple Queue Service
- D. Amazon Route 53

Answer B

Explanation: Amazon CloudWatch is a cloud monitoring tool and hence this is the right service for the mentioned use case. The other options listed here are used for other purposes for example route 53 is used for DNS services, therefore CloudWatch will be the apt choice.

What happens when one of the resources in a stack cannot be created successfully in AWS OpsWorks?

When an event like this occurs, the “automatic rollback on error” feature is enabled, which causes all the AWS resources which were created successfully till the point where the error occurred to be deleted. This is helpful since it does not leave behind any erroneous data, it ensures the fact that stacks are either created fully or not created at all. It is useful in events where you may accidentally exceed your limit of the no. of Elastic IP addresses or maybe you may not have access to an EC2 AMI that you are trying to run etc.

What automation tools can you use to spin up servers?

The API tools can be used for spinup services and also for the written scripts.

- Those scripts could be coded in Perl, bash or other languages of your preference.
- There is one more option that is patterned administration and stipulating tools such as a dummy or improved descendant.
- A tool called [Scalar](#) can also be used and finally we can go with a controlled explanation like a Rightscale.



Application Integration & Customer Engagement

AWS Step Functions Coordinate Distributed Applications	AWS Simple Queue Service (SQS) Managed Message Queues	Amazon CloudFormation Pub/Sub, Mobile Push and SMS
Amazon MQ Managed Message Broker for ActiveMQ		



Application Integration

Amazon Connect Cloud based Contact Center	Amazon Pinpoint Push Notifications for Mobile Apps	Amazon Simple Email Service (SES) Email Sending and Receiving



Customer Engagement

AWS Simple Queue Service

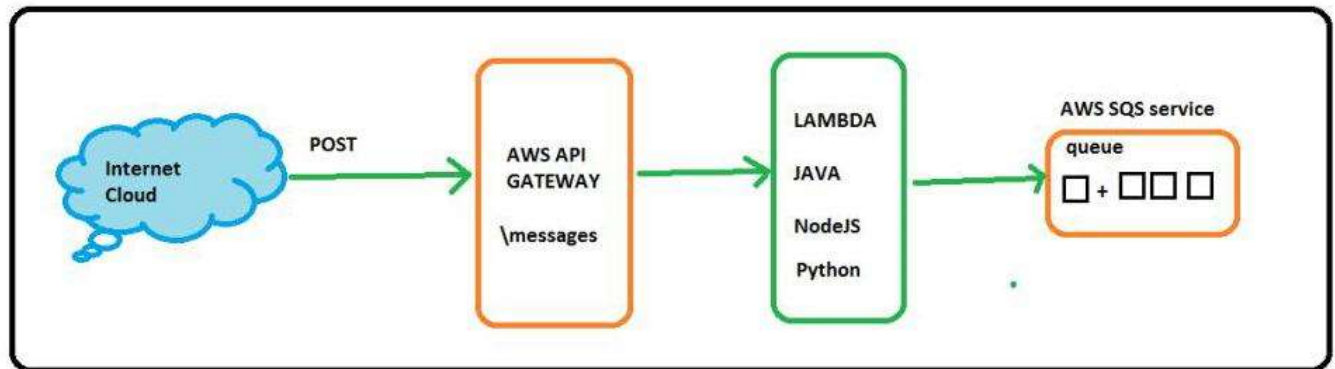
SQS Highlights

- SQS is pull based, not pushed base
- Messages are 256 KB in size
- Messages can be kept in the queue from 1 minute to 14 days. The default is 4 days
- Visibility Time Out is the amount of time that the message is invisible in the SQS queue after a reader picks up that message
- Provided the job is processed before the visibility time out expires, the message will then be deleted from the queue. If the job is not processed within that time, the message will become visible again and another reader will process it. This could result in the same message being delivered twice.
- Visibility time out maximum is 12 hours
- SQS guarantees that your messages will be processed at least once.
- Amazon SQS long polling is a way to retrieve messages from your Amazon SQS queues. While the regular short polling returns immediately, even if the message queue being polled is empty, long polling does not return a response until a message arrives in the message queue or the long poll time out
- Queues can either be standard or [FIFO](#)

Share the SQS Configuration Step by Step?

To Configure and use Simple Queue Service (SQS)

Topology



Pre-requisites

User should have AWS account, or IAM user with SQSfullaccess

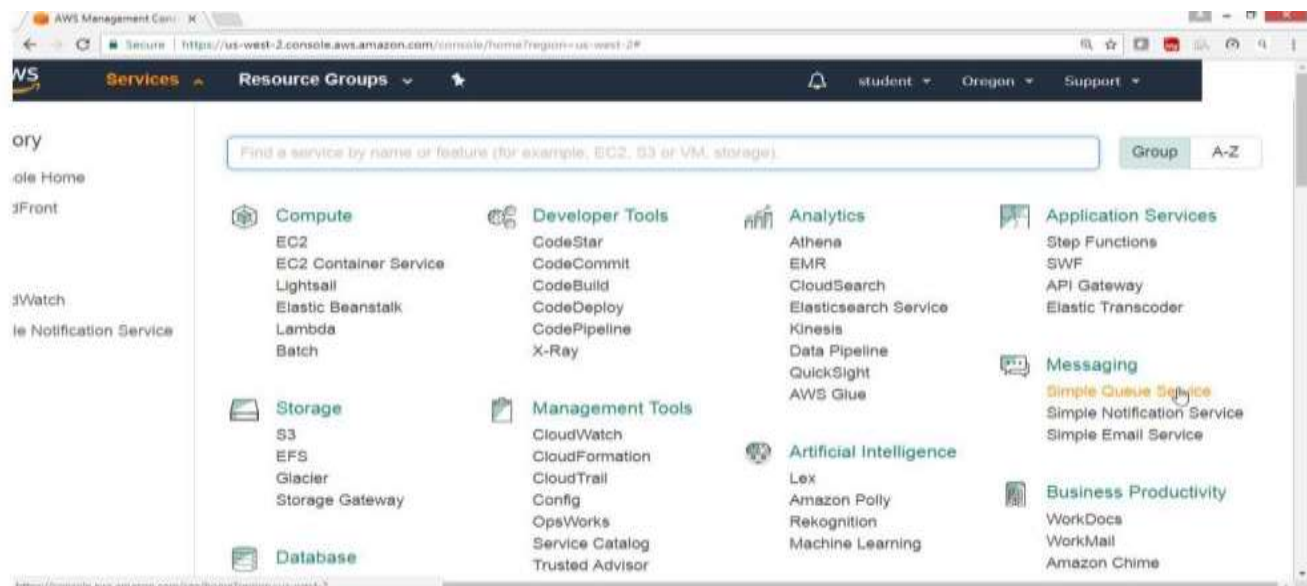
To Configure SQS with following task:

- Create the queue
- Send the message
- Pool the queue
- View the message
- Delete the message

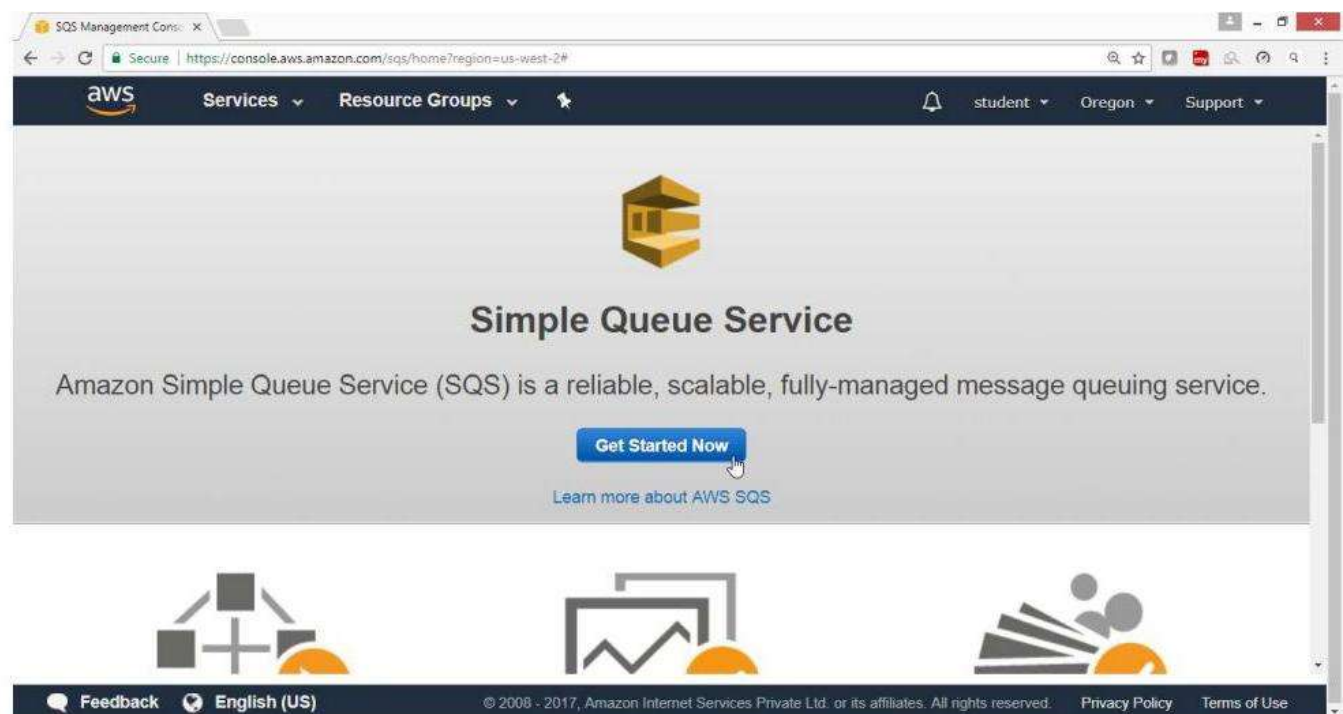
1) To Configure Amazon Simple Queue Service (SQS)

From the AWS console select service Application Integration

Select **Simple Queue Service**



Click on Get started on

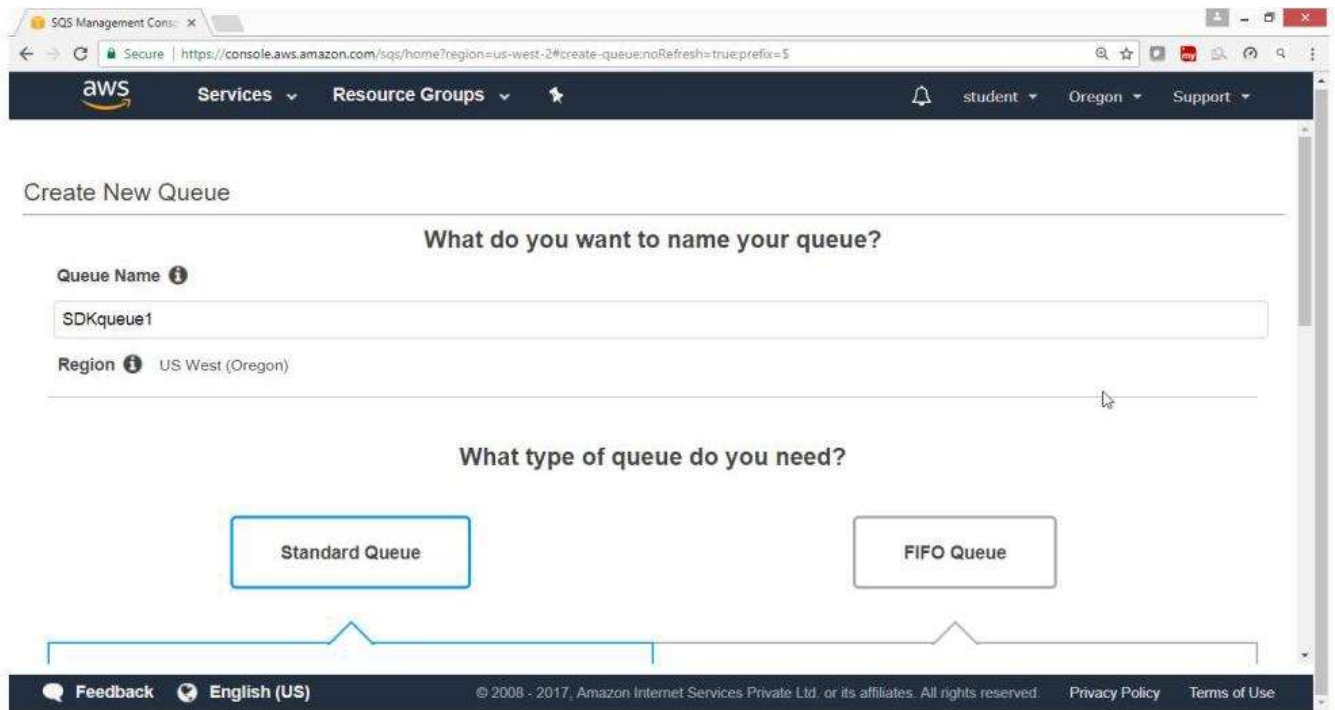


In "Create New Queue" wizard

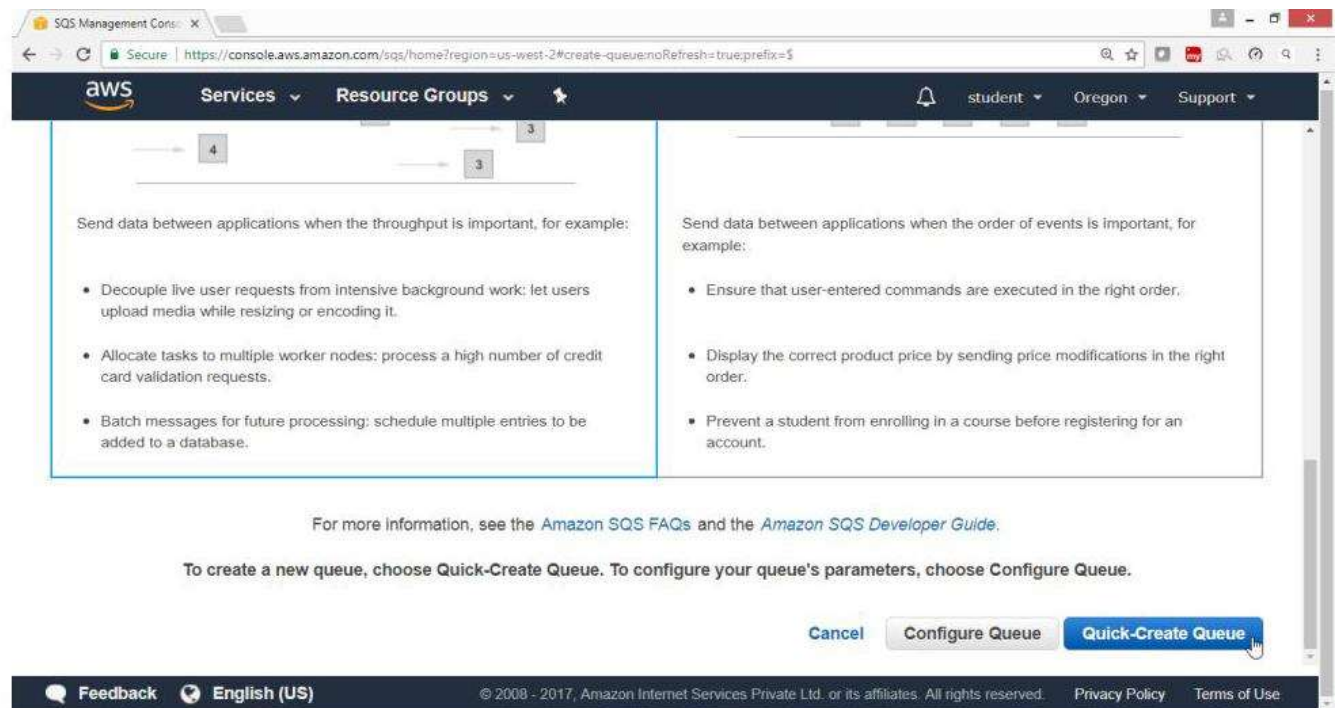
Provide the following values

Queue Name => SDKqueue1 Region => US West (Oregon)

Please leave the remaining values as default



Click on "Quick>Create Queue" button



Verify Queue is created

The screenshot shows the AWS SQS Management Console. At the top, there's a navigation bar with the AWS logo, 'Services', 'Resource Groups', and user information. Below this, a 'Create New Queue' button and a 'Queue Actions' dropdown are visible. A search bar labeled 'Filter by Prefix' is present. A table lists the queues, with 'SDKQueue1' selected. Below the table, the 'Details' tab is active, showing the queue's configuration: Name (SDKQueue1), URL, ARN, Created time, Last Updated time, Default Visibility Timeout (30 seconds), Message Retention Period (4 days), Maximum Message Size (256 KB), Receive Message Wait Time (0 seconds), Messages Available (Visible) (0), and Messages in Flight (Not Visible) (0).

Name	Queue Type	Content-Based Deduplication	Messages Available	Messages in Flight	Created
SDKQueue1	Standard	N/A	0	0	2017-11-12 18:42:48 GMT+05:30

1 SQS Queue selected

Details | Permissions | Redrive Policy | Monitoring | Tags | Encryption

Name: SDKQueue1
URL: https://sqs.us-west-2.amazonaws.com/523251683217/SDKQueue1
ARN: arn:aws:sqs:us-west-2:523251683217:SDKQueue1
Created: 2017-11-12 18:42:48 GMT+05:30
Last Updated: 2017-11-12 18:42:48 GMT+05:30

Default Visibility Timeout: 30 seconds
Message Retention Period: 4 days
Maximum Message Size: 256 KB
Receive Message Wait Time: 0 seconds
Messages Available (Visible): 0
Messages in Flight (Not Visible): 0

Select the queue

Drop down "Queue Action"

Select "Send Message"

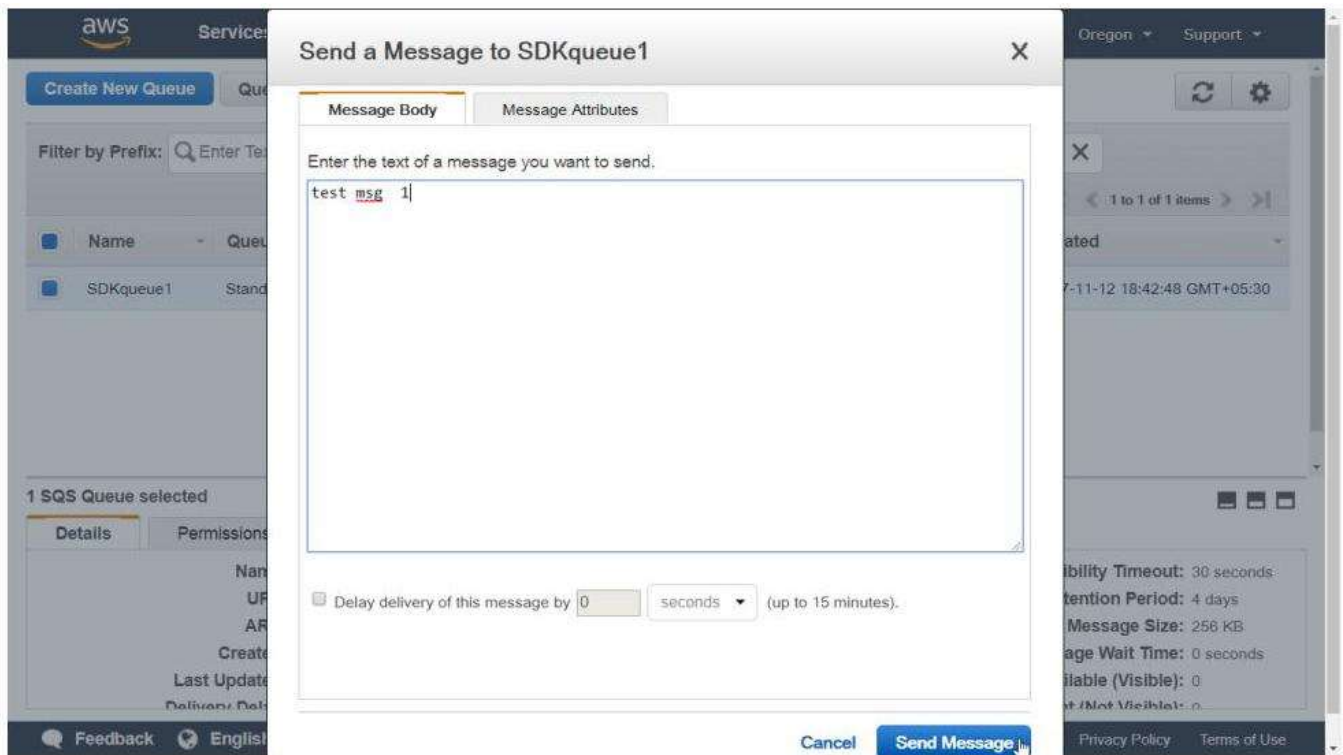
This screenshot shows the same AWS SQS Management Console interface, but with the 'Queue Actions' dropdown menu open. The menu options are: 'Send a Message' (highlighted with a mouse cursor), 'View/Delete Messages', 'Configure Queue', 'Add a Permission', 'Purge Queue', 'Delete Queue', and 'Subscribe Queue to SNS Topic'. The background shows the queue details for 'SDKQueue1'.

Queue Actions

- Send a Message
- View/Delete Messages
- Configure Queue
- Add a Permission
- Purge Queue
- Delete Queue
- Subscribe Queue to SNS Topic

From "Send a Message to SDKQueue1" wizard

In Message Body type the Message



Note: Message Size should not be more than 64K

Click on "Send Message" then select "Close"

2) To view the message

Select the queue

Drop down Queue Action button

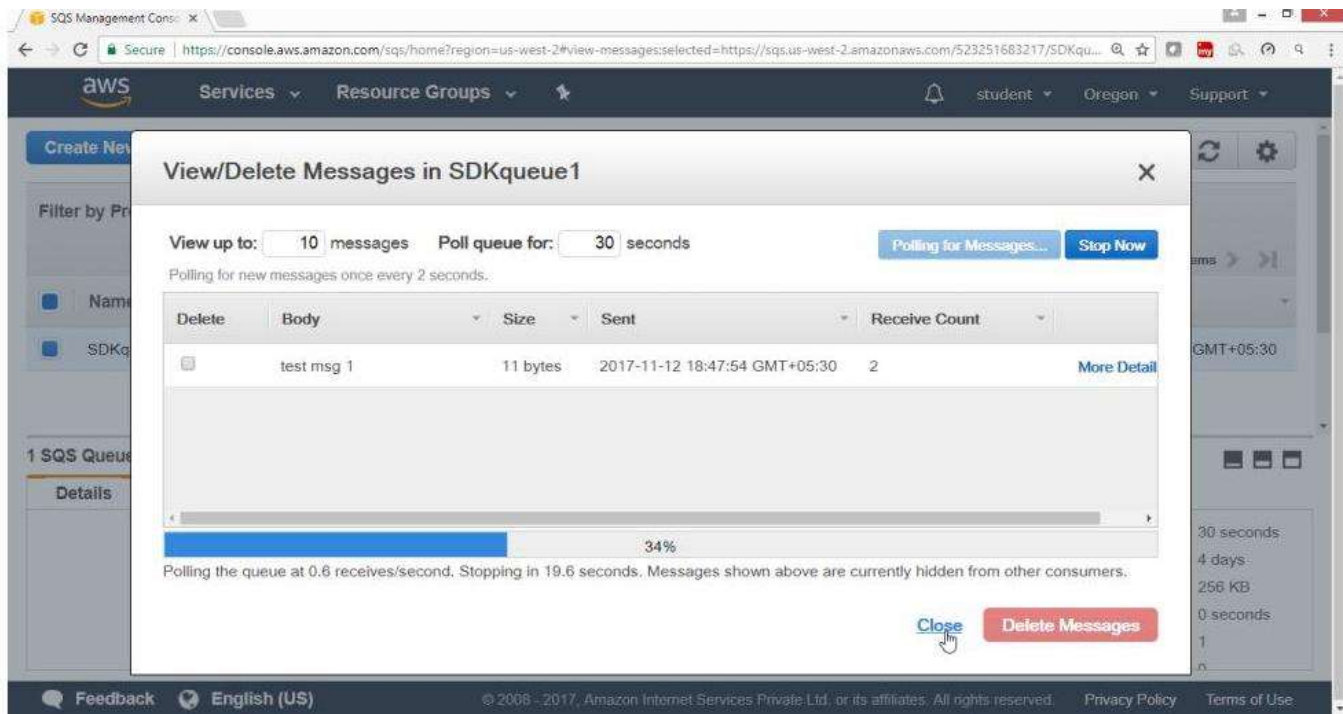
Select the option "**View/Delete Message**"

The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, 'Services', 'Resource Groups', and user information. Below this, a 'Queue Actions' dropdown menu is open for the selected queue 'SDKQueue1'. The menu options are: 'Send a Message', 'View/Delete Messages' (highlighted with a mouse cursor), 'Configure Queue', 'Add a Permission', 'Purge Queue', 'Delete Queue', and 'Subscribe Queue to SNS Topic'. In the background, a table lists the queue's details: Name (SDKQueue1), URL, ARN, Created time, Last Updated time, Delivery Delay, Default Visibility Timeout (30 seconds), Message Retention Period (4 days), Maximum Message Size (256 KB), Receive Message Wait Time (0 seconds), and Messages Available (Visible) (1).

Click "**Start Polling for Message**"

The screenshot shows the 'View/Delete Messages in SDKQueue1' dialog box. It has a title bar with a close button. Inside, there are input fields for 'View up to: 10 messages' and 'Poll queue for: 30 seconds'. Below these, a button labeled 'Polling for Messages...' is highlighted with a mouse cursor, and a 'Stop Now' button is next to it. A progress bar at the bottom shows 0% completion. At the bottom right, there are 'Close' and 'Delete Messages' buttons. The background shows the same queue details as the previous screenshot.

Verify **message** is in the queue

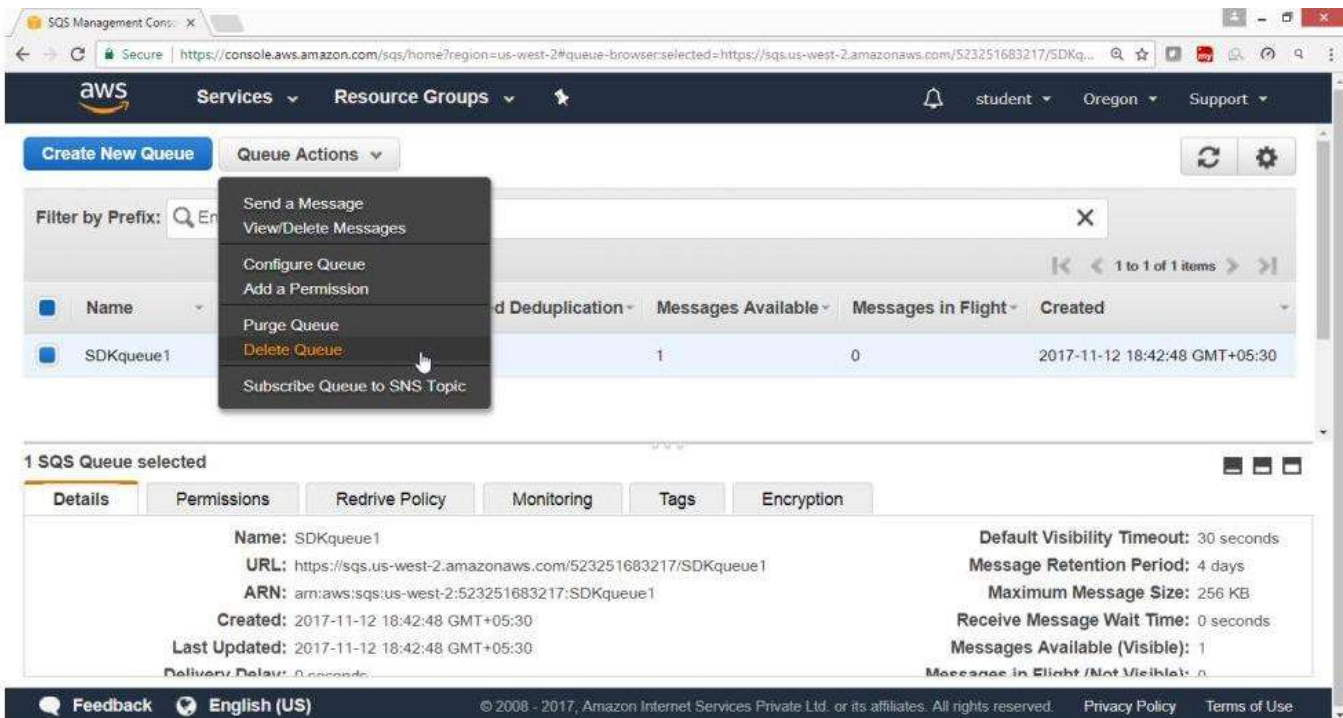


3) To delete the message

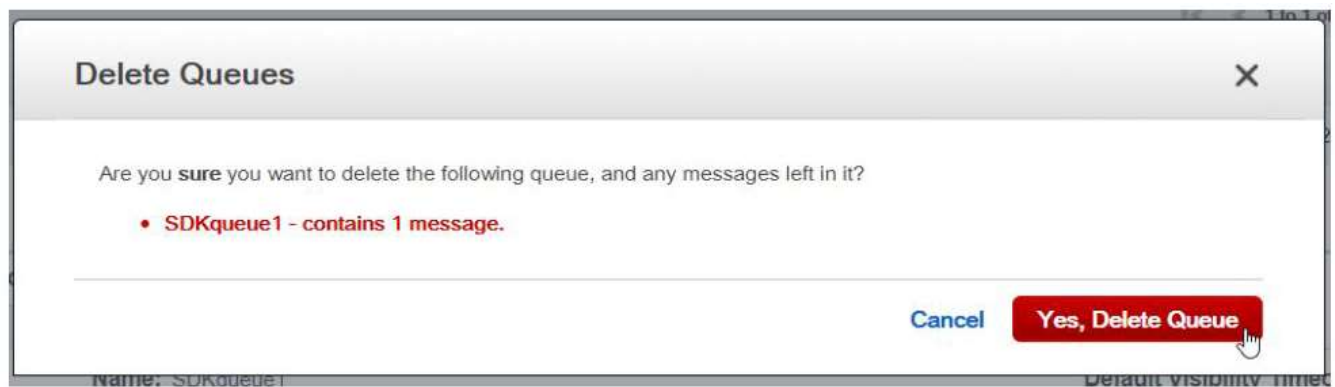
Select the Queue

Drop Down Queue Action

Select "Delete Message"



Confirm



What is SQS?

Amazon SQS is a web service that gives you access to a message queue that can be used to store messages while waiting for a computer to process them

Amazon SQS is a distributed queue system that enables web service applications to quickly and reliably queue messages that one component in the application generates to be consumed by another component. A queue is a temporary repository for messages that are awaiting processing.

Using Amazon SQS, you can decouple the components of an application so they run independently, with Amazon SQS easing message management between components.

Any component of a distributed application can store messages in a fail-safe queue. Message can contain up to 256 KB text in any format. Any component can later retrieve the messages programmatically using the Amazon SQS API.

The queue act as a buffer between the component producing and saving data, and the component receiving the data for processing.

This means the queue resolves issues that arise if the producer is producing work faster than the consumer can process it, or if the producer or consumer are only intermittently connected to the network.

What are the Queue Types?

There are two types of Queue namely: -

- Standard Queue (default)
- FIFO Queues

Standard Queues

Amazon SQS offers standard as the default queue type. A standard queue lets you have a nearly-unlimited number of transactions per second.



Standard queues guarantee that a message is delivered at least once.

However, occasionally (because of the highly distributed architecture that allows high throughput), more than one copy of a message might be delivered out of order. Standard queues provide the best effort ordering which ensures that messages are generally delivered in the same order as they sent.

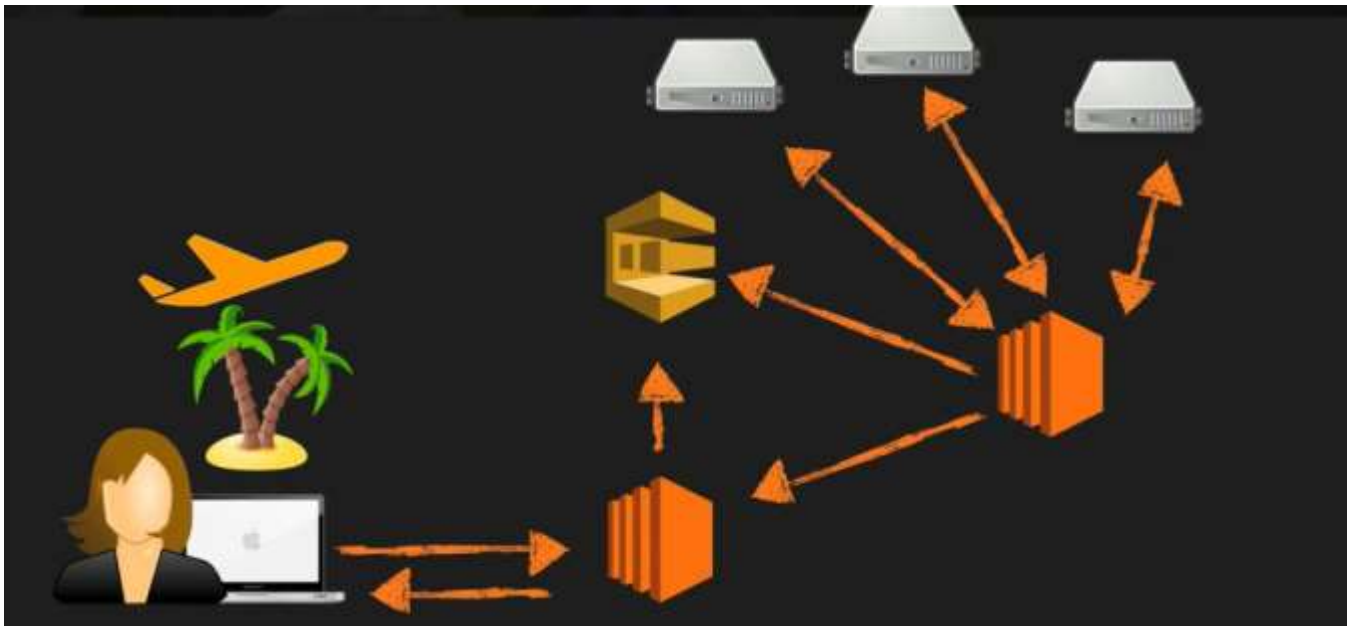
FIFO Queues

The FIFO queue complements the standard queue. The most important features of this queue type are FIFO (First-In-First-Out) delivery and exactly once processing: The order in which messages are sent and received is strictly preserved and a message is deleted; duplicates are not introduced into the queue.

FIFO queues also support message groups that allow multiple ordered message groups within a single queue. FIFO queues are limited to 300 transactions per second (TPS) but have all the capabilities of standard queues.



How SQS will be more effective in Ecommerce Travel Website?



How to use Amazon SQS?

Amazon SQS is a message passing mechanism that is used for communication between different connectors that are connected with each other. It also acts as a communicator between various components of Amazon. It keeps all the different functional components together. This functionality helps different components to be loosely coupled and provide an architecture that is more failure resilient system.

What are the advantages of messaging queues to decouple components?

Messaging queues is a very good approach to build a decoupled system. In a messaging queue there is asynchronous communication. The components are connected by using a queue or a buffer.

It provides following advantages:

Concurrency: More than one component can concurrently access the messaging queue.

High Availability: Since messages are persisted in the queue, a component can re-read a message even in case of failure. This leads to higher availability of the whole system.

Load Spikes: In case of sudden increase in load, a messaging queue can gracefully handle the scenario. It will collect all the messages and process these asynchronously.

How can we implement Message Queue based system in AWS?

Following techniques can be used to build a Message Queue based system in AWS: **Amazon SQS:** We can use Amazon SQS as a queue/buffer between components. In this way different components can be isolated. **Service Interface:** We can design every component to expose a service interface and make it responsible for its own scalability.

The component will interact with other components asynchronously by using SQS. **Machine Image:** We can put the logical parts of software in the Amazon Machine image for that component, so that it can be deployed in an automated manner. **Stateless:** Also, is it important to make the applications stateless for asynchronous communication.

What is SWF?

Amazon Simple Workflow Service (Amazon SWF) is a web service that makes it easy to coordinate work across distributed application components.

Amazon SWF enables applications for a range of use cases, including media processing, web application back-ends, business process workflows, and analytics pipelines, to be designed as a coordination of tasks. Tasks represent invocations of various processing steps in an application which can be performed by executable code, web service calls, human actions, and scripts.

Compare SWF Vs SQS?

- SQS has a retention period of 14 days, SWF up to 1 year for workflow executions
- Amazon SWF presents a task-oriented API, whereas Amazon SQS offers a message-oriented API
- Amazon SWF ensures that a task is assigned only once and is never duplicated. With Amazon SQS, you need to handle the duplicated messages and may also need to ensure that a message is processed only once.
- Amazon SWF keeps track of all the tasks and events in an application. With Amazon SQS, you need to implement your own application-level tracking, especially if your application uses multiple queues.

SWF Actors

Workflow Starters - An Application that can initiate (start) a workflow. Could be your e-commerce website when placing an order or a mobile app searching for bus times

Deciders - Control the flow of activity tasks in a workflow execution. If something has finished in a workflow (or fails) a Decider decides what to do next

Activity Workers - Carry out the activity tasks

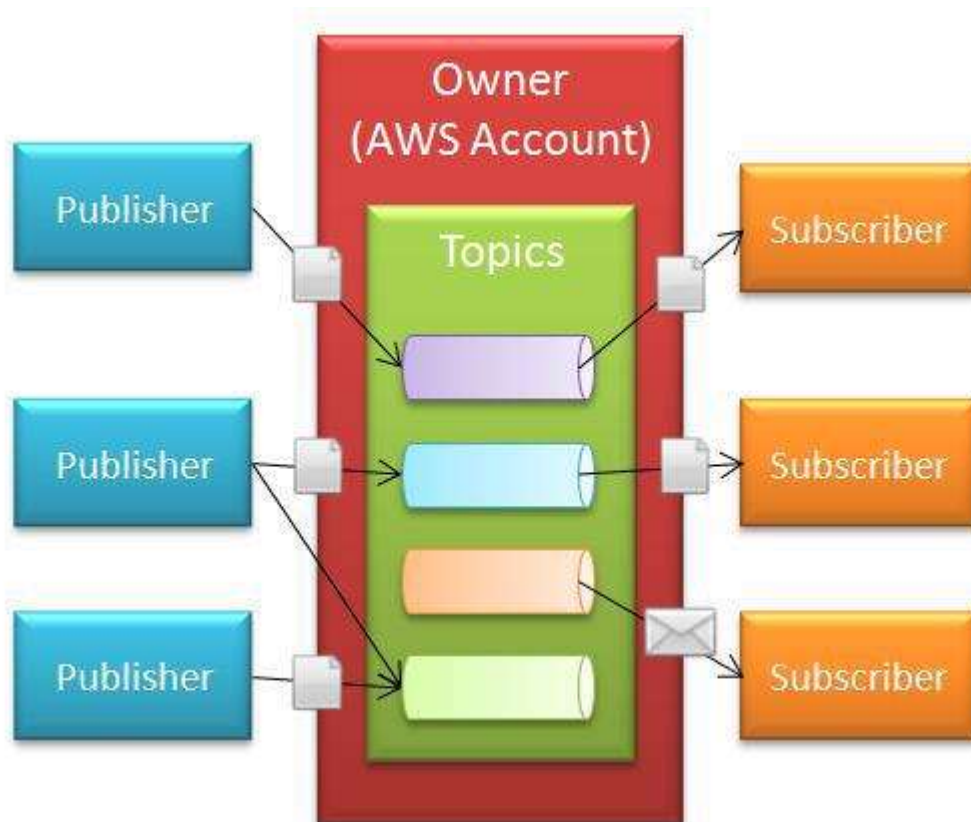


AWS Simple Notification Service

SNS Highlights

Amazon Simple Notification Service (Amazon SNS) is a web service that makes it easy to set up, operate, and send notifications from the cloud. It provides developers with a highly scalable, flexible, and cost-effective capability to publish message from an application and immediately deliver them to subscribers or other applications.

Push notifications to Apple, Google, Fire OS, and Windows devices, as well as Android devices in China with Baidu Cloud Push.



Besides pushing cloud notifications directly to mobile devices, Amazon SNS can also deliver notifications by SMS text message or email, to Amazon Simple Queue Service (SQS) queues, or to any HTTP endpoint.

SNS notifications can also trigger Lambda functions. When a message is published to an SNS topic that has a Lambda function subscribed to it, the Lambda function is invoked with the payload of the published message. The Lambda function receives the message payload as an input parameter and can manipulate the information in the message, publish the message to other SNS topics, or send the message to other AWS services.

SNS allows you to group multiple recipients using topics. A topic is an "[Access Point](#)" for allowing recipients to dynamically subscribe for identical copies of the same notification. One topic can support deliveries to multiple endpoint types - For example, you can group together iOS, Android and SMS

recipients. When you publish once to a topic, SNS delivers appropriately formatted copies of your message to each subscriber.

To prevent messages from being lost, all messages published to Amazon SNS are stored redundantly across multiple availability zones.

The benefits of SNS

- Instantaneous, push-based delivery (no polling)
- Simple APIs and easy integration with applications
- Flexible message delivery over multiple transport protocols
- Inexpensive, Pay-as-you-go model with no up-front costs
- Web-based AWS Management Console offers the simplicity of a point-and-click interface

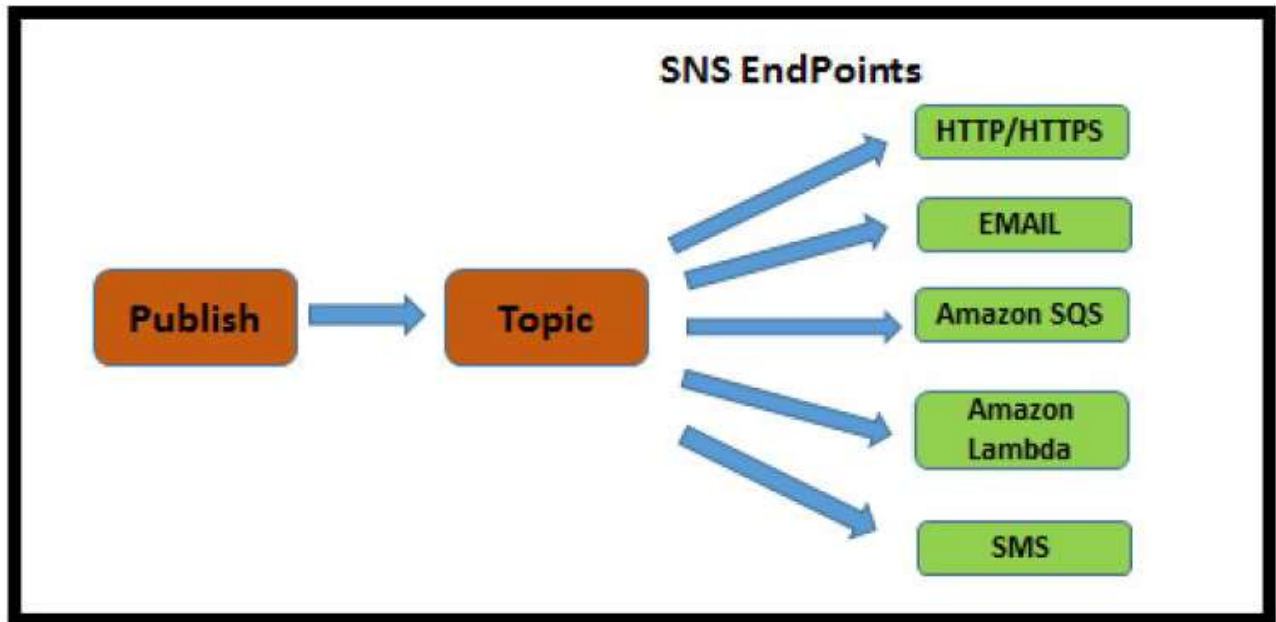
SNS Subscribers are: -

- HTTP
- HTTPS
- Email
- Email-JSON
- SQS
- Application
- Lambda

Share the SNS Configuration Step by Step?

To Configure and use Simple Notification Service (SNS)

Topology



Pre-requisites

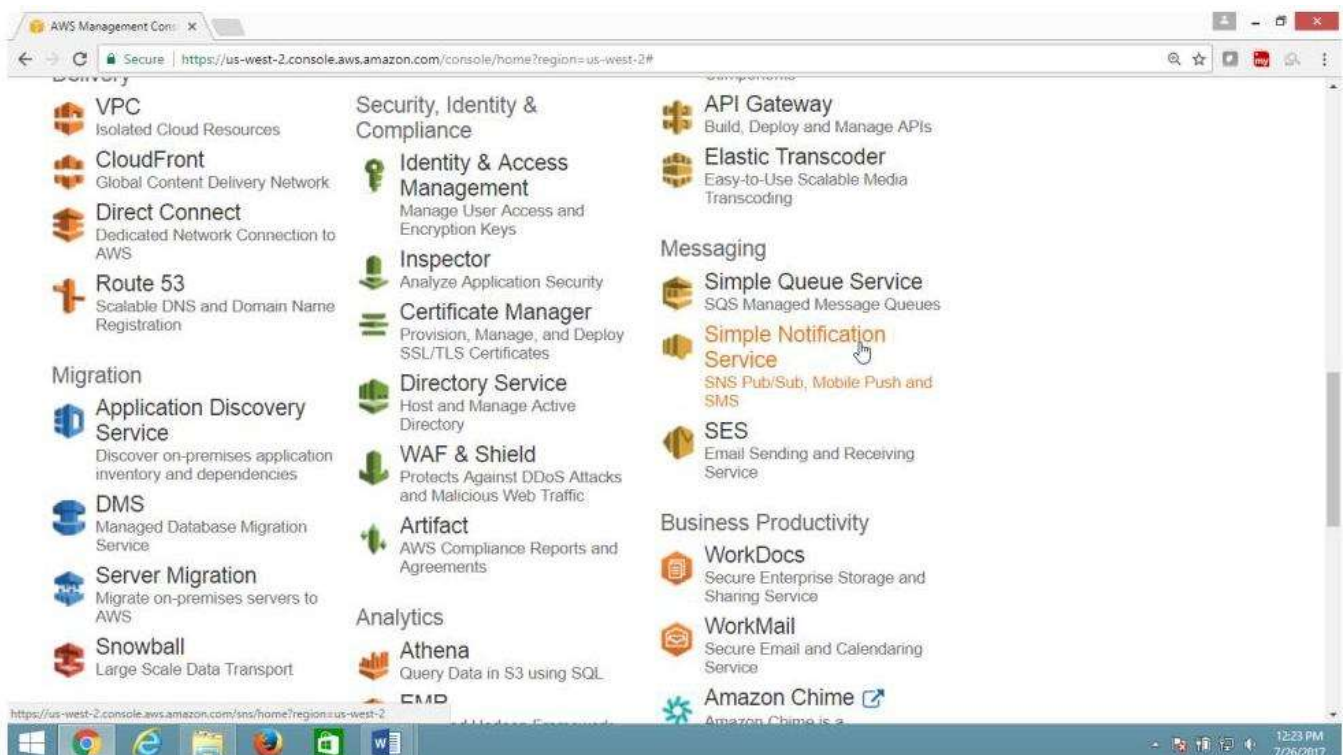
User should have AWS account, or IAM user with AmazonSNSFullAccess

To Configure SNS with following task:

- Create a Topic
- Subscribe your topic
- Verify in your mail account

Step-1) To configure Amazon Simple Notification Service (SNS)

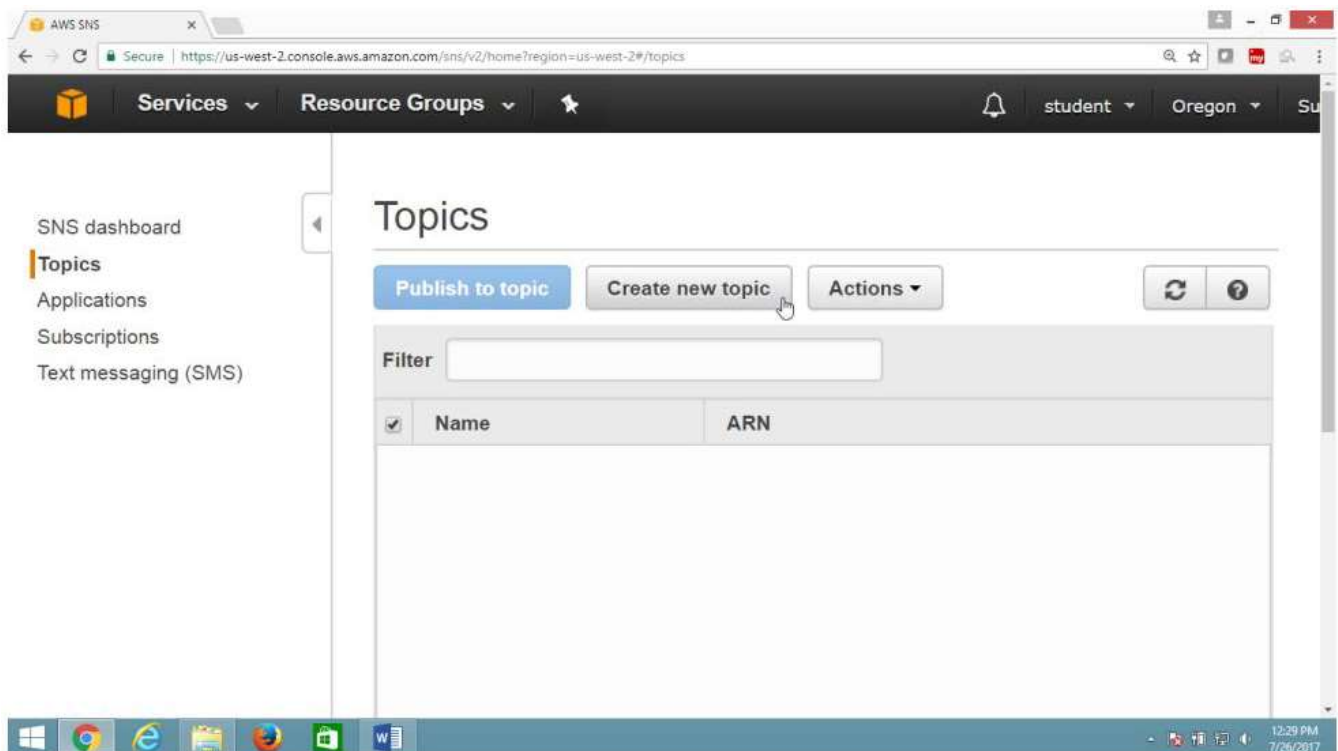
- Open **AWS Console**
- Select "**Messaging**" service"
- Click on "**Simple Notification Service**"



From "SNS Dashboard" panel

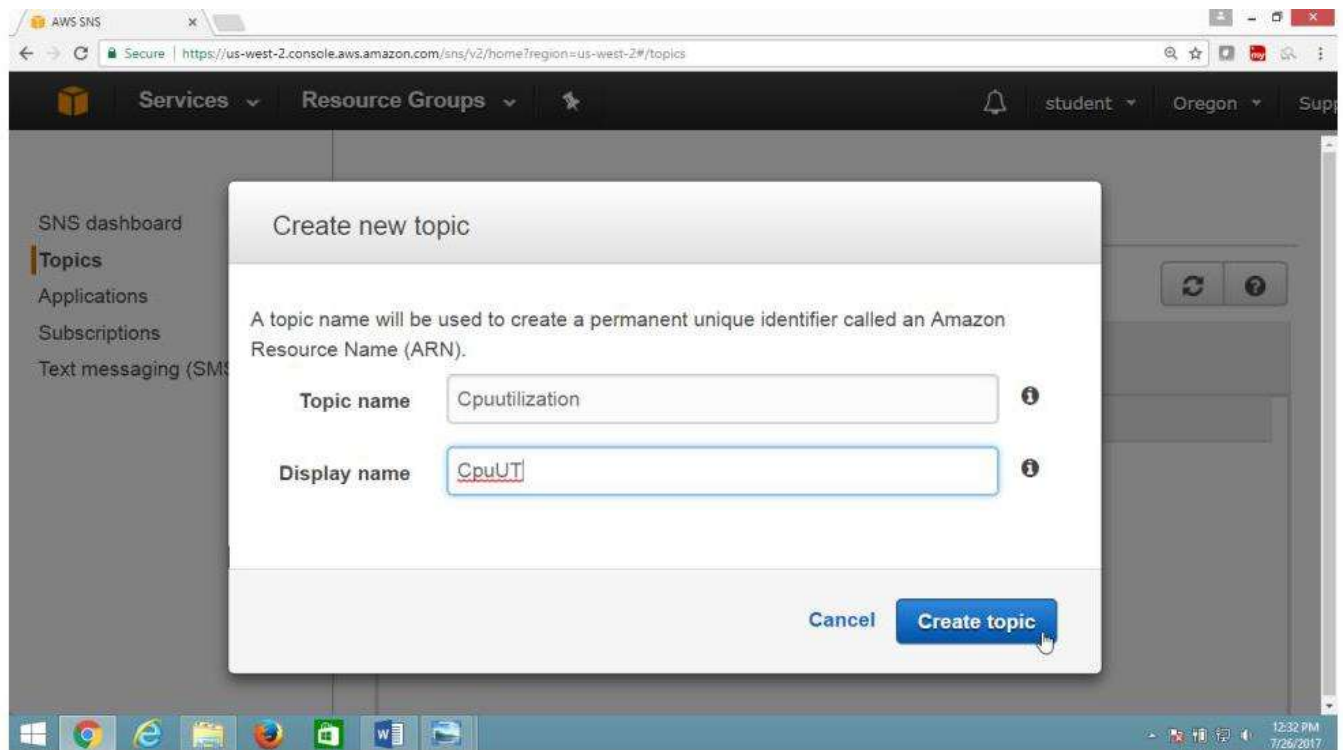
Select Topic

Click on "Create new topic" button

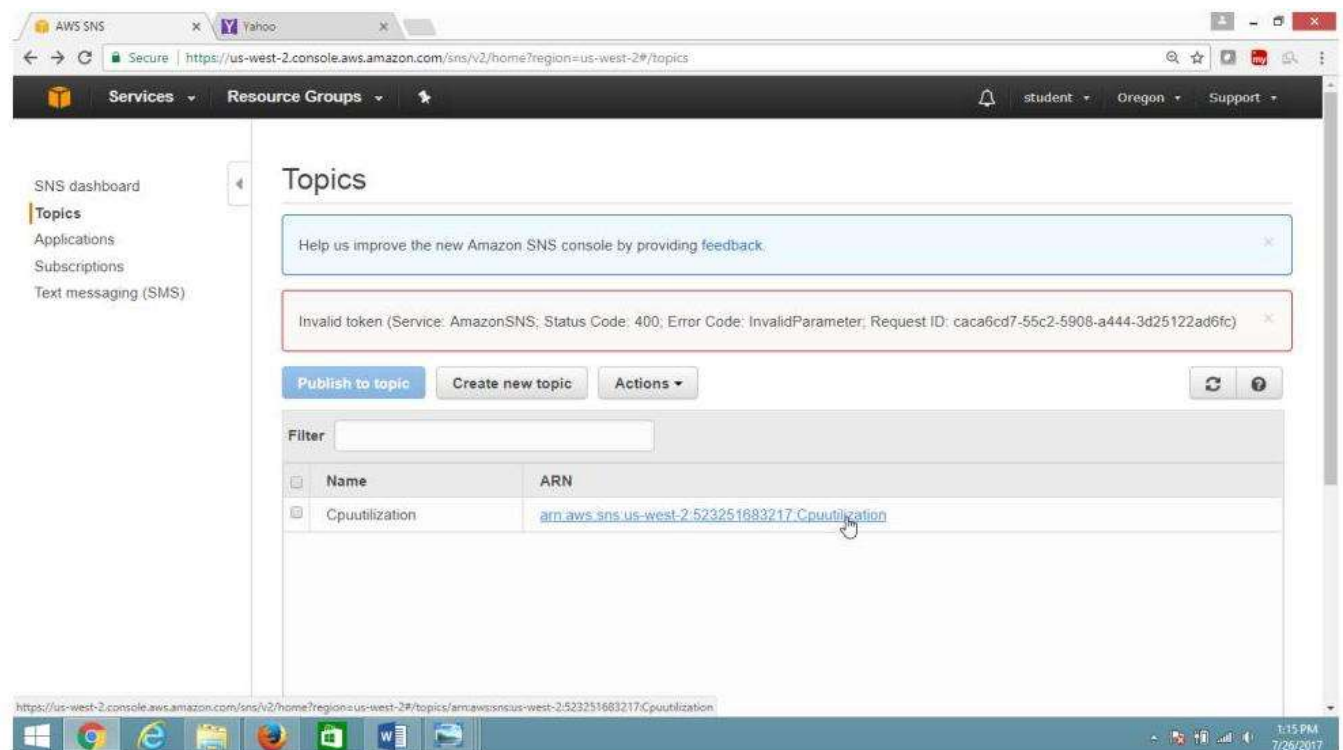


In "Create new topic" box

Provide Topic name and Display name

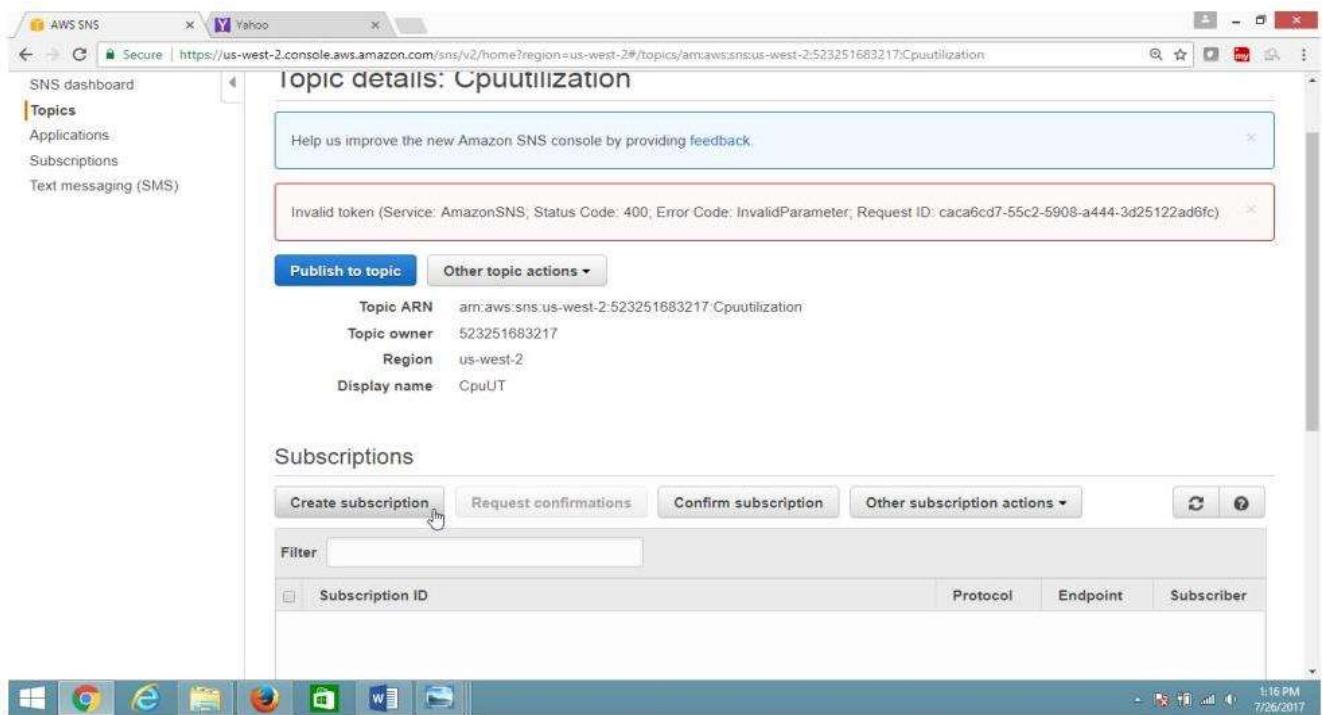


Click of ARN link



Step-2) To create Subscription

Click on "Create Subscription" button

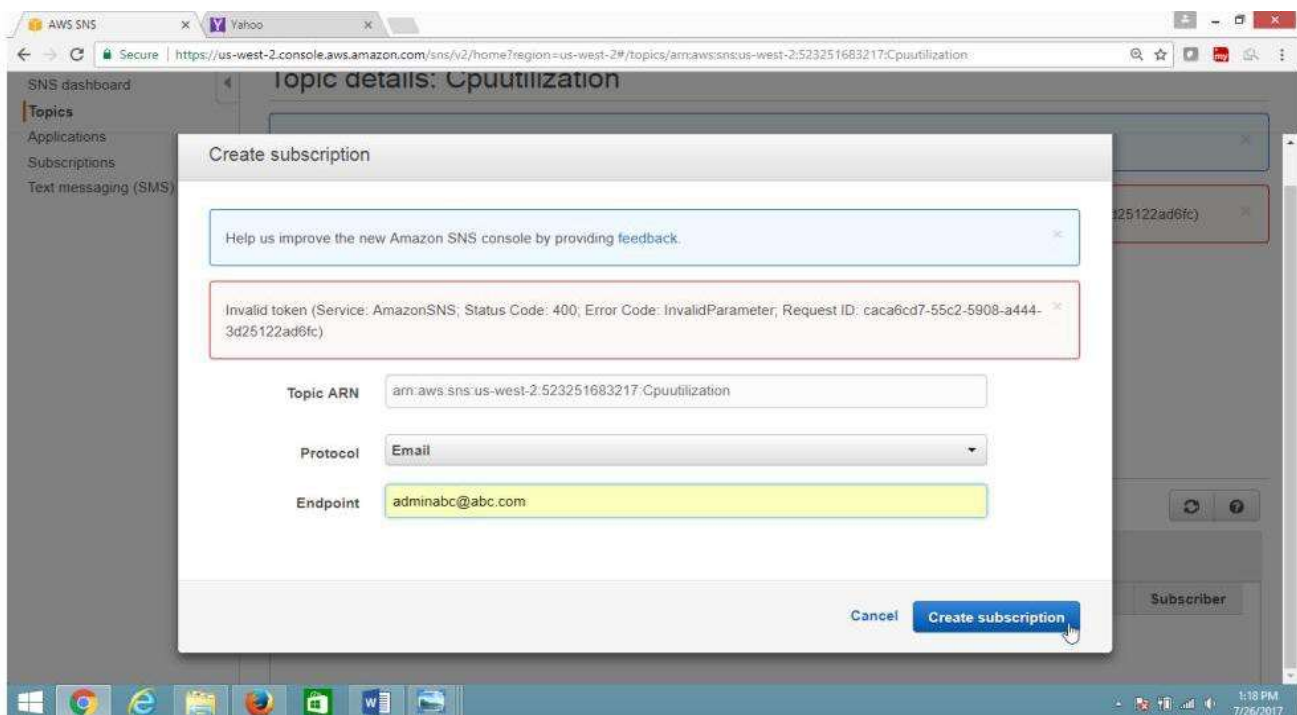


Provide values as

Protocol ->EMAIL

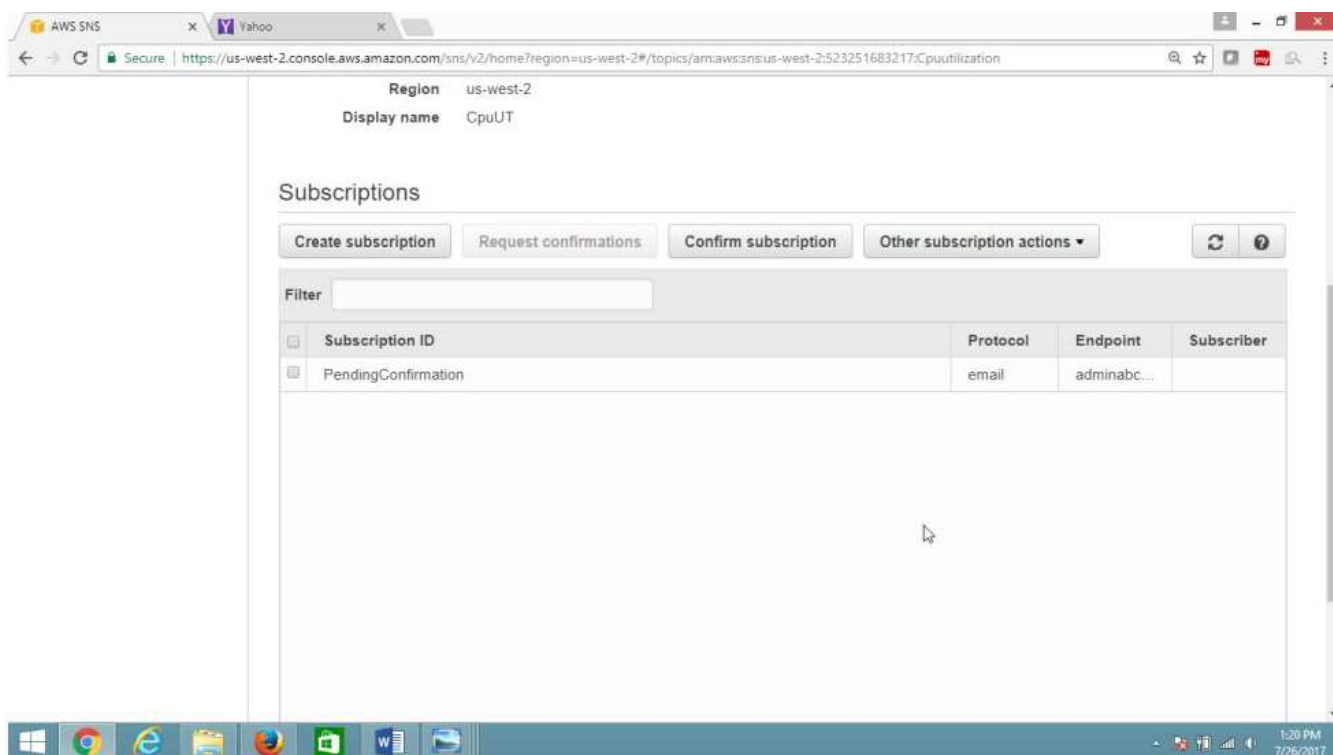
Endpoint ->adminaws@abc.com

Click "**Create Subscription**" button

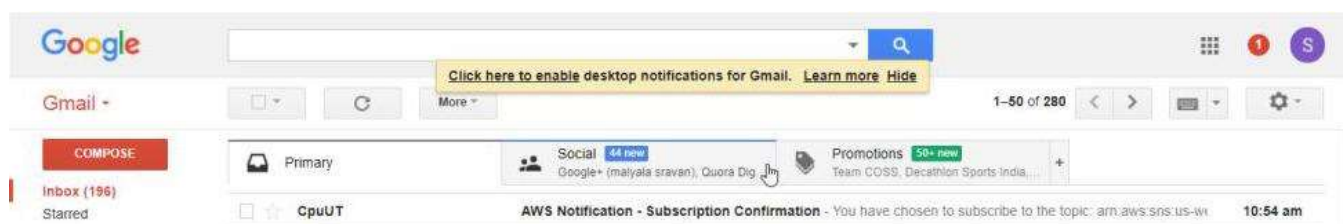


Step-3)Verification

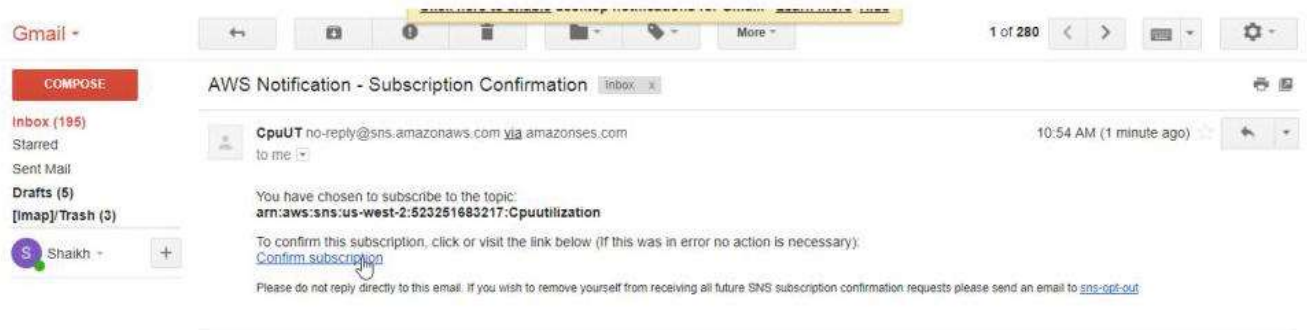
Now **Subscription** is in pending state



Go to your mail account & Click on the mail



Click on "Confirmation Message"



Now **subscription** is verified

Publish to topic

Other topic actions ▾

Topic ARN

arn:aws:sns:us-west-2:523251683217:Cpuutilization

Topic owner

523251683217

Region

us-west-2

Display name

CpuUT

Subscriptions

Create subscription

Request confirmations

Confirm subscription

Other subscription actions ▾

↺

?

Filter

<div></div>	Subscription ID	Protocol	En	Su
<div></div>	arn:aws:sns:us-west-2:523251683217:Cpuutilization:b5f880a3-4631-405e-b5e1-a37209c3...	email	ski	52:

What is the difference between Amazon SNS and Amazon SQS?

- Amazon SNS allows applications to send time-critical messages to multiple subscribers through a “push” mechanism, eliminating the need to periodically check or “poll” for updates.
- Amazon SQS is a message queue service used by distributed applications to exchange messages through a polling model and can be used to decouple sending and receiving components—without requiring each component to be concurrently available.
- Amazon SQS stands for Simple Queue Service. Whereas, Amazon SNS stands for Simple Notification Service. SQS is used for implementing Messaging Queue solutions in an application. We can decouple the applications in cloud by using SQS.
- Since all the messages are stored redundantly in SQS, it minimizes the chance of losing any message. SNS is used for implementing Push notifications to a large number of users. With SNS we can deliver messages to Amazon SQS, AWS Lambda or any HTTP endpoint. Amazon SNS is widely used in sending messages to mobile devices as well. It can even send SMS messages to cell phones.

In Short

- Both messaging services in AWS
- SNS - Push
- SQS - Polls (Pulls)

How about SNS Pricing?

- Users pay \$0.50 per 1 million Amazon SNS Requests
- \$0.06 per 100,000 Notification deliveries over HTTP
- \$0.75 per 100 Notification deliveries over SMS
- \$2.00 per 100,000 Notification deliveries over Email



AWS Simple Email Service

SES Highlights

The Amazon Simple Email Service (SES) will make it easy for you to send email with minimal setup and maximum scalability.

The Simple Email Service will provide you with performance data on your email so that you can track your status and adjust your email sending model if necessary. SES will also provide you with valuable feedback from ISPs in the form of complaints from email recipients.

You can use SES by calling the SES APIs or from the command line. You can also configure your current Mail Transfer Agent to route your email through SES using the directions contained in the SES Developer Guide.

The SES APIs are pretty simple:

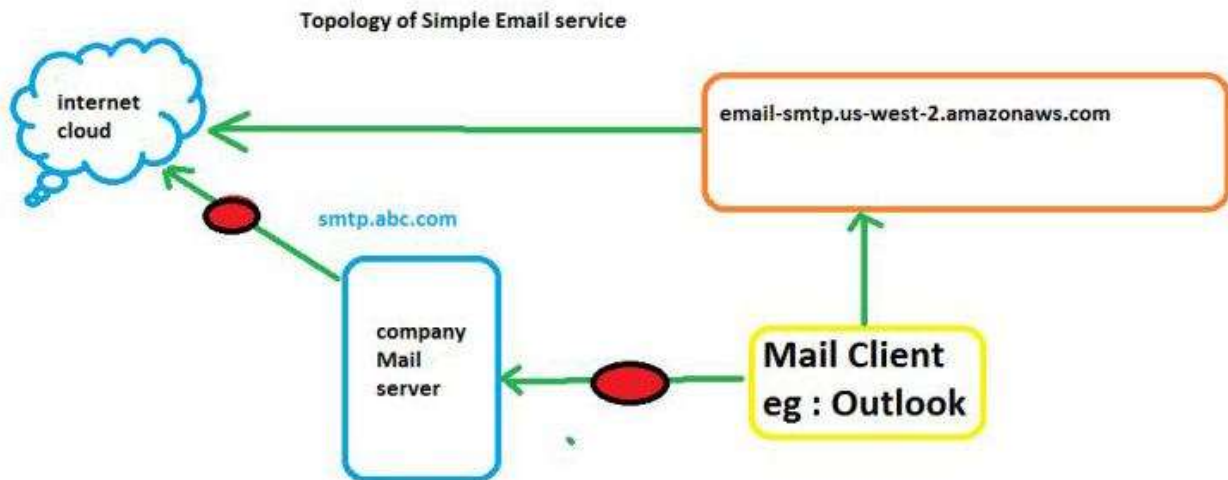
- You use `VerifyEmailAddress`, `ListVerifiedEmailAddresses`, and `DeleteVerifiedEmailAddress` to manage the list of verified email addresses associated with your account.
- You use `SendEmail` to send properly formatted emails (supplying `From`, `To`, `Subject` and a message body) and `SendRawEmail` to manually compose and send more sophisticated emails which include additional headers or `MIME` data.
- You use `GetSendQuota` and `GetSendStatistics` to retrieve your sending quotas and your statistics (delivery attempts, rejects, bounces, and complaints).

Share the SES Configuration Step by Step?

Pre-requisites

To configure and use Simple Email Service (SES)

Topology



Pre-requisites

User should have AWS account, or IAM user with Amazon SESFullAccess

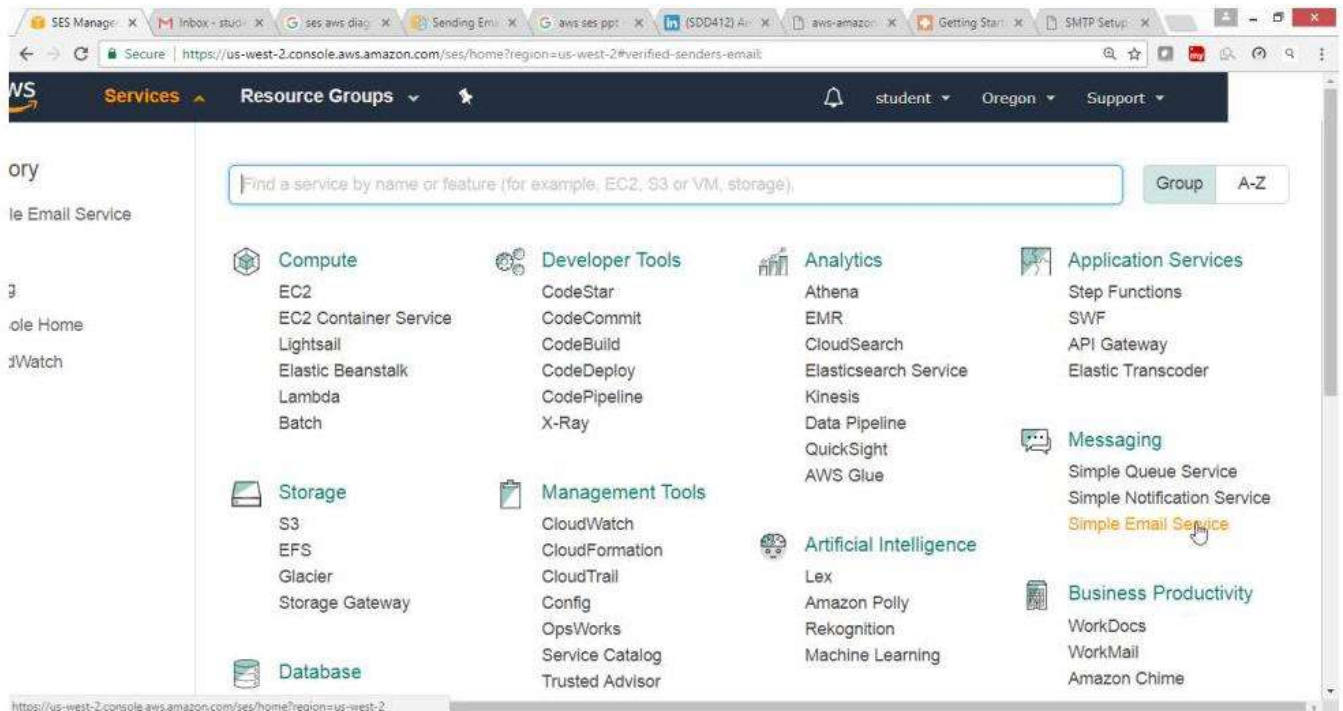
To Configure SES with following task: -

- Provide valid Mail Account
- Verify Email Address
- Configure SMTP settings
- Download the credentials keep at safe place
- Configure Mail Client for example Outlook

To use Amazon Simple E-mail Service SES

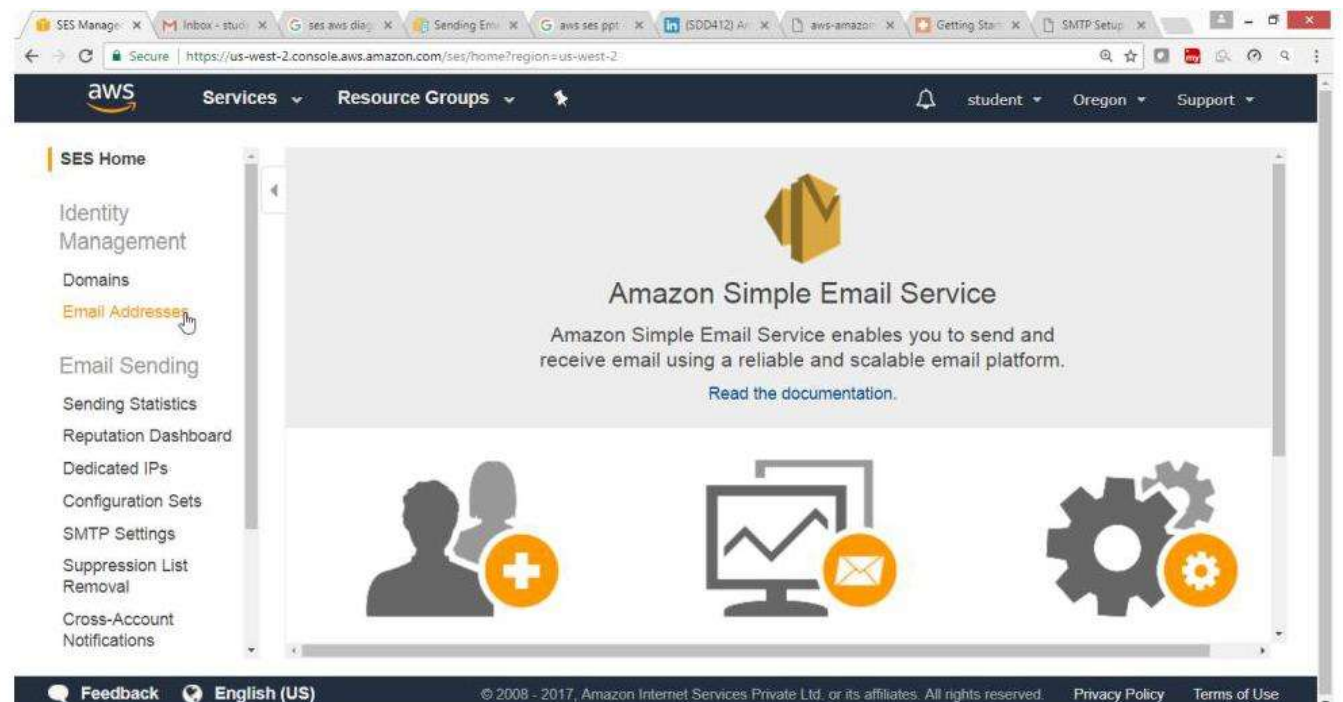
1. Create SES account

From the AWS console select service "**Application Integration**", Choose SES service

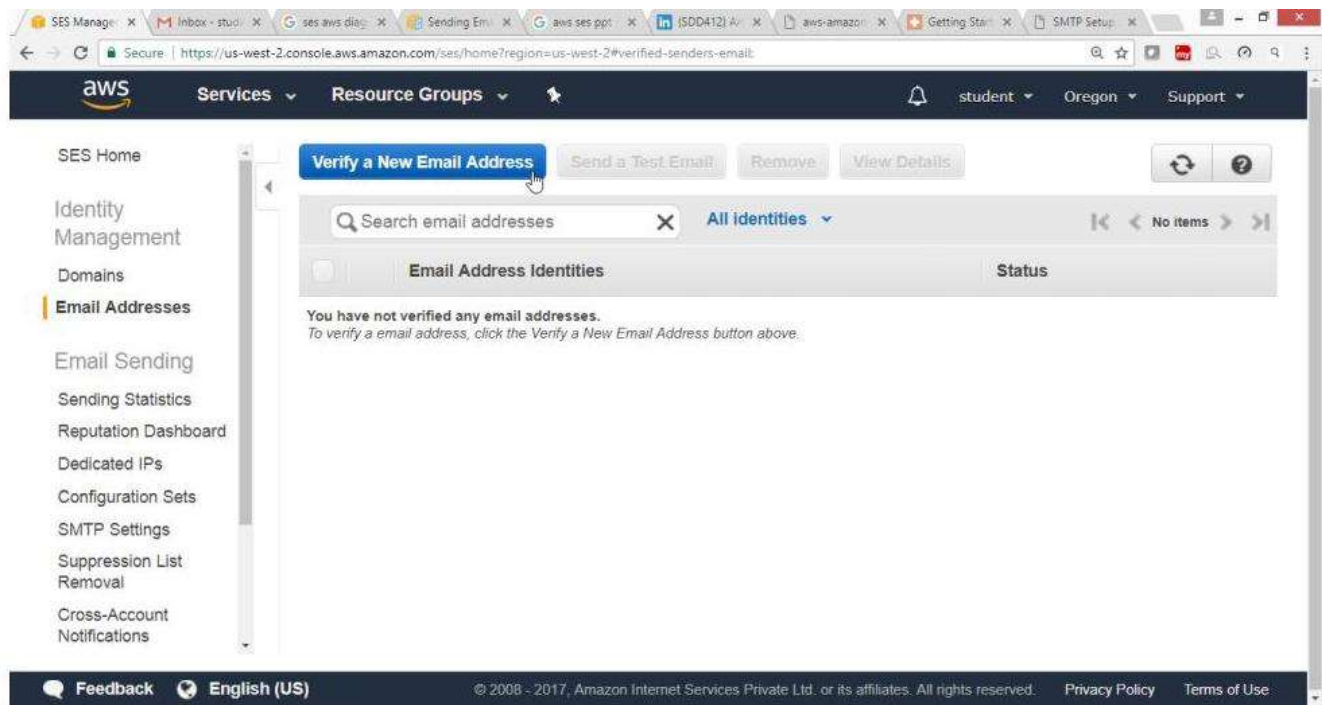


From SES Home, panel

Select "**Email Address**"



Select "**Verify a New Email Address**" button



In "[Verify a New Email Address](#)", wizard provide email id
Click "[Verify This Email Address](#)" button

The image shows a modal window titled 'Verify a New Email Address'. Inside, there is a text input field labeled 'Email Address:' containing the text 'studentcloud09@***.com'. Below the input field, there are two buttons: 'Cancel' and 'Verify This Email Address'. A hand cursor is pointing at the 'Verify This Email Address' button. The text above the input field says: 'To verify a new email address, enter it below and click the Verify This Email Address button. A verification email will be sent to the email address you entered.'

2. Now login to your companies mail account, to confirm your [email address](#)

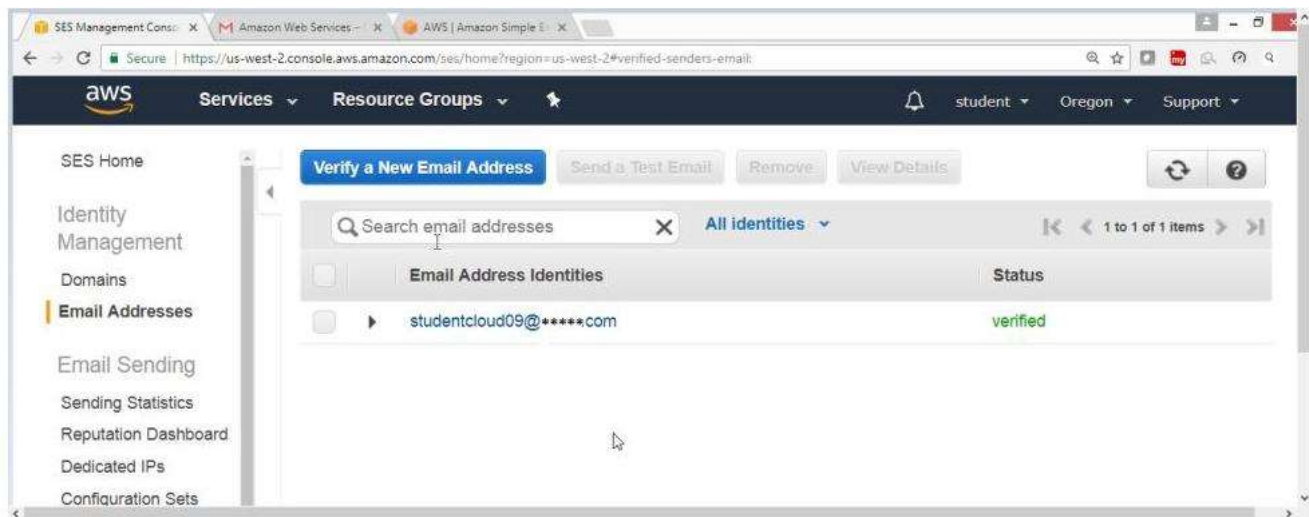
Click on "[Confirm the address using this URL. This link expires 24 hours after your original verification request](#)"

Go back to your Amazon Console, Select [SES service](#)

Under SES home dashboard select "[Email Address](#)"

Check your [email is verified](#)

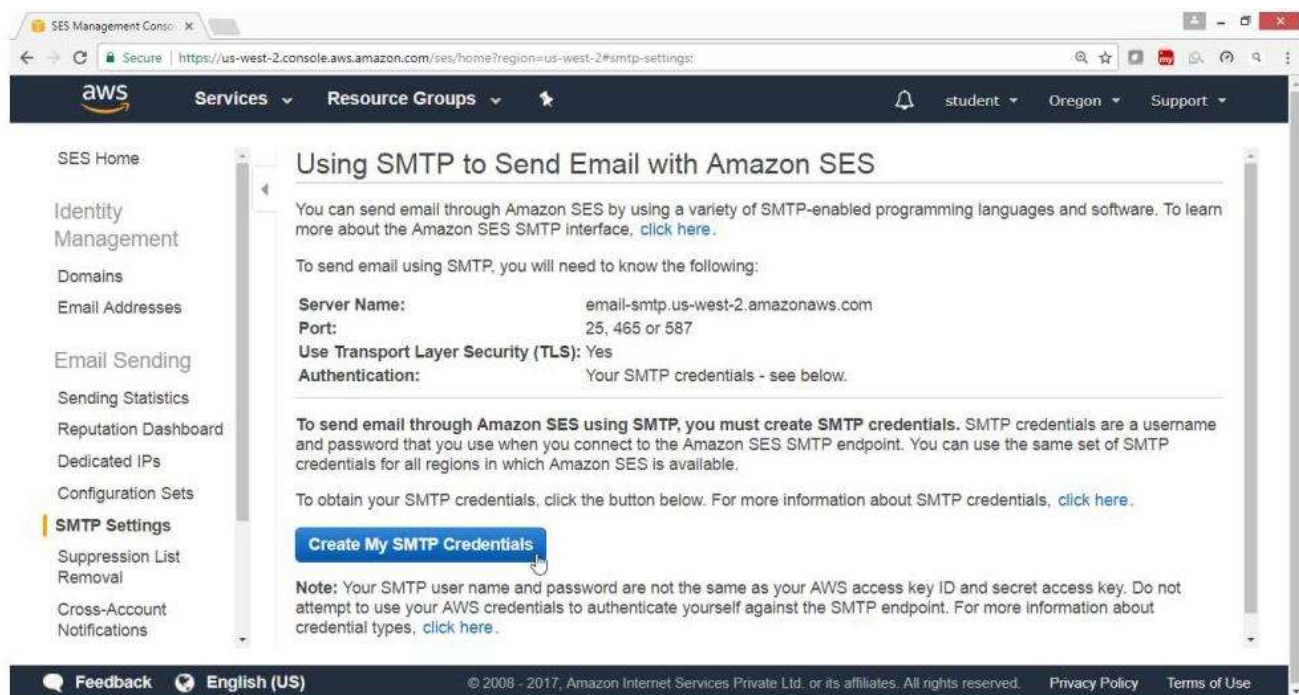
Note: If mail is not received check in spam box, you should have a valid email id



3. To configure SMTP settings

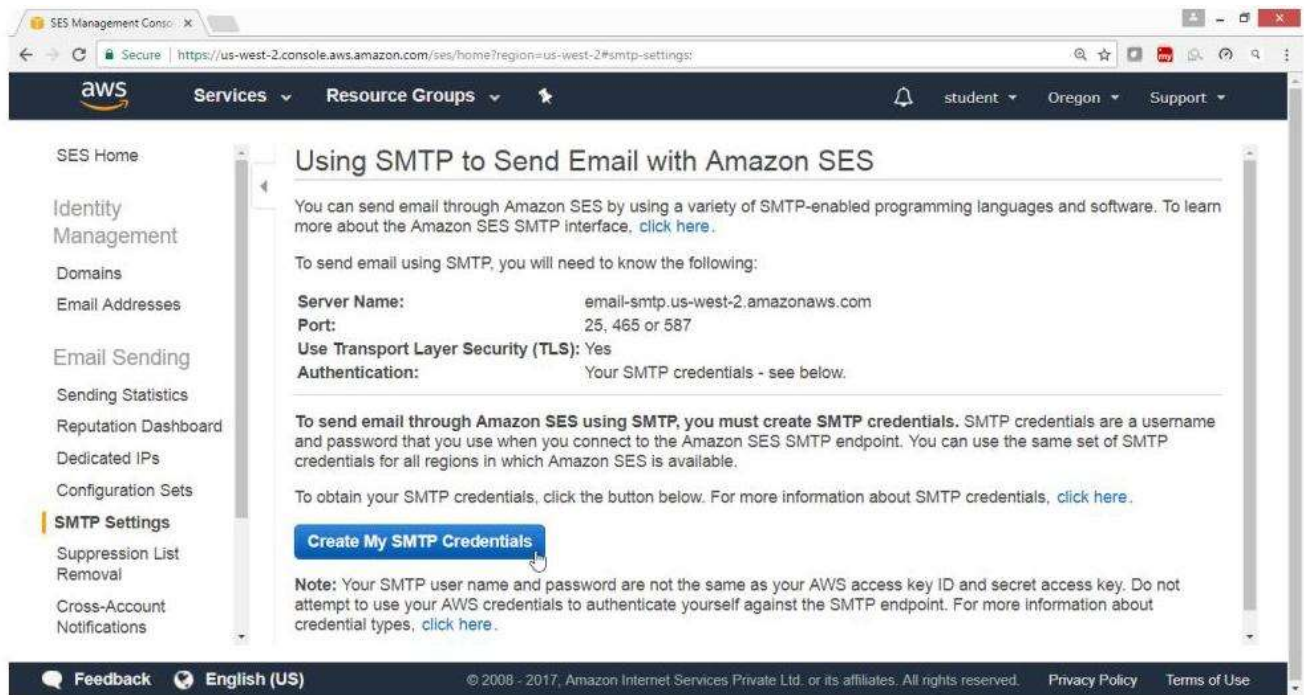
From SES Home Panel

- Select **"SMTP Setting"**
- Click on **"Create MySMTP Credentials"** button



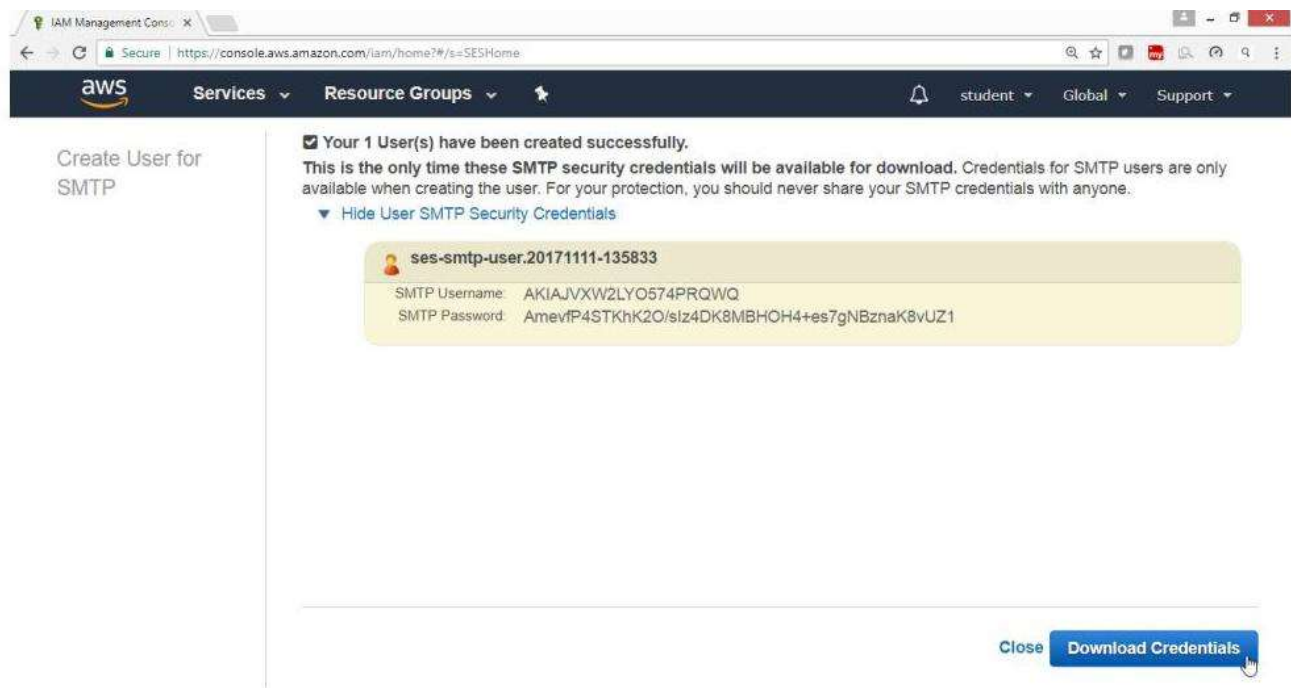
Default **IAM user Name** will be provided

Click **Create** button

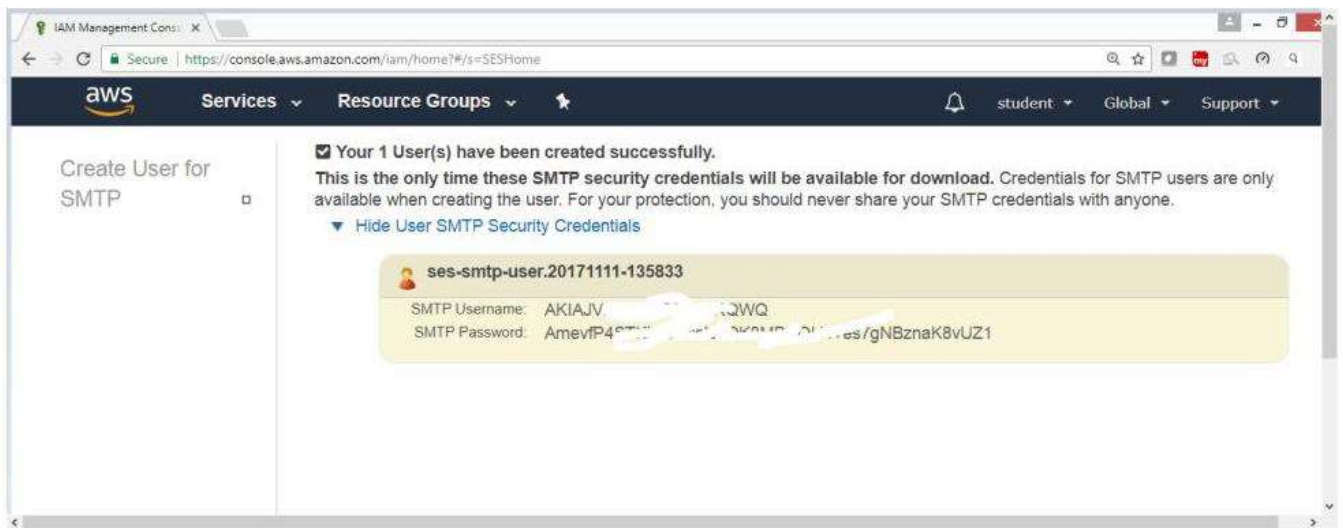


User SMTP Security credentials will be displayed

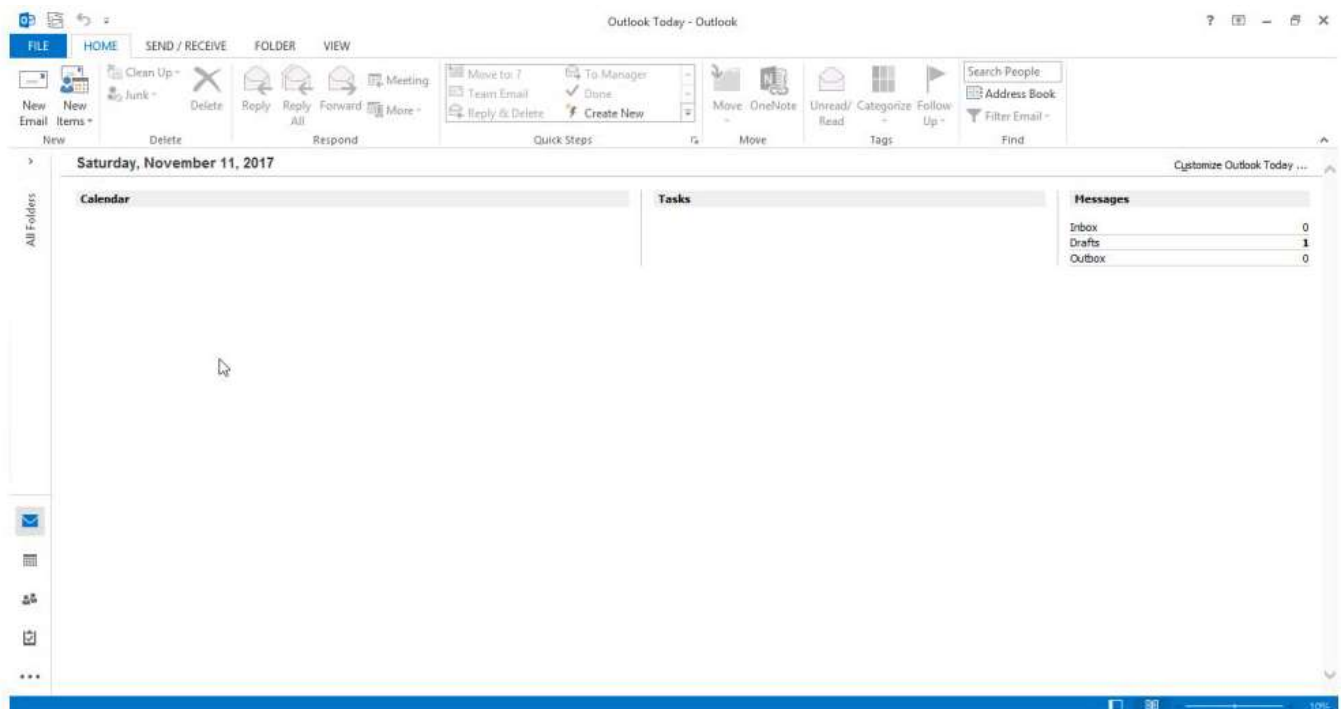
Click "[Download Credentials](#)" keep at safe place



Verify Credentials



Open Outlook



Click **Add Account**



Select **Manual Setup**

A screenshot of the 'Add Account' dialog box. The title bar says 'Add Account'. Inside, there's a section 'Auto Account Setup' with the text 'Manual setup of an account or connect to other server types.' and a question mark icon. Below this are two radio button options: 'E-mail Account' (unselected) and 'Manual setup or additional server types' (selected). The 'E-mail Account' section has four input fields: 'Your Name:' (with example 'Ellen Adams'), 'E-mail Address:' (with example 'ellen@contoso.com'), 'Password:', and 'Retype Password:'. A note below the password fields says 'Type the password your Internet service provider has given you.' At the bottom right are three buttons: '< Back', 'Next >' (highlighted with a mouse cursor), and 'Cancel'.

Select POP or IMAP, click on **Next**

Add Account



Choose Service



- ☐ **Microsoft Exchange Server or compatible service**
Connect to an Exchange account to access email, calendars, contacts, tasks, and voice mail
- ☐ **Outlook.com or Exchange ActiveSync compatible service**
Connect to a service such as Outlook.com to access email, calendars, contacts, and tasks
- ☒ **POP or IMAP**
Connect to a POP or IMAP email account

< Back

Next >

Cancel

Provide the following details

Add Account

POP and IMAP Account Settings

Enter the mail server settings for your account.

User Information

Your Name: studentcloud09

Email Address: studentcloud09@***.com

Server Information

Account Type: IMAP

Incoming mail server: imap.***.com

Outgoing mail server (SMTP): email-smtp.us-west-2.amazo

Logon Information

User Name: studentcloud09@***.com

Password: *****

☒ Remember password

☐ Require logon using Secure Password Authentication (SPA)

Mail to keep offline: All

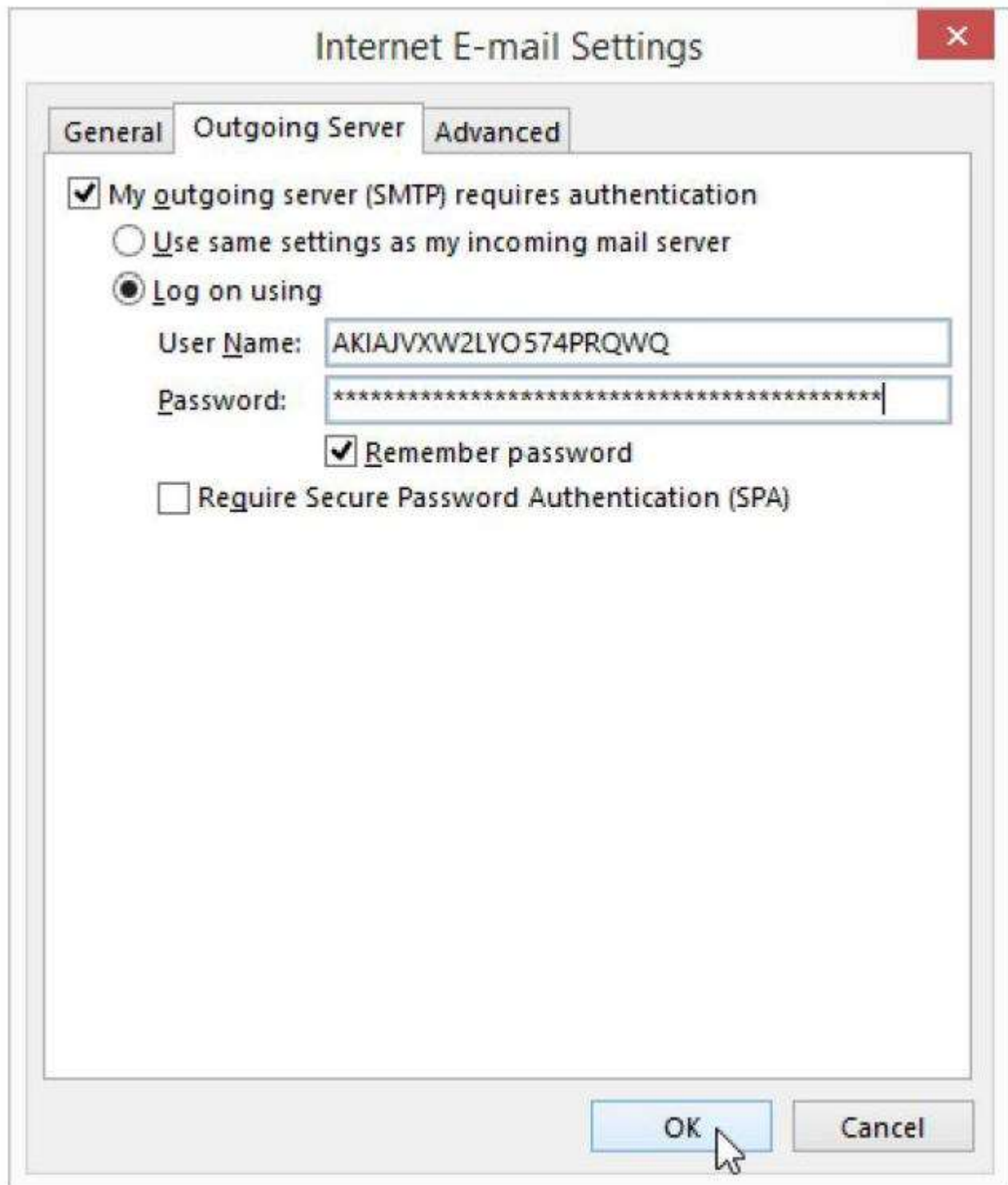
More Settings ...

< Back

Next >

Cancel

Provide the following details in Outgoing Server



The image shows a Windows-style dialog box titled "Internet E-mail Settings". It has three tabs: "General", "Outgoing Server", and "Advanced". The "Outgoing Server" tab is selected. Inside the tab, there are several options: a checked checkbox for "My outgoing server (SMTP) requires authentication", a radio button for "Use same settings as my incoming mail server", and a selected radio button for "Log on using". Below these are text boxes for "User Name" (containing "AKIAJVXW2LYO574PRQWQ") and "Password" (containing asterisks). There are also checkboxes for "Remember password" (checked) and "Require Secure Password Authentication (SPA)" (unchecked). At the bottom right are "OK" and "Cancel" buttons, with a mouse cursor pointing at the "OK" button.

Internet E-mail Settings

General Outgoing Server Advanced

☒ My outgoing server (SMTP) requires authentication

☐ Use same settings as my incoming mail server

☒ Log on using

User Name: AKIAJVXW2LYO574PRQWQ

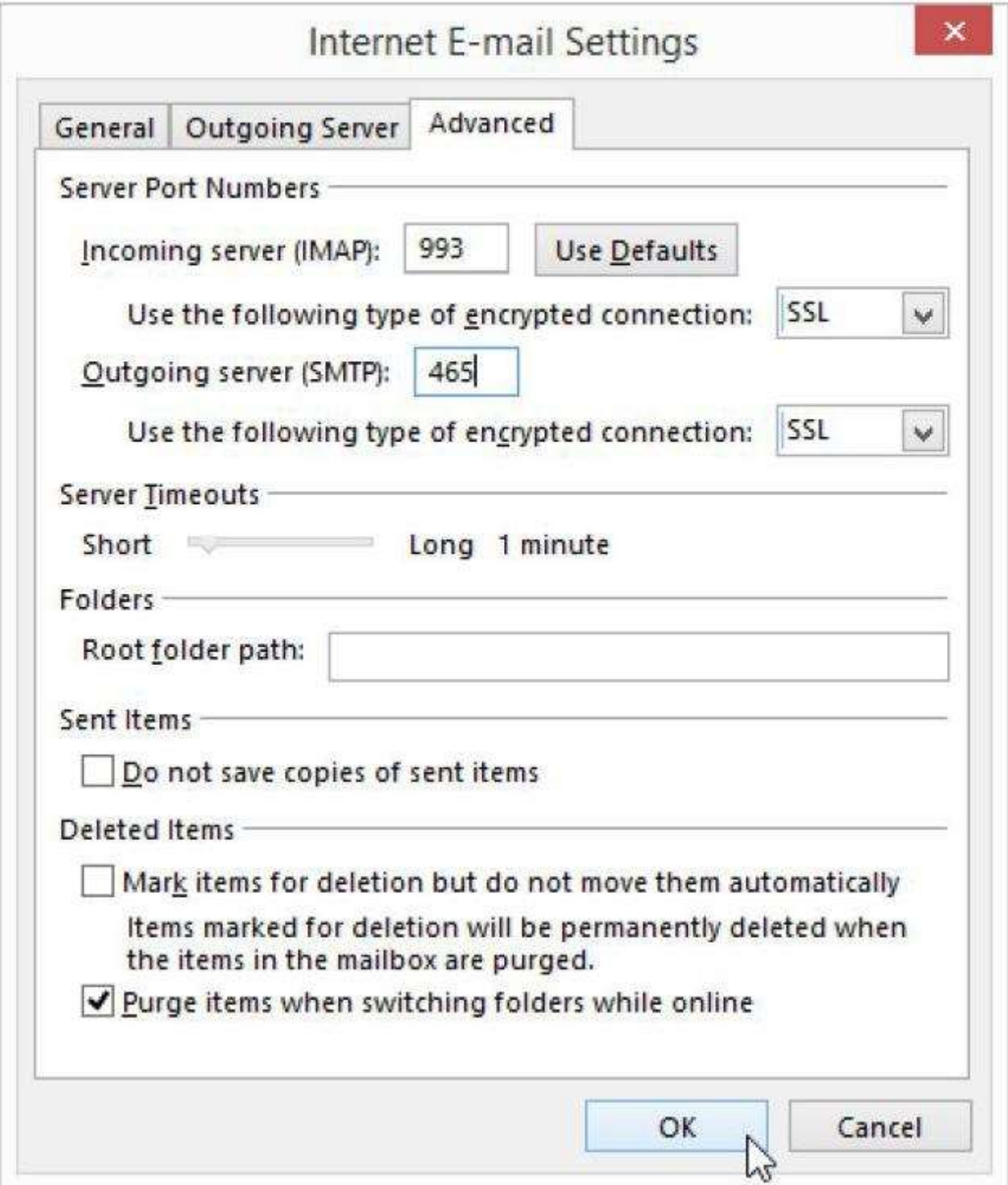
Password: *****

☒ Remember password

☐ Require Secure Password Authentication (SPA)

OK Cancel

Provide the following details in Advance

The image shows a 'Internet E-mail Settings' dialog box with a title bar containing a close button (X). It has three tabs: 'General', 'Outgoing Server', and 'Advanced'. The 'Advanced' tab is selected. The 'Server Port Numbers' section includes a text box for 'Incoming server (IMAP)' with the value '993', a 'Use Defaults' button, a label 'Use the following type of encrypted connection:', and a dropdown menu set to 'SSL'. Below this, the 'Outgoing server (SMTP)' text box contains '465', followed by another 'Use the following type of encrypted connection:' label and a dropdown menu set to 'SSL'. The 'Server Timeouts' section features a slider between 'Short' and 'Long 1 minute'. The 'Folders' section has a 'Root folder path:' label and an empty text box. The 'Sent Items' section contains a checkbox labeled 'Do not save copies of sent items'. The 'Deleted Items' section has a checkbox 'Mark items for deletion but do not move them automatically' with explanatory text below it, and a checked checkbox 'Purge items when switching folders while online'. At the bottom right are 'OK' and 'Cancel' buttons, with a mouse cursor pointing at the 'OK' button.

Internet E-mail Settings

General Outgoing Server **Advanced**

Server Port Numbers

Incoming server (IMAP): 993 Use Defaults

Use the following type of encrypted connection: SSL

Outgoing server (SMTP): 465

Use the following type of encrypted connection: SSL

Server Timeouts

Short Long 1 minute

Folders

Root folder path:

Sent Items

☐ Do not save copies of sent items

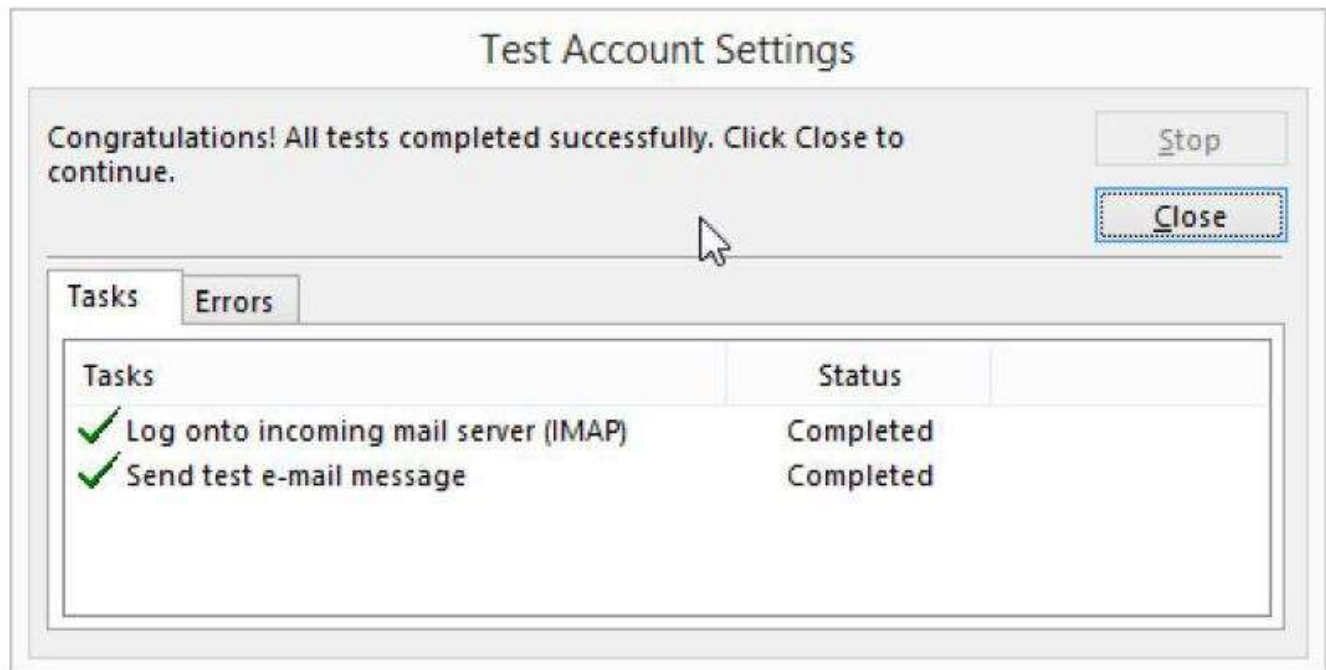
Deleted Items

☐ Mark items for deletion but do not move them automatically
Items marked for deletion will be permanently deleted when the items in the mailbox are purged.

☒ Purge items when switching folders while online

OK Cancel

Verify successfully connected



What is SES?

Amazon SES is an email platform that provides an easy, cost-effective way for you to send and receive email using your own email addresses and domains.

For example, you can send marketing emails such as special offers, transactional emails such as order confirmations, and other types of correspondence such as newsletters. When you use Amazon SES to receive mail, you can develop software solutions such as email autoresponders, email unsubscribe systems, and applications that generate customer support tickets from incoming emails. You only pay for what you use, so you can send and receive as much or as little email as you like.

Why use Amazon SES?

Building a large-scale email solution is often a complex and costly challenge for a business. You must deal with infrastructure challenges such as email server management, network configuration, and IP address reputation. Additionally, many third-party email solutions require contract and price negotiations, as well as significant up-front costs. Amazon SES eliminates these challenges and enables you to benefit from the years of experience and sophisticated email infrastructure Amazon.com has built to serve its own large-scale customer base.



Analytics

Amazon Athena Query Data in S3 using SQL	Amazon EMR Hosted Hadoop Framework	Amazon CloudSearch Managed Search Service	
Amazon Elasticsearch Service Run and Scale Elasticsearch Clusters	Amazon Kinesis Work with Real-time Streaming Data	Amazon Kinesis Fast, Simple, Cost-Effective Data Warehousing	
Amazon Kinesis Fast Business Analytics Service	AWS Data Pipeline Orchestration Service for Periodic, Data-Driven Workflows	Amazon Glue Prepare and Load Data	



Analytics

What is Amazon EMR?

Amazon Elastic MapReduce (Amazon EMR) is a managed cluster platform that simplifies running big data frameworks, such as Apache Hadoop and Apache Spark, on AWS to process and analyze vast amounts of data. By using these frameworks and related open-source projects, such as Apache Hive and Apache Pig, you can process data for analytics purposes and business intelligence workloads. Additionally, you can use Amazon EMR to transform and move large amounts of data into and out of other AWS data stores and databases, such as Amazon Simple Storage Service (Amazon S3) and Amazon DynamoDB.

What is Amazon Elastic Map Reduce (EMR)?

Amazon provides support for running MapReduce algorithm by Amazon Elastic MapReduce platform. Amazon EMR can be used to run the big data-based software like- Apache Hadoop, Apache Spark etc. in AWS cloud. Amazon EMR can be used for running on large datasets as well as for analyzing the big data. It supports business analytics and related functions. In addition, we can use Amazon EMR with Amazon S3 to transform very large data sets and databases. It also works very well with NoSQL DB like DynamoDB.

What are the main features of Amazon CloudSearch?

Amazon CloudSearch is mainly used for implementing a search solution of a website or application in AWS. It is highly scalable service and very easy to manage. It can index and search structured data as well as plain text.

Main feature of Amazon CloudSearch is: -

Prefix search, Full text search, Boolean search, Term boosting, Range search, Autocomplete Suggestions, Faceting, Highlighting

What is AWS Data Pipeline?

AWS Data Pipeline is a web service for automating the transformation of large scale data in AWS cloud.

We can use AWS Data Pipeline to define data-driven workflows. In such a workflow tasks follow a sequential pattern, where one task waits for completion of another task in the flow.

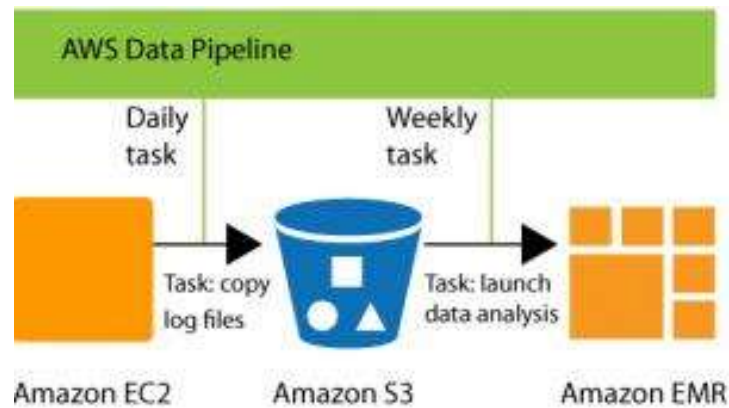
What is the difference between AWS Data Pipeline and Amazon Simple Workflow Service?

AWS Data pipeline is mainly used for data driven workflows that are popular in Big Data systems. AWS Data pipeline can easily copy data between different data stores and it can execute data transformations. To create such data flows, little programming knowledge is required. Amazon Simple Workflow Service (SWS) is mainly used for process automation. It can easily coordinate work across distributed application components.

We can do media processing, backend flows, analytics pipelines etc. with SWS. So, it is not limited to just Data driven flows.

What is AWS Data Pipeline? and what are the components of AWS Data Pipeline?

AWS Data Pipeline is a web service that you can use to automate the movement and transformation of data. With AWS Data Pipeline, you can define data-driven workflows, so that tasks can be dependent on the successful completion of previous tasks.



The following components of AWS Data Pipeline work together to manage your data:

- A pipeline definition specifies the business logic of your data management. For more information, see [Pipeline Definition File Syntax](#).
- A pipeline schedules and runs tasks. You upload your pipeline definition to the pipeline, and then activate the pipeline. You can edit the pipeline definition for a running pipeline and activate the pipeline again for it to take effect. You can deactivate the pipeline, modify a data source, and then activate the pipeline again. When you are finished with your pipeline, you can delete it.
- Task Runner polls for tasks and then performs those tasks. For example, Task Runner could copy log files to Amazon S3 and launch Amazon EMR clusters. Task Runner is installed and runs automatically on resources created by your pipeline definitions. You can write a custom task runner application, or you can use the Task Runner application that is provided by AWS Data Pipeline. For more information, see [Task Runners](#).

What is Amazon Kinesis Firehose?

Amazon Kinesis Firehose is a fully managed service for delivering real-time streaming data to destinations such as Amazon Simple Storage Service (Amazon S3) and Amazon Redshift.

What Is Amazon CloudSearch and its features?

Amazon CloudSearch is a fully managed service in the cloud that makes it easy to set up, manage, and scale a search solution for your website or application.

You can use Amazon CloudSearch to index and search both structured data and plain text. Amazon CloudSearch features:

- Full text search with language-specific text processing
- Boolean search
- Prefix searches
- Range searches
- Term boosting
- Faceting
- Highlighting
- Autocomplete Suggestions

What is an activity in AWS Data Pipeline?

An activity in AWS Data Pipeline is an Action that is initiated as a part of the pipeline. Some of the activities are: -

- Elastic MapReduce (EMR)
- Hive jobs
- Data copies
- SQL queries
- Command-line scripts

What is a schedule in AWS Data Pipeline?

In AWS Data Pipeline we can define a Schedule. The Schedule contains the information about when will pipeline activities run and with what frequency. All schedules have a start date and a frequency. E.g. One schedule can be run every day starting Mar 1, 2016, at 6am. Schedules may also have an end date, after which the AWS Data Pipeline service will not execute any activity.

What is the main framework behind Amazon Elastic MapReduce (EMR)?

Apache Hadoop is the main framework behind Amazon EMR. It is a distributed data processing engine. Hadoop is Open source Java based software framework. It supports data-intensive distributed applications running on large clusters of commodity hardware. Hadoop is based on MapReduce algorithm in which data is divided into multiple small fragments of work. Each of these tasks can be executed on any node in the cluster. In AWS EMR, Hadoop is run on the hardware provides by AWS cloud.

What are different states in AWS EMR cluster?

AWS EMR has following cluster states: **STARTING** – In this state, cluster provisions, starts, and configures EC2 instances **BOOTSTRAPPING** – In this state cluster is executing the Bootstrap process **RUNNING** – State in which cluster is currently being run **WAITING** – In this state cluster is currently active, but there are no steps to run **TERMINATING** - Shut down of cluster has started **TERMINATED** - The cluster is shut down without any error **TERMINATED_WITH_ERRORS** - The cluster is shut down with errors.

What are the use cases for Amazon Kinesis Streams?

Amazon Kinesis Streams helps in creating applications that deal with streaming data. Kinesis streams can work with data streams up to terabytes per hour rate. Kinesis streams can handle data from thousands of sources. We can also use Kinesis to produce data for use by other Amazon services. Some of the main use cases for Amazon Kinesis Streams are as follows:

Real-time Analytics: At times for real-time events like-Big Friday sale or a major game event, we get a large amount of data in a short period of time. Amazon Kinesis Streams can be used to perform real time analysis on this data and make use of this analysis very quickly. Prior to Kinesis, this kind of analysis would take days. Whereas now within a few minutes we can start using the results of this analysis.

Gaming Data: In online applications, thousands of users play and generate a large amount of data. With Kinesis, we can use the streams of data generated by an online game and use it to implement dynamic features based on the actions and behavior of players.

Log and Event Data: We can use Amazon Kinesis to process the large amount of Log data that is generated by different devices. We can build live dashboards, alarms, triggers based on this streaming data by using Amazon Kinesis.

Mobile Applications: In Mobile applications, there is wide variety of data available due to the large number of parameters like- location of mobile, type of device, time of the day etc. We can use Amazon Kinesis Streams to process the data generated by a Mobile App. The output of such processing can be used by the same Mobile App to enhance user experience in real time.



Business Productivity

Alexa for Business Empower your organization with Alexa	Amazon Chime Frustration-free Meetings, Video Calls, and Chat	Amazon WorkDocs Enterprise Storage and Sharing Service
Amazon WorkMail Secure and Managed Business Email and Calendaring		



Business Productivity

What is WorkDocs?

Amazon WorkDocs is a fully managed, secure enterprise storage and sharing service with strong administrative controls and feedback capabilities that improve user productivity.

What is WorkMail?

Amazon WorkMail is a managed email and calendaring service that offers strong security controls and support for existing desktop and mobile clients.

Which AWS responsible for managed email and calendaring?

WorkMail is a managed email and calendaring service with strong security controls and support for existing desktop and mobile email clients. You can access their email, contacts, and calendars wherever you use Microsoft Outlook, your browser, or your iOS and Android mobile devices. You can integrate Amazon WorkMail with your existing corporate directory and control both the keys that encrypt your data and the location where your data is stored.



Desktop & App Streaming

Amazon WorkSpaces	Amazon AppStream 2.0		
Desktop Computing Service	Stream Desktop Applications Securely to a Browser		



Desktop & App Streaming

What is WorkSpaces?

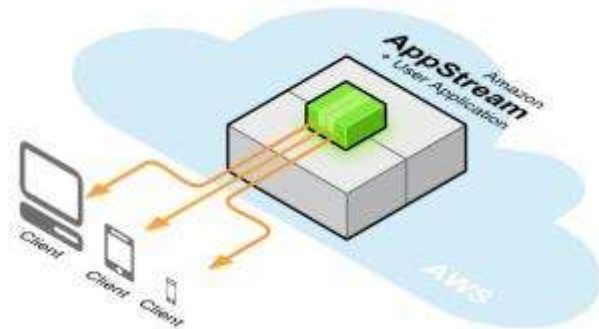
Amazon WorkSpaces is a fully managed desktop computing service in the cloud.

What is AppStream?

Amazon AppStream lets you stream resource intensive applications and games from the cloud to multiple end-user devices.

What is Amazon AppStream and advantage of using AppStreaming?

Amazon AppStream is an application streaming service that lets you stream your existing resource-intensive applications from the cloud without code modifications.



Advantages of Streaming Your Application:

Interactively streaming your application from the cloud provides several benefits:

- **Remove Device Constraints** – You can leverage the compute power of AWS to deliver experiences that wouldn't normally be possible due to the GPU, CPU, memory or physical storage constraints of local devices.
- **Support Multiple Platforms** – You can write your application once and stream it to multiple device platforms. To support a new device, just write a small client to connect to your streaming application.
- **Fast and Easy Updates** – Because your streaming application is centrally managed by Amazon AppStream, updating your application is as simple as providing a new version of your streaming application to Amazon AppStream. You can immediately upgrade all of your customers without any action on their part.
- **Instant On** – Streaming your application with Amazon AppStream lets your customers start using your application or game immediately, without the delays associated with large file downloads and time-consuming installations.
- **Improve Security** – Unlike traditional boxed software and digital downloads, where your application is available for theft or reverse engineering, Amazon AppStream stores your streaming application binary securely in AWS datacenters.
- **Automatic Scaling** – You can use Amazon AppStream to specify capacity needs, and then the service automatically scales your streamed application and connects customers' devices to it.

What are the advantages of using AppStream in AWS?

We can use Amazon AppStream to stream our resource-intensive applications from AWS cloud.

The Main advantages of AppStream are: -

Device Constraints: Since WS provides computing environment, there are no device constraints like CPU, memory etc. on the application.

Fast and Easy Upgrade: since AppStream manages the application, it is very easy to upgrade and send updates to application. New version of the application can be immediately released.

Multiple Platforms: Since AppStream supports multiple platforms, we can write the application code one time and stream it to multiple device platforms. We just have to write a small client to support a new device.

Instant On: Amazon AppStream can be used for starting the application immediately. There is no delay related to large file download or installation in AppStream. **Security:** Application in AppStream is safe from theft or any other form of plagiarism. Applications are stored very securely in AppStream.

Auto-Scaling: Amazon AppStream supports Autoscaling based on the need as well as spikes in traffic requests.



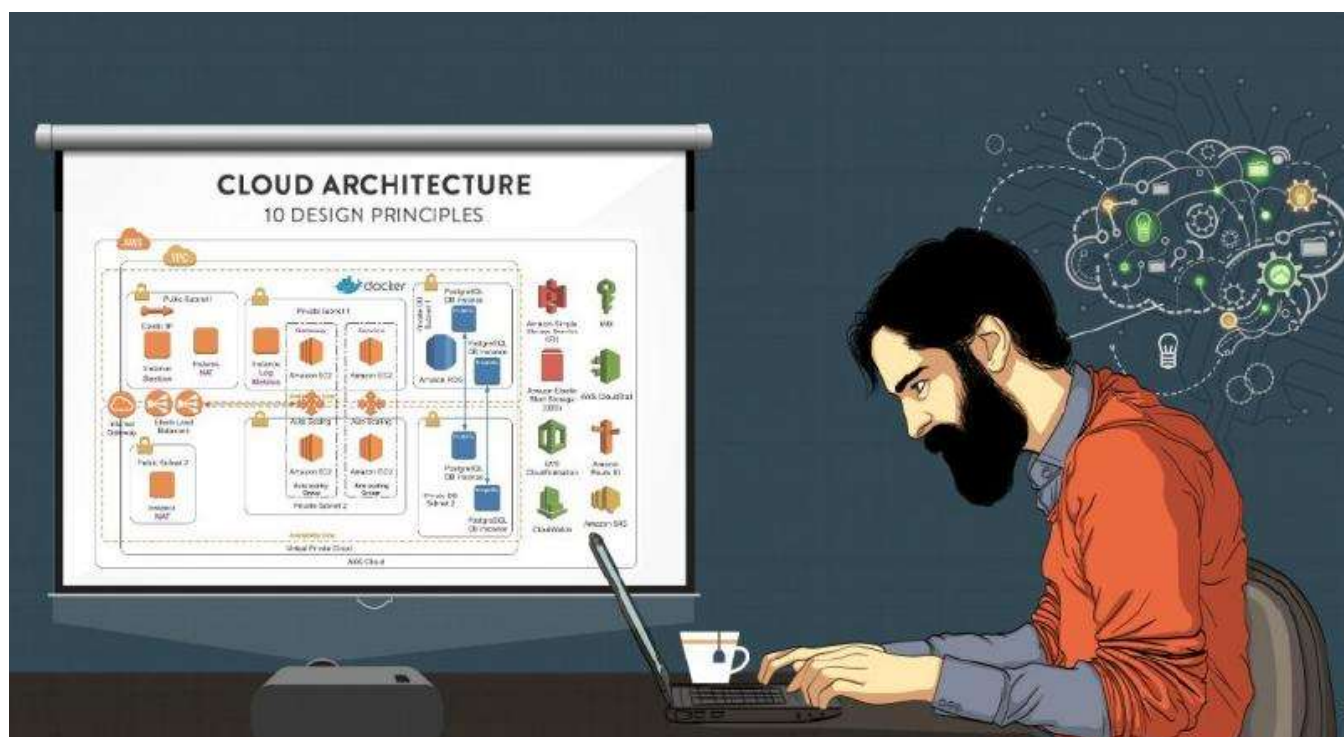
AWS Architecture

10 Design Principles of Cloud Architecture	Architecture Scenarios	General Q&A	

10 Design Principles for Cloud Architecture

Cloud computing is one of the boons of technology, making storage and access of documents easier and efficient. For it to be reliable, the cloud architecture need to be impeccable. It needs to be secure, high performing and cost efficient. A good cloud architecture design should take advantage of some of the inherent strengths of cloud computing – elasticity, ability to automate infrastructure management etc. Cloud architecture design needs to be well thought out because it forms the backbone of a vast network. It cannot be arbitrarily designed.

There are certain principles that one needs to follow to make the most of the tremendous capabilities of the Cloud. Here are ten design principles that you must consider while architecting for AWS cloud.



1. Automate Everything

Unlike traditional IT infrastructure, Cloud enables automation of a number of events, improving both your system's stability and the efficiency of your organization. Some of the AWS resources you can use to get automated are: -

AWS Elastic Beanstalk: This resource is the fastest and simplest way to get an application up and running on AWS. You can simply upload their application code and the service automatically handles all the details, such as resource provisioning, load balancing, auto scaling, and monitoring.

Amazon EC2 Auto recovery: You can create an Amazon CloudWatch alarm that monitors an Amazon EC2 instance and automatically recovers it if it becomes impaired. A word of caution though – During instance recovery, the instance is migrated through an instance reboot, and any data that is in-memory is lost.

Auto Scaling: With Auto Scaling, you can maintain application availability and scale your Amazon EC2 capacity up or down automatically according to conditions you define.

Amazon CloudWatch Alarms: You can create a CloudWatch alarm that sends an Amazon Simple Notification Service (Amazon SNS) message when a particular metric goes beyond a specified threshold for a specified number of periods.

Amazon CloudWatch Events: The CloudWatch service delivers a near real-time stream of system events that describe changes in AWS resources. Using simple rules that you can set up in a couple of minutes, you can easily route each type of event to one or more targets: AWS Lambda functions, Amazon Kinesis streams, Amazon SNS topics, etc.

AWS OpsWorks Lifecycle events: AWS OpsWorks supports continuous configuration through lifecycle events that automatically update your instances configuration to adapt to environment changes. These events can be used to trigger Chef recipes on each instance to perform specific configuration tasks.

AWS Lambda Scheduled events: These events allow you to create a Lambda function and direct AWS Lambda to execute it on a regular schedule.

As an architect for the AWS Cloud, these automation resources are a great advantage to work with.

2. Implement loose coupling

IT systems should ideally be designed in a way that reduces interdependencies. Your components need to be loosely coupled to avoid changes or failure in one of the components from affecting others.

Your infrastructure also needs to have well defined interfaces that allow the various components to interact with each other only through specific, technology-agnostic interfaces. Modifying any underlying operations without affecting other components should be made possible.

In addition, by implementing service discovery, smaller services can be consumed without prior knowledge of their network topology details through loose coupling. This way, new resources can be launched or terminated at any point of time.

Loose coupling between services can also be done through asynchronous integration. It involves one component that generates events and another that consumes them. The two components do not integrate through direct point-to-point interaction, but usually through an intermediate durable storage layer. This approach decouples the two components and introduces additional resiliency. So, for example, if a process that is reading messages from the queue fails, messages can still be added to the queue to be processed when the system recovers.

Lastly, building applications in such a way that they handle component failure in a graceful manner helps you reduce impact on the end users and increase your ability to make progress on your offline procedures.

3. Focus on services, not servers

A wide variety of underlying technology components are required to develop, manage and operate applications. Your architecture should leverage a broad set of compute, storage, database, analytics, application, and deployment services. On AWS, there are two ways to do that. The first is through managed services that include databases, machine learning, analytics, queuing, search, email, notifications, and more. For example, with the Amazon Simple Queue Service (Amazon SQS) you can offload the administrative burden of operating and scaling a highly available messaging cluster, while paying a low price for only what you use. Not only that, Amazon SQS is inherently scalable.

The second way is to reduce the operational complexity of running applications through server-less architectures. It is possible to build both event-driven and synchronous services for mobile, web, analytics, and the Internet of Things (IoT) without managing any server infrastructure.

4. Database is the base of it all

On AWS, managed database services help remove constraints that come with licensing costs and the ability to support diverse database engines that were a problem with the traditional IT infrastructure? You need to keep in mind that access to the information stored on these databases is the main purpose of cloud computing.

There are three different categories of databases to keep in mind while architecting: -

Relational databases: Data here is normalized into tables and also provided with powerful query language, flexible indexing capabilities, strong integrity controls, and the ability to combine data from multiple tables in a fast and efficient manner. They can be scaled vertically and are highly available during failovers (designed for graceful failures).

NoSQL databases: These databases trade some of the query and transaction capabilities of relational databases for a more flexible data model that seamlessly scales horizontally. NoSQL databases utilize a variety of data models, including graphs, key-value pairs, and JSON documents. NoSQL databases are widely recognized for ease of development, scalable performance, high availability, and resilience.

Introduce redundancy to remove single points of failure, by having multiple resources for the same task. Redundancy can be implemented in either standby mode (functionality is recovered through failover while the resource remains unavailable) or active mode (requests are distributed to multiple redundant compute resources, and when one of them fails, the rest can simply absorb a larger share of the workload).

Data warehouse: A specialized type of relational database, optimized for analysis and reporting of large amounts of data. It can be used to combine transactional data from disparate sources making them available for analysis and decision-making.

5. Be sure to remove single points of failure

A system is highly available when it can withstand the failure of an individual or multiple component (e.g., hard disks, servers, network links etc.). You can think about ways to automate recovery and reduce disruption at every layer of your architecture. This can be done with the following processes:

It is crucial to have a durable data storage that protects both data availability and integrity. Redundant copies of data can be introduced either through synchronous, asynchronous or Quorum based replication. New item

Detection and reaction to failure should both be automated as much as possible.

Automated Multi –Data Center resilience is practiced through Availability Zones across data centers that reduce the impact of failures. Fault isolation improvement can be made to traditional horizontal scaling by Sharding (a method of grouping instances into groups called shards, instead of sending the traffic from all users to every node like in the traditional IT structure.)

Introduce redundancy to remove single points of failure, by having multiple resources for the same task. Redundancy can be implemented in either standby mode (functionality is recovered through failover while the resource remains unavailable) or active mode (requests are distributed to multiple redundant compute resources, and when one of them fails, the rest can simply absorb a larger share of the workload).

6. Optimize for cost

At the end of the day, it often boils down to cost. Your cloud architecture should be designed for cost optimization by keeping in mind the following principles:

You can reduce cost by selecting the right types, configurations and storage solutions to suit your needs. Implementing Auto Scaling so that you can scale horizontally when required or scale down when necessary can be done without any extra cost. List item #1

You can reduce cost by selecting the right types, configurations and storage solutions to suit your needs. Implementing Auto Scaling so that you can scale horizontally when required or scale down when necessary can be done without any extra cost.

7. Caching

Applying data caching to multiple layers of an IT architecture can improve application performance and cost efficiency of application.

There are two types of caching: -

Application data caching: Information can be stored and retrieved from fast, managed, in-memory caches in the application, which decreases load for the database and increases latency for end users.

Edge caching: Content is served by infrastructure that is closer to the viewers lowering latency and giving you the high, sustained data transfer rates needed to deliver large popular objects to end users at scale.

Amazon CloudFront, the content delivery network consisting of multiple edge locations around the world is the edge caching service whereas Amazon ElastiCache makes it easy to deploy, operate and scale in-memory cache in the cloud.

8. Security

Security is everything! Most of the security tools and techniques used in the traditional IT infrastructure can be used in the cloud as well. AWS is a platform that allows you to formalize the design of security controls in the platform itself. It simplifies system use for administrators and those running IT and makes your environment much easier to audit in a continuous manner.

Some ways to improve security in AWS are:

Utilize AWS features for Defense in depth – Starting at the network level, you can build a VPC topology that isolates parts of the infrastructure through the use of subnets, security groups, and routing controls.

AWS operates under a shared security responsibility model, where AWS is responsible for the security of the underlying cloud infrastructure and you are responsible for securing the workloads you deploy in AWS.

Reduce privileged access to the programmable resources and servers to avoid breach of security. The overuse of guest operating systems and service accounts can breach security.

Create an AWS CloudFormation script that captures your security policy and reliably deploys it, allowing you to perform security testing as part of your release cycle, and automatically discover application gaps and drift from your security policy.

Testing and auditing your environment is key to moving fast while staying safe. On AWS, it is possible to implement continuous monitoring and automation of controls to minimize exposure to security risks.

Services like AWS Config, Amazon Inspector, and AWS Trusted Advisor continually monitor for compliance or vulnerabilities giving you a clear overview of which IT resources are in compliance, and which are not.

9. Think Adaptive and Elastic

The architecture of the cloud should be such that it supports growth of users, traffic, or data size with no drop-in performance. It should also allow for linear scalability when and where an additional resource is added. The system needs to be able to adapt and proportionally serve additional load.

Whether the architecture includes vertical scaling, horizontal scaling or both; it is up to the designer, depending on the type of application or data to be stored. But your design should be equipped to take maximum advantage of the virtually unlimited on-demand capacity of cloud computing.

Consider whether your architecture is being built for a short-term purpose, wherein you can implement vertical scaling. Else, you will need to distribute your workload to multiple resources to build internet-scale applications by scaling horizontally. Either way, your architecture should be flexible enough to adapt to the demands of cloud computing.

Also, knowing when to engage stateless applications, stateful applications, stateless components and distributed processing, makes your cloud very effective in its storage.

10. Treat servers as disposable resources

One of the biggest advantages of cloud computing is that you can treat your servers as disposable resources instead of fixed components. However, resources should always be consistent and tested. One way to enable this is to implement the immutable infrastructure pattern, which enables you to replace the server with one that has the latest configuration instead of updating the old server.

It is important to keep the configuration and coding as an automated and repeatable process, either when deploying resources to new environments or increasing the capacity of the existing system to cope with extra load. Bootstrapping, Golden Images or a Hybrid of the two will help you keep the process automated and repeatable without any human errors.

Bootstrapping can be executed after launching an AWS resource with default configuration. This will let you reuse the same scripts without modifications.

But in comparison, the Golden Image approach results in faster start times and removes dependencies to configuration services or third-party repositories. Certain AWS resource types like Amazon EC2 instances, Amazon RDS DB instances, Amazon Elastic Block Store (Amazon EBS) volumes, etc., can be launched from a golden image.

When suitable, use a combination of the two approaches, where some parts of the configuration get captured in a golden image, while others are configured dynamically through a bootstrapping action. Not to be limited to the individual resource level, you can apply techniques, practices, and tools from software development to make your whole infrastructure reusable, maintainable, extensible, and testable.

Architecture Scenarios

Architecture Scenario 1: Web Application Hosting

Highly available and scalable web hosting can be complex and, expensive, Dense peak periods and wild swings in traffic patterns result in low utilization of expensive hardware. Amazon Web Services provides the reliable scalable, secure and high performance, infrastructure required for web applications while enabling an elastic scale out and scale down infrastructure to match IT costs in real time as customer traffic fluctuates.

Architecture Scenario 2: Disaster Recovery for Local Applications

Disaster recovery is about preparing for and recovering from any event that has a negative impact on your IT systems. A typical approach involves duplicating infrastructure to ensure the availability of spare capacity in the event of a disaster.

Amazon Web Services allows you to scale up your infrastructure on an as-needed basis. For a disaster recovery solution, this results in significant cost savings. The following diagram shows an example of a disaster recovery setup for a local application.

Architecture Scenario 3: File Synchronization Service

Given the straightforward, stateless client-server architecture in which web services are viewed as resources and can be identified by their URLs, development teams are free to create file sharing and syncing applications for their departments, for enterprises, or for consumers directly.

This diagram represents the core architecture of a scalable and cost-effective file sharing and synchronization platform, using Amazon Web Services.

Architecture Scenario 4: Online Games

Online games back-end infrastructures can be challenging to maintain and operate. Peak usage periods, multiple players, and high volumes of write operations are some of the most common problems that operations teams face.

But the most difficult challenge is ensuring flexibility in the scale of that system. A popular game might suddenly receive millions of users in a matter of hours, yet it must continue to provide a satisfactory player experience. Amazon Web Services provides different tools and services that can be used for building online games that scale under high usage traffic patterns.

This document presents a cost-effective online game architecture featuring automatic capacity adjustment, a highly available and high-speed database, and a data processing cluster for player behavior analysis.

Architecture Scenario 5: Financial Services Grid Computing

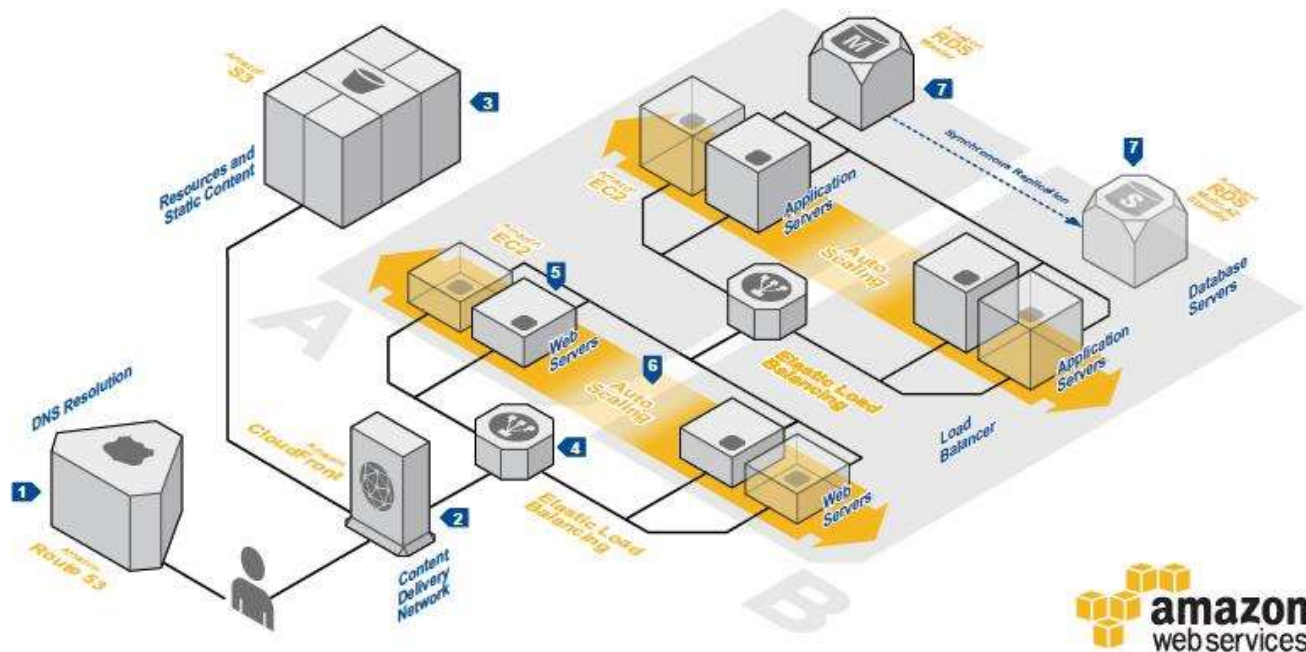
Financial services grid computing on the cloud provides dynamic scalability and elasticity for operation when compute jobs are required and utilizing services for aggregation that simplify the development of grid software.

On demand provisioning of hardware, and template driven deployment, combined with low latency access to existing on-premise data sources make AWS a powerful platform for high performance grid computing systems.

Architecture: Web Application Hosting

WEB APPLICATION HOSTING

Highly available and scalable web hosting can be complex and expensive. Dense peak periods and wild swings in traffic patterns result in low utilization of expensive hardware. Amazon Web Services provides the reliable, scalable, secure, and high-performance infrastructure required for web applications while enabling an elastic, scale-out and scale-down infrastructure to match IT costs in real time as customer traffic fluctuates.



System Overview

1 The user's DNS requests are served by **Amazon Route 53**, a highly available Domain Name System (DNS) service. Network traffic is routed to infrastructure running in Amazon Web Services.

2 Static, streaming, and dynamic content is delivered by **Amazon CloudFront**, a global network of edge locations. Requests are automatically routed to the nearest edge location, so content is delivered with the best possible performance.

3 Resources and static content used by the web application are stored on **Amazon Simple Storage Service (S3)**, a highly durable storage infrastructure designed for mission-critical and primary data storage.

4 HTTP requests are first handled by **Elastic Load Balancing**, which automatically distributes incoming application traffic among multiple **Amazon Elastic Compute Cloud (EC2)** instances across Availability Zones (AZs). It enables even greater fault tolerance in your applications, seamlessly providing the amount of load balancing capacity needed in response to incoming application traffic.

5 Web servers and application servers are deployed on **Amazon EC2** instances. Most organizations will select an **Amazon Machine Image (AMI)** and then customize it to their needs. This custom AMI will then become the starting point for future web development.

6 Web servers and application servers are deployed in an **Auto Scaling** group. **Auto Scaling** automatically adjusts your capacity up or down 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 to minimize costs.

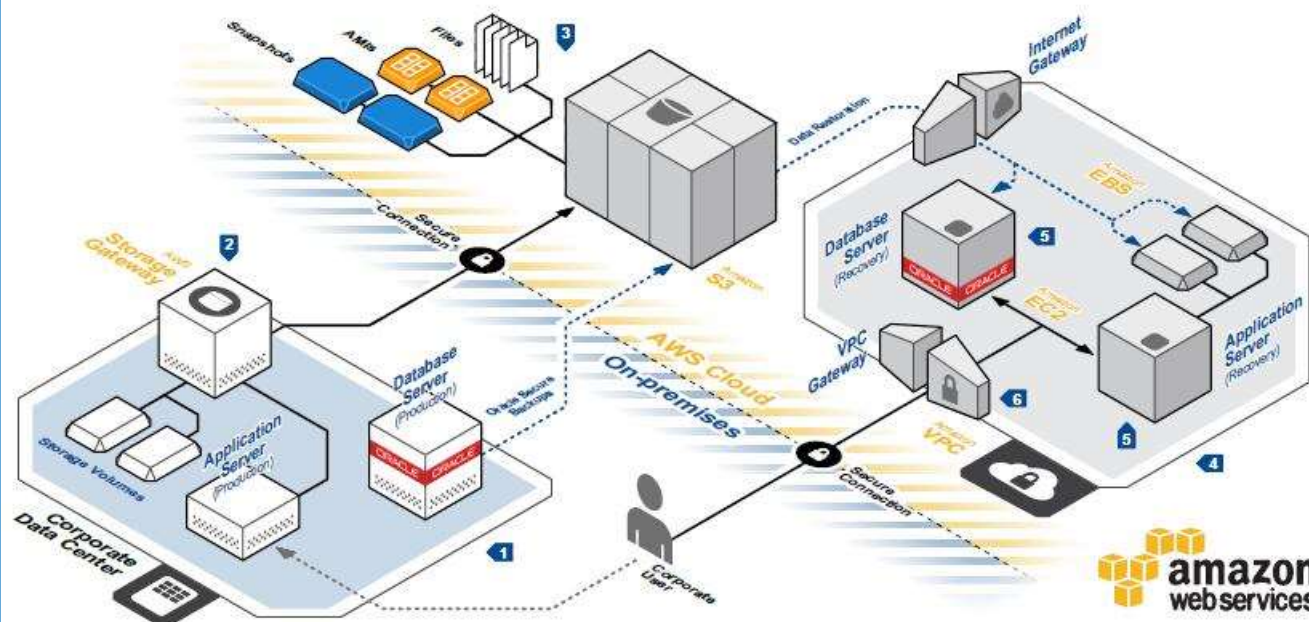
7 To provide high availability, the relational database that contains application's data is hosted redundantly on a multi-AZ (multiple Availability Zones—zones A and B here) deployment of **Amazon Relational Database Service (Amazon RDS)**.

Architecture: Disaster Recovery for Local Applications

DISASTER RECOVERY FOR LOCAL APPLICATIONS

Disaster recovery is about preparing for and recovering from any event that has a negative impact on your IT systems. A typical approach involves duplicating infrastructure to ensure the availability of spare capacity in the event of a disaster.

Amazon Web Services allows you to scale up your infrastructure on an as-needed basis. For a disaster recovery solution, this results in significant cost savings. The following diagram shows an example of a disaster recovery setup for a local application.



System Overview

- 1** A corporate data center hosts an application consisting of a database server and an application server with local storage for a content management system.
- 2** **AWS Storage Gateway** is a service connecting an on-premises software appliance with cloud-based storage. **AWS Storage Gateway** securely uploads data to the AWS cloud for cost effective backup and rapid disaster recovery.
- 3** Database server backups, application server volume snapshots, and **Amazon Machine Images (AMI)** of the

recovery servers are stored on **Amazon Simple Storage Service (Amazon S3)**, a highly durable and cost-effective data store. AMIs are pre-configured operating system and application software that are used to create a virtual machine **Amazon Elastic Compute Cloud (Amazon EC2)**. Oracle databases can directly back up to Amazon S3 using the **Oracle Secure Backup (OSB) Cloud Module**.

- 4** In case of disaster in the corporate data center, you can recreate the complete infrastructure from the backups

on **Amazon Virtual Private Cloud (Amazon VPC)**. **Amazon VPC** lets you provision a private, isolated section of the AWS cloud where you can recreate your application.

- 5** The application and database servers are recreated using **Amazon EC2**. To restore volume snapshots, you can use **Amazon Elastic Block Store (EBS)** volumes, which are then attached to the recovered application server.

- 6** To remotely access the recovered application, you use a VPN connection created by using the VPC Gateway.

FILE SYNCHRONIZATION SERVICE

Given the straightforward, stateless client-server architecture in which web services are viewed as resources and can be identified by their URLs, development teams are free to create file sharing and syncing applications for their departments, for enterprises, or for consumers directly.

This diagram represents the core architecture of a scalable and cost-effective file sharing and synchronization platform, using Amazon Web Services.

The diagram illustrates the core architecture of a File Synchronization Service using Amazon Web Services (AWS). The components and their interactions are as follows:

- Client Devices:** Represented by icons of a smartphone and a laptop, connected to the system via **DNS** and **Route 53**.
- Application Servers:** A fleet of servers managed by **Elastic Load Balancing** (1) and **Auto Scaling**, which handle client requests.
- Files Repository:** A storage component (4) that stores files, connected to the Application Servers via **Amazon S3**.
- Security Token Service:** (3) Provides temporary security credentials to the Application Servers.
- File Metadata Store:** A **Amazon DynamoDB** database (5) that stores file metadata, connected to the Application Servers.
- Email Sender:** A service (6) that sends notifications, connected to the Application Servers via **Amazon SES**.
- File Followers:** Represented by a group of people icon, who receive updates from the system.

The architecture is designed for scalability and cost-effectiveness, leveraging the capabilities of AWS services.

- 1 The file synchronization service endpoint consists of an **Elastic Load Balancer** distributing incoming requests to a group of application servers hosted on **Amazon Elastic Compute Cloud** (Amazon EC2) instances. An **Auto Scaling** group automatically adjusts the number of **Amazon EC2** instances depending on the application needs.
- 2 To upload a file, a client first needs to request the permission to the service and get a security token.
- 3 After checking the user's identity, application servers get a temporary credential from **AWS Security Token Service** (STS). This credential allows users to upload files.

5 File metadata, version information, and unique identifiers are stored by the application servers on an **Amazon DynamoDB** table. As the number of files to maintain in the application grows, **Amazon DynamoDB** tables can store and retrieve any amount of data, and serve

6 File change notifications can be sent via email to users following the resource with **Amazon Simple Email Service** (Amazon SES), an easy-to-use, cost-effective email solution.

7 Other clients sharing the same files will query the service endpoint to check if newer versions are available. This query compares the list of local files checksums with the checksums listed in an **Amazon DynamoDB** table. If the query finds newer files, they can be retrieved from **Amazon S3** and sent to the client application.

ONLINE GAMES

Online games back-end infrastructures can be challenging to maintain and operate. Peak usage periods, multiple players, and high volumes of write operations are some of the most common problems that operations teams face.

But the most difficult challenge is ensuring flexibility in the scale of that system. A popular game might suddenly receive millions of users in a matter of hours, yet it must continue to provide a

satisfactory player experience. Amazon Web Services provides different tools and services that can be used for building online games that scale under high usage traffic patterns.

This document presents a cost-effective online game architecture featuring automatic capacity adjustment, a highly available and high-speed database, and a data processing cluster for player behavior analysis.

The diagram illustrates a multi-tier architecture for an online game. It starts with a DNS service (Route 53) pointing to a website (www.mygame.com) and a game interaction service (Game Interaction, Game SDK, etc.). The game interaction service connects to a Game Client (Players). The Game Client interacts with a Game File (Game files) and a Game Log (Game log files). The Game File is stored in a Content Delivery Network (CDN) and a Game Database. The Game Log is stored in a Files Repository. The Files Repository feeds into a Game Analysis cluster, which then feeds into an Email Emitter. The Game Database is connected to a Game File (Game files) and a Game Log (Game log files). The Game File is stored in a Game Database. The Game Log is stored in a Files Repository. The Files Repository feeds into a Game Analysis cluster, which then feeds into an Email Emitter. The Game Database is connected to a Game File (Game files) and a Game Log (Game log files). The Game File is stored in a Game Database. The Game Log is stored in a Files Repository. The Files Repository feeds into a Game Analysis cluster, which then feeds into an Email Emitter. The Game Database is connected to a Game File (Game files) and a Game Log (Game log files). The Game File is stored in a Game Database. The Game Log is stored in a Files Repository. The Files Repository feeds into a Game Analysis cluster, which then feeds into an Email Emitter.

amazon web services

1 Browser games can be represented as client-server applications. The client generally consists of static files, such as images, sounds, flash applications, or Java applets. Those files are hosted on **Amazon Simple Storage Service** (Amazon S3), a highly available and reliable data store.

2 As the user base grows and becomes more geographically distributed, a high-performance cache like **Amazon CloudFront** can provide substantial improvements in latency, fault tolerance, and cost. By using **Amazon S3** as the origin server for the **Amazon CloudFront** distribution, the game infrastructure benefits from fast network data transfer rates and a simple publishing/caching workflow.

3 Requests from the game application are distributed by **Elastic Load Balancing** to a group of web servers running on **Amazon Elastic Compute Cloud** (Amazon EC2) instances. **Auto Scaling** automatically adjusts the size of this group, depending on rules like network load, CPU usage, and so on.

4 Player data is persisted on **Amazon DynamoDB**, a fully managed NoSQL database service. As the player population grows, **Amazon DynamoDB** provides predictable performance with seamless scalability.

5 Log files generated by each web server are pushed back into **Amazon S3** for long-term storage.

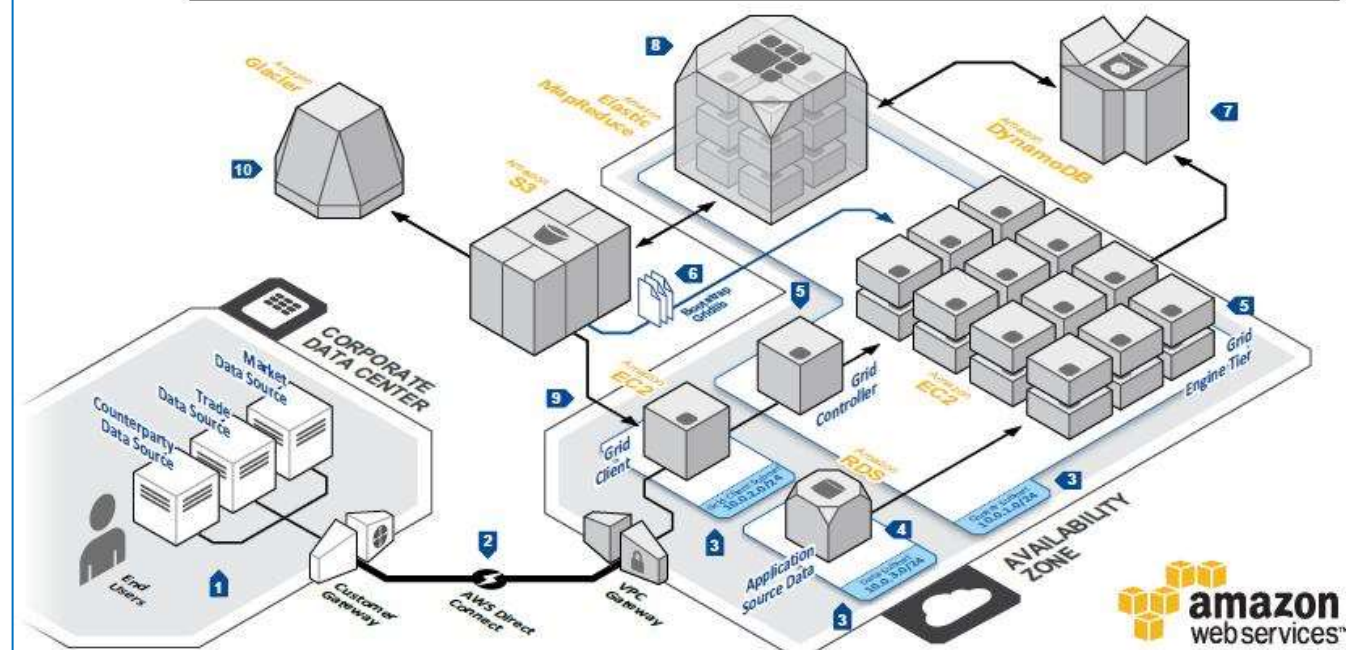
6 Managing and analyzing high data volumes produced by online games platforms can be challenging. **Amazon Elastic MapReduce** (Amazon EMR) is a service that processes vast amounts of data easily. Input data can be retrieved from web server logs stored on **Amazon S3** or from player data stored in **Amazon DynamoDB** tables to run analytics on player behavior, usage patterns, etc. Those results can be stored again on **Amazon S3**, or inserted in a relational database for further analysis with classic business intelligence tools.

7 Based on the needs of the game, **Amazon Simple Email Service** (Amazon SES) can be used to send email to players in a cost-effective and scalable way.

Architecture: Financial Services Grid Computing

FINANCIAL SERVICES GRID COMPUTING

Financial services grid computing on the cloud provides dynamic scalability and elasticity for operation when compute jobs are required, and utilizing services for aggregation that simplify the development of grid software. On demand provisioning of hardware, and template driven deployment, combined with low latency access to existing on-premise data sources make AWS a powerful platform for high performance grid computing systems.



System Overview

- 1** Date sources for market, trade, and counterparties are installed on startup from on premise data sources, or from **Amazon Simple Storage Service (Amazon S3)**.
- 2** **AWS DirectConnect** can be used to establish a low latency and reliable connection between the corporate data center site and AWS, in 1 to 10Gbit increments. For situations with lower bandwidth requirements, a VPN connection to the **VPC Gateway** can be established.
- 3** Private subnetworks are specifically created for customer source data, compute grid clients, and the grid controller and engines.

4 Application and corporate data can be securely stored in the cloud using the **Amazon Relational Database Service (Amazon RDS)**.

5 Grid controllers and grid engines are running **Amazon Elastic Compute Cloud (Amazon EC2)** instances started on demand from **Amazon Machine Images (AMIs)** that contain the operating system and grid software.

6 Static data such as holiday calendars and QA libraries and additional gridlib bootstrapping data can be downloaded on startup by grid engines from **Amazon S3**.

7 Grid engine results can be stored in **Amazon DynamoDB**, a fully managed database providing configurable read and write throughput, allowing scalability on demand.

8 Results in **Amazon DynamoDB** are aggregated using a map/reduce job in **Amazon Elastic MapReduce (Amazon EMR)** and final output is stored in **Amazon S3**.

9 The compute grid client collects aggregate results from **Amazon S3**.

10 Aggregate results can be archived using **Amazon Glacier**, a low-cost, secure, and durable storage service.

What key components of Amazon Web Service (AWS) do you use in your project?

We use following key components of AWS in our project:

Amazon Simple Storage Service or (S3): We use AWS S3 to store our data in cloud. Mostly the data is encrypted before storing it. Also, data is replicated in multiple availability zones.

Amazon Elastic Compute Cloud (EC2): We use Amazon EC2 for running our programs in cloud. It gives us scalable computing resources on-demand for hosting applications. We can use Autoscaling to handle high traffic demands.

Elastic Load Balancing (ELB): ELB is used for distributing the traffic in multiple nodes. This is also an important part of scalability solution. Amazon

CloudWatch: We use Amazon CloudWatch to monitor resources in AWS cloud. It helps in not only viewing but also in setting alerts based on key metrics of the AWS components.

Route 53: For DNS management we use Route 53 service of AWS.

Identity and Access Management (IAM): We use IAM for implementing security, identity management and authentication in AWS cloud.

How can your failover gracefully in AWS?

We can use **Elastic IPs** to implement failover in AWS. Elastic IP is a static IP and it is dynamically remappable. In case there is a failure at one node, we can quickly remap and failover to another set of servers. It will lead to routing of traffic to the new servers. It is also useful when we upgrade from old to new versions or when some piece of hardware fails.

What is the use of Availability Zones in AWS?

In AWS, Availability Zone is similar to a logical datacenter. We can deploy our application in multiple availability zones to ensure high availability of the application. Amazon provides RDS Multi-Availability Zone deployment functionality to automatically replicate database updates across multiple Availability Zones. This makes it easier to create and maintain highly available enterprise software systems.

Why AWS systems are built on “Design to Fail” approach?

At the core of an AWS system is, “Design for Fail” principle. It means if we design the software for failure nothing will fail. If we follow a pessimist approach while designing architecture in the cloud, we will assume that things will fail. To handle such failure, we will always create a system that can have automated recovery from failure. An AWS system is designed to automatically recover from design, execution and deploy stage failures.

What are the best practices to build a resilient system in AWS?

We can follow these best practices to build a resilient system in AWS:

Backup: We need a useful and fast, backup and restore strategy for our data. The backup and restore process should be automated. **Reboot:** Since nodes crash and new nodes restart in AWS, it is good to build threads that automatically resume on reboot of the node.

Re-sync: The system in AWS cloud should be able to re-sync itself by reloading messages from queues.

Images: We need to maintain pre-configured and pre-optimized virtual images to restore the system. Also, these images should be pre-configured to restart processes on reboot automatically.

In-memory sessions: Wherever possible we should minimize the use of in-memory sessions and stateful user context in AWS.

What are the tools in AWS that can be used for creating a system based on “Design to Fail” principle?

AWS provides many tools for creating a strong system based on “Design to Fail” principle. Some of these are:

Elastic IPs: We can failover gracefully by using Elastic IPs in AWS. An Elastic IP is a static IP that is dynamically re-mappable. We can quickly remap and failover to another set of servers so that application traffic is routed to the new set of servers. It is also very useful when we want to upgrade from old to new version of software.

Availability Zones: We can use multiple Availability Zones to introduce resiliency in AWS system. An Availability Zone is like a logical datacenter. By deploying application in multiple availability zones, we can ensure highly availability. **Amazon RDS:** In AWS, Amazon RDS provides deployment functionality to automatically replicate database updates across multiple Availability Zones. **Machine Image:** We can maintain an Amazon Machine Image to restore and clone environments easily in a different Availability Zone. We can use multiple Database slaves across Availability Zones and setup hot replication with these Machine images.

Amazon CloudWatch: This is a real-time open source monitoring tool in AWS that provides visibility

visibility on AWS cloud. We can take appropriate actions in case of hardware failure or performance degradation by setting alerts on CloudWatch. Auto scaling: We can maintain an auto-scaling group to maintain a fixed number of servers. In case of failure or performance degradation unhealthy Amazon EC2 instances are replaced by new ones. Amazon EBS: We can set up cron jobs to take incremental snapshots of Database and upload it automatically to Amazon S3. In this way, data is persisted independent of the instances.

Amazon RDS: We can set the retention period for backups by using Amazon RDS. It can also perform automated backups.

How can we build a Scalable system in AWS?

To build a scalable system, we have to follow the principle of Service Oriented Architecture (SOA). The modern word for this is Microservices architecture. Behind a scalable system there are loosely coupled components. Once we build components that are loosely coupled i.e. there is less dependency between them. If one component fails or performs slow, still the other components keep working as if there is no failure. In such a system, it is very easy to build horizontal scaling. We can add multiple servers for components that are heavily used based on the load. We can also add asynchronous communication between components to make the system scalable. This reduces the probability of single point of failure. With loosely coupled components, it is easier to use scalability options present in AWS cloud.

What are the different ways to implement Elasticity in AWS?

Elasticity can be built in AWS in following ways: Periodic Cyclic Scaling: In this case we scale the system at a fixed interval of time like-daily, monthly, quarterly. This is Period based scaling. Proactive Event-based Scaling: When we are expecting a big spike in traffic due to seasonal nature or a special business event (new product launch, holiday weekend), we go for proactive event-based scaling. It is done on one-time basis for a limited time. Auto-scaling: Based on increase in demand that is not known in advance, we can setup a monitoring service. Once demand reaches a threshold, we can scale up the system automatically. It can be based on metrics like- CPU load, memory usage, number of client requests.

What are the benefits of bootstrapping instances in AWS?

Following are the main benefits of bootstrapping instances in AWS: We can recreate the different environments for Dev, QA, and Production etc. with minimal effort by using bootstrapping. Bootstrapping instances gives more control over cloud-based resources in AWS. It also minimizes the occurrence of human related deployment errors.

One main benefit for Bootstrapping is that it can create a Self-Healing and Self-discoverable environment. Such a system is more resilient to hardware failure in Production.

What are the best practices to Automate deployment in AWS?

Some of the best practices to automate deployment in AWS are: **Library:** We can create a library of scripts that are frequently used for installation and configuration. **AMI:** We can manage the configuration

and deployment process using agents bundled inside an Amazon Machine Image. **Bootstrap:** We can Bootstrap the instances of components in AWS.

How will you automate your software infrastructure in AWS?

We can use following tactics to automate the software infrastructure in AWS: Auto-scaling: Amazon EC2 can be used for defining Auto-scaling groups for different clusters of servers. It helps in automated handling of traffic spikes and server failure.

CloudWatch: We can monitor vital metrics like- CPU, Memory, Disk I/O, and Network I/O of our servers by using Amazon CloudWatch. It can also help us in taking appropriate actions like launching new servers etc.

Simple DB: We can store and retrieve machine configuration information in machine images in AWS. This helps in automated deployment process. We can store these images in Simple DB in AWS. SimpleDB can also be used to store information about an instance such as its IP address, machine name and role.

Amazon S3: We can create an automated build process that dumps the latest builds into a bucket in Amazon S3. During startup an application read the latest version from Amazon S3 bucket. Failover: While creating AWS architecture, an application component should not assume that it would be up all the time. SO, we can dynamically attach the IP address of a new node to the cluster. Also, we can build automatic failover for servers and start a new clone in case of a hardware failure.

What are the AWS specific techniques for parallelization of software work?

We can use following techniques to parallelize the work in AWS:

Multi-threading: Amazon S3 can handle requests in multi-threading mode. We can create application that can serve concurrent requests from Amazon S3.

DB Requests: Amazon Simple DB also supports multiple threads. It can be used for concurrent GET requests to get data from Simple DB. For writing to DB, we can use BATCHPUT requests.

MapReduce: Another parallelization technique is to create a JobFlow by using Amazon Elastic MapReduce Service batch processes. It can make the long running tasks finish faster in MapReduce execution mode.

Elastic Load Balancing: Also, we can use Elastic Load Balancing service to distribute the load across multiple web app servers dynamically.

Why it is recommended to keep dynamic data closer to the compute and static data closer to the end user in Cloud computing?

Data proximity is an important principle of Cloud Computing. If we keep the right kind of data at right place, it can help build an excellent enterprise software system. The purpose of keeping dynamic data closer to compute resources is that it can reduce the latency while processing. There is no need for servers to fetch data from remote locations. Even MapReduce algorithm recommends keeping dynamic data nodes closer to compute servers.

Since there is always inherent network latency in a cloud computing environment, this practice can improve the overall performance of computation by saving time from data transfer between servers for processing. Another benefit is that in Cloud we pay for the in and out bandwidth by the GBs of data transfer. So, the cost of data transfer can increase overall costs. In case there is a big chunk of external data that has to be processed in the cloud, we first transfer the data to nodes near the execution environment.

And then process the data in parallel mode. It is a common practice in Data warehouse operations to first move the entire database in cloud and then process it in parallel threads. For multi-tier web applications data is stored into and retrieved from relational databases. In such a scenario the recommended architecture is to create app server and db nodes in same cloud environment. Generally there is free data transfer within cloud nodes. Keeping app and db nodes in same cloud can save time as well as money for internal data transfer. For static data like images, pdf, video etc., the recommended approach is to keep it closer to the end user. This kind of data can be cached in nodes that are closer to the user consuming it. This can drastically reduce the access latency for consumer, and provide better user experience.

What are the features in AWS for keeping static data closer to end user?

AWS provides following features to support the static data proximity to end user:

CloudFront: Amazon CloudFront can cache the content in an Amazon S3 bucket for multiple edge locations that are closer to the end user location.

Availability Zones: We can use the same Availability Zone to create a cluster of servers. This makes sure that data is in proximity to the processing servers.

Physically Ship Data: Yes, we can ship data drives to Amazon by using Import/Export service

Many at times it is cheaper and faster to move large amounts of data using the sneakernet than to upload it over the Internet.

What are the best practices to ensure the security of an application in cloud?

Following are the best practices to ensure the security of a cloud-based application:

Latest Patches: We should regularly download patches from a third-party vendor's web site and update our Amazon Machine Images (AMI).

Amazon Machine Image (AMI): It is advisable to redeploy server instances from the new Amazon Machine Images (AMI) and test the applications for any regression failure. The new patches should not break the existing functionality. **Uniform Deployment:** We have to ensure that all the instances are deployed with the latest AMI.

Test Automation: Automated test scripts have to be developed to run periodic security checks on applications in cloud.

Third Party: All the third-party software in cloud should be configured with the most secure settings.

Admin User: Whenever possible, the running of any process as a root or Administrator login should be avoided.

Why encryption should be used in Amazon S3?

Amazon S3 is simple storage service. We can create a highly-scalable, reliable, and low-latency data store in Amazon S3. We can use a simple web service interface to store and retrieve data in S3 buckets. These APIs are available at all the time from anywhere in the world. Since these APIs are widely accessible data stored needs security. To keep the data secure, we can encrypt it. Since S3 is Amazon proprietary technology, it is recommended to use our own Encryption strategy on the data stored in Amazon S3.

What are the best practices of Software Security in Cloud?

Some of the best practices of Software Security in cloud are:

Protect data in transit: During transmission of data from one place to another place, we should use secure socket layer (SSL). This is usually done by HTTPS protocol. To do this we need a certificate from a reputed certification authority like VeriSign. Based on the certificate the server can be authenticated by a client browser.

Virtual Private Cloud: We can create virtual private cloud by using Amazon VPC. This can help us in isolating the servers logically within AWS cloud. This can ensure that data transfer is secure within our virtual private cloud.

Protect data at rest: In case we have sensitive information like- Date of Birth, SSN, Passwords etc., we can encrypt this data. So that even if someone gets a copy of the data they cannot decrypt it easily. In Amazon S3, we should always encrypt the sensitive data.

Protect AWS credentials: In AWS there are different types of credentials. We AWS access keys that are used for accessing REST API. Since these keys are sent over web, we should use HTTPS protocol so that these cannot be compromised or tampered during transit.

Embedding Credentials in AMI: Some people make the mistake of embedding AWS credentials in Amazon Machine Image (AMI). We should pass these credentials as an argument during the launch of an AMI.

Key Rotation: We should keep rotating the secret access key on a regular basis. So that even if it is compromised, it cannot be used.

What automation tools can be used to create new servers in AWS?

We can use following ways to create new servers in AWS:

- **Puppet:** We can use tool like Puppet to write scripts that can create new servers in AWS.
- **Custom solution:** We can also write our own Perl/bash scripts to spin up new servers in AWS.
- **Opscode Chef:** We can use third party tools like Opscode Chef for creating new servers in AWS.



AWS Migration

Migration Questionnaire	Migration Steps	General Q&A	



What are the Checklist for AWS Cloud Migration?

Migration Checklist

Evaluate your Actual Environment

of Servers running =

Check for your current Storage:

How much data will you need to migrate?

Review your actual infrastructure and determine service version number. (Ex. php 5.6, Apache 2.4.x, etc)

--

1. Migration Status

Are you already in the Cloud?

If you already own an account and have something in the cloud, review:

Are you using the Correct type of servers? What type? -----

Have you migrated your application data?

- ☐ Yes
- ☐ No

Using Dedicated IP.

2. Databases

Are you running an optimized database server or on an RDS ?

Correct Database Engine used? (MariaDB, Percona DB, Mongo DB, MySQL)

Is the database separated from the Web Server?

- ☐ Server
- ☐ RDS

Is it highly available?

- ☐ Yes
- ☐ No

Is there any replication/Failover Solution?

- ☐ Yes
- ☐ No

3. Backups & Disaster Recovery

The perfect solution, recover from any disaster or hard situation

- Automated Snapshots.
- Automated AMI's.
- Content Being sent to Amazon S3.
- RDS Automated Snapshots.
- RDS Backup Policy in Days.

4. Security

- Assigned just the essential Security Groups to Instances
- Assigned just the essential Security Groups to Databases
- Is the environment secured in a VPC?

5. Monitoring - CloudWatch Metrics & Custom Metrics

Do not forget to monitor these metrics on your servers, so you can notice any trouble at any

- CPU Utilization
- Memory Available
- Disk IO
- NetworkIn
- NetworkOut
- Free Storage Space Available

[Operational Checklist](#)



aws-operational-checklists.pdf

Sample Migration Plan for AWS Cloud Migration?

Please download the migration plan to build for your project.



AWS Migration Plan.xlsx

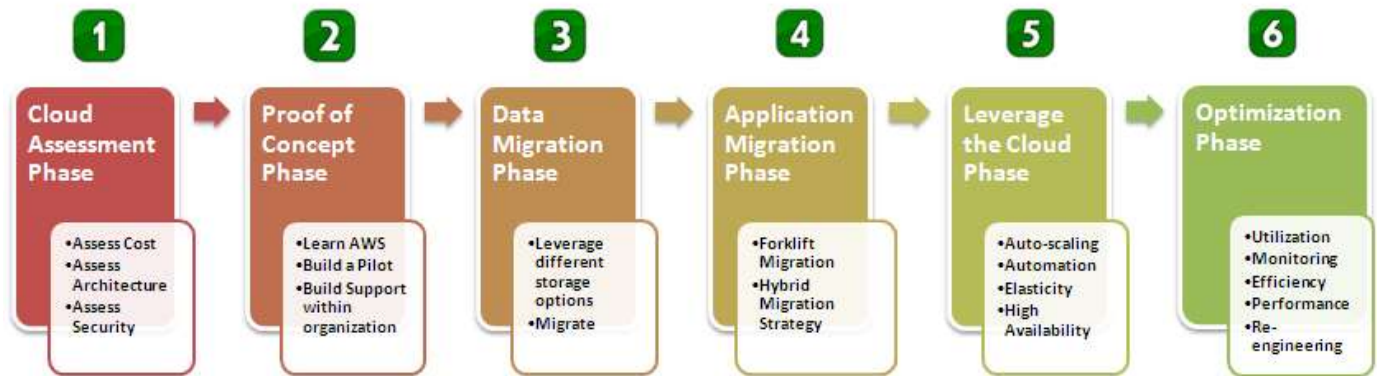
How will you migrate your existing application to AWS Cloud?



cloud-migration-main.pdf

What are the steps involved in for AWS Cloud Migration?

Migration Strategy



Phases	Benefits
Cloud Assessment <ul style="list-style-type: none"> Financial Assessment (TCO calculation) Security and Compliance Assessment Technical Assessment (Classify application types) Identify the tools that can be reused and the tools that need to be built Migrate licensed products Create a plan and measure success 	Business case for migration (Lower TCO, faster time to market, higher flexibility & agility, scalability + elasticity) Identify gaps between your current traditional legacy architecture and next -generation cloud architecture
Proof of Concept <ul style="list-style-type: none"> Get your feet wet with AWS Build a pilot and validate the technology Test existing software in the cloud 	Build confidence with various AWS services Mitigate risk by validating critical pieces of your proposed architecture
Moving your Data <ul style="list-style-type: none"> Understand different storage options in the AWS cloud Migrate file servers to Amazon S3 Migrate commercial RDBMS to EC2 + EBS Migrate MySQL to Amazon RDS 	Redundancy, Durable Storage, Elastic Scalable Storage Automated Management Backup
Moving your Apps <ul style="list-style-type: none"> Forklift migration strategy Hybrid migration strategy Build "cloud-aware" layers of code as needed Create AMIs for each component 	Future-proof scaled-out service-oriented elastic architecture
Leveraging the Cloud <ul style="list-style-type: none"> Leverage other AWS services Automate elasticity and SDLC Harden security Create dashboard to manage AWS resources Leverage multiple availability zones 	Reduction in CapEx in IT Flexibility and agility Automation and improved productivity Higher Availability (HA)
Optimization <ul style="list-style-type: none"> Optimize usage based on demand Improve efficiency Implement advanced monitoring and telemetry Re-engineer your application Decompose your relational databases 	Increased utilization and transformational impact in OpEx Better visibility through advanced monitoring and telemetry



AWS IoT

AWS IoT Core Connect Devices to the Cloud	Amazon FreeRTOS IoT Operating System for Microcontrollers	AWS Greengrass Local Compute, Messaging and Sync for Devices
AWS IoT 1-Click Once Click Creation of an AWS Lambda Trigger	AWS IoT Analytics Analytics for IoT Devices	AWS IoT Button Cloud Programmable Dash Button
AWS IoT Device Defender Security Management for IoT Devices	AWS IoT Device Management Onboard, Organize, and Remotely Manage IoT Devices	



Internet of Things

IoT Highlights

AWS IoT provides secure, bi-directional communication between Internet-connected devices such as sensors, actuators, embedded micro-controllers, or smart appliances and the AWS Cloud. This enables you to collect telemetry data from multiple devices, and store and analyze the data. You can also create applications that enable your users to control these devices from their phones or tablets.

AWS IOT Components are: -

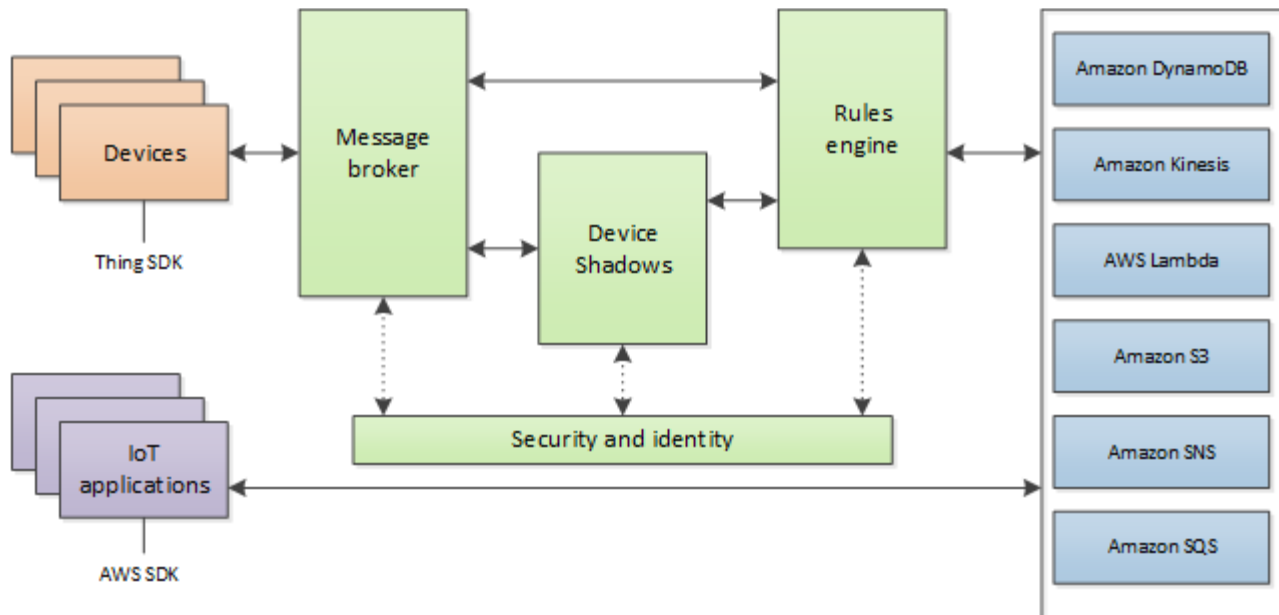
- **Device Gateway**
- **Message Broker**
- **Rules Engine**
- **Security and Identity Service**
- **Registry**
- **Group Registry**
- **Device Shadow**
- **Device Shadow Service**
- **Device Provisioning Service**
- **Custom Authentication Service**
- **Jobs Service**

Share the AWS IoT Configuration Step by Step?

Pre-requisites

To deploy an End to End AWS IoT Service

Topology



Pre-requisites

User should have AWS account, or IAM user with AWSIoTFullAccess

How to create resources required to send, receive, and process MQTT messages from devices using AWS IoT.

You need the following: -

- A computer with Wi-Fi access.
- If you have an AWS IoT button (pictured here), you can use it to complete this tutorial.
- If you do not have a button, you can purchase one or you can use the MQTT client in the AWS IoT console to emulate a device.



Tasks

Step 1: Set Up the Environment

- Create an SSH Keypair
- Deploy the AWS CloudFormation Template
- Confirmation: Connecting to your Instance

Step 2: Set Up AWS IoT

- AWS IoT Overview
- Create the AWS IoT Resources
- Create an IoT Thing
- Create an IoT Policy
- Create an IoT Certificate
- Configure and Run the Device Simulator
- Create an IoT Rule and Action
- Confirmation: View Device Messages with the AWS IoT MQTT Client

Step 3: Process and Visualize Streaming Data

- Dashboard Overview
- Create the IoT Rules and Actions
- Test the APIs
- Deploy the Real-Time Dashboard
- Host a Static Website on Amazon S3

Step 4: Clean Up the Environment

- Clean up IOT Resources
- Clean up the S3 bucket
- Delete the CloudFormation Stack

Step 1: Set Up the Environment

1.1 Create an SSH Keypair

To create your IoT environment, you will need to create an SSH keypair that will be used to access your device simulator EC2 instance. The following steps outline creating a unique SSH keypair to use in this lab.

1. Sign into the AWS Management Console and open the Amazon EC2 console at <https://console.aws.amazon.com/ec2>.

2. In the upper-right corner of the AWS Management Console, confirm you are in the desired AWS region. Make sure to select a region that supports AWS IoT

3. In the navigation pane on the left, under NETWORK & SECURITY, choose Key Pairs

The screenshot displays the AWS Management Console interface. On the left, the navigation pane is expanded to 'NETWORK & SECURITY', and 'Key Pairs' is highlighted with a red arrow. The main content area shows the 'Resources' section for the US West (N. California) region, listing various EC2 resources. Below this is a 'Create Instance' section with a 'Launch Instance' button. The right sidebar contains 'Account Attributes' and 'AWS Marketplace' sections.

Resources

You are using the following Amazon EC2 resources in the US West (N. California) region:

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

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 US West (N. California) region.

Service Health

Service Status:

- US West (N. California): This service is operating normally

Availability Zone Status:

- us-west-1b: Availability zone is operating normally
- us-west-1c: Availability zone is operating normally

[Service Health Dashboard](#)

Scheduled Events

US West (N. California):

- No events

Account Attributes

Supported Platforms

- VPC

Default VPC

- vpc-7d886e18

Additional Information

- [Getting Started Guide](#)
- [Documentation](#)
- [All EC2 Resources](#)
- [Forums](#)
- [Pricing](#)
- [Contact Us](#)

AWS Marketplace

Find free software trial products in the AWS Marketplace from the EC2 Launch Wizard. Or try these popular AMIs:

- Vyatta Virtual Router/Firewall/VPN

Provided by Vysitta, Inc. Rating: ★★★★★

Pay by the hour for software and AWS usage

[View all Networking](#)

[Alert Logic Threat Manager for AWS](#)

4. Choose Create Key Pair.



5. Enter a name for the new key pair in dialog box, and then choose Create.



The private key file is automatically downloaded by your browser. The base file name is the name you specified as the name of your key pair, and the file name extension is .pem. Save the private key file in a safe place.

Important: This is the only chance for you to save the private key file. You'll need to provide the name of your key pair when you launch an instance and the corresponding private key each time you connect to the instance.

1.2 Deploy the AWS CloudFormation Template

AWS CloudFormation is a service that helps you model and set up your Amazon Web Services resources as code so that you can spend less time managing those resources and more time focusing on your applications that run in AWS. We have created a template (written in JSON) that defines the AWS resources that are needed for the sample IoT application. AWS CloudFormation then uses that template to provision and configure those resources for you. You don't need to individually create and configure AWS resources and figure out what's dependent on what; AWS CloudFormation handles all of that.

1. Sign in to the AWS Management Console

2. If this is a new AWS CloudFormation account, click **Create New Stack**. Otherwise, click **Create Stack**.

3. In the Template section, select Specify an Amazon S3 Template URL to type or paste the following URL for the IoT Getting Started template:

<https://s3.amazonaws.com/awsprojects-code/iotGettingStartedTemplate.json>

The screenshot shows the 'Create stack' wizard in the AWS Management Console, specifically the 'Select Template' step. On the left, a sidebar contains links for 'Select Template' (highlighted), 'Specify Details', 'Options', and 'Review'. The main area is titled 'Select Template' and includes a descriptive paragraph: 'Select the template that describes the stack that you want to create. A stack is a group of related resources that you manage as a single unit.' Below this, there are three sections: 'Design a template' with a 'Design template' button; 'Choose a template' with a description of templates and three radio button options: 'Select a sample template' (with a dropdown menu), 'Upload a template to Amazon S3' (with a 'Choose File' button and 'No file chosen' text), and 'Specify an Amazon S3 template URL' (which is selected). The selected option has a text input field containing the URL 'on/static/code/iotGettingStartedTemplate.json' and a 'View/Edit template in Designer' link. At the bottom right, there are 'Cancel' and 'Next' buttons.

4. Click Next.

5. In the Stack name field, enter a friendly name for the IoT stack. A shorter name here will improve readability in future modules (e.g. IoTGS).

6. In the KeyName field, select the keypair you created earlier. This will "key" your EC2 instance with the appropriate public key.

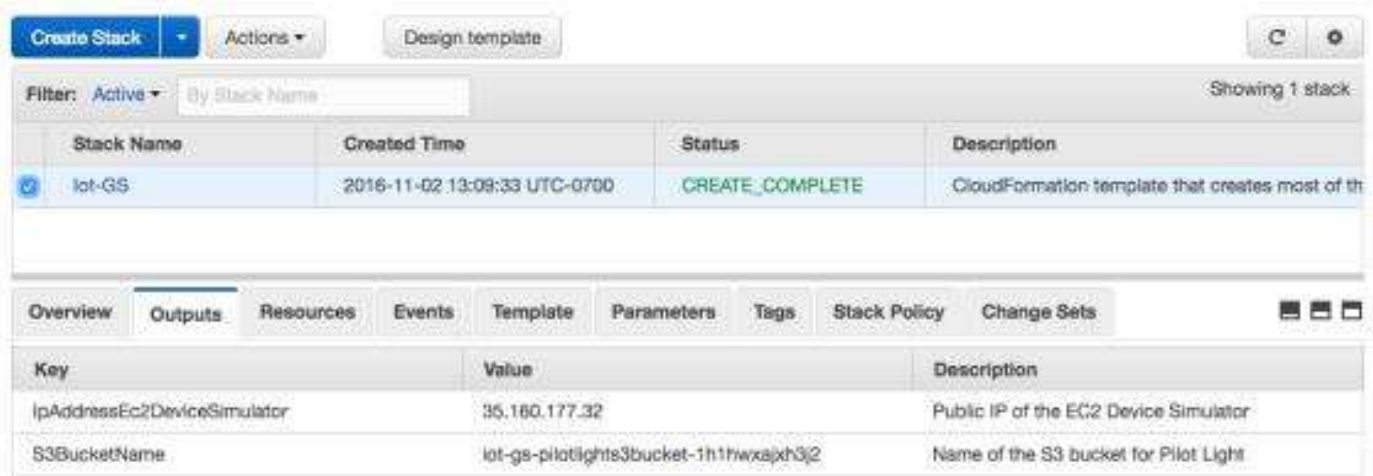
7. On the Options page, leave all defaults and click Next.

8. On the Review screen, confirm the configuration, check the box that says I acknowledge that AWS CloudFormation might create IAM resources, and click Create.

9. The environment can take a few minutes to provision completely. You can refresh periodically to monitor the creation of the stack. When AWS CloudFormation is finished creating the stack, the status will show `CREATE_COMPLETE`.

10. Select the check box beside your stack and then click on the Outputs tab below.

11. Note the `IpAddressEc2DeviceSimulator` Value. This is the public IP address of your IoT Device Simulator EC2 instance.



The screenshot shows the AWS CloudFormation console. At the top, there are buttons for 'Create Stack', 'Actions', and 'Design template'. Below these is a filter section with 'Filter: Active' and a search box 'By Stack Name'. A table lists the stacks, with one stack named 'iot-GS' having a status of 'CREATE_COMPLETE'. Below the stack list, there are tabs for 'Overview', 'Outputs', 'Resources', 'Events', 'Template', 'Parameters', 'Tags', 'Stack Policy', and 'Change Sets'. The 'Outputs' tab is selected, showing a table with two outputs: 'IpAddressEc2DeviceSimulator' with the value '35.160.177.32' and 'S3BucketName' with the value 'iot-gs-pilotlights3bucket-1h1hwaxph3j2'.

Stack Name	Created Time	Status	Description
iot-GS	2016-11-02 13:09:33 UTC-0700	CREATE_COMPLETE	CloudFormation template that creates most of th

Key	Value	Description
IpAddressEc2DeviceSimulator	35.160.177.32	Public IP of the EC2 Device Simulator
S3BucketName	iot-gs-pilotlights3bucket-1h1hwaxph3j2	Name of the S3 bucket for Pilot Light

1.3 Confirmation: Connecting to your Instance

We will now confirm that we have access to the EC2 instance that will be simulating the IoT devices. Follow the instructions for your operating system.

Mac or Linux (OpenSSH)

By default, both Mac OS X and Linux operating systems ship with an SSH client that you can use to connect to your EC2 Linux instances. To use the SSH client with the key you created, a few steps are required.

1. Use the following command to set the permissions of your private key file so that only you can read it. Replace `IoT-GettingStarted-Key.pem` with the name of your SSH key pair.

```
$ chmod 400 IoT-GettingStarted-Key.pem
```

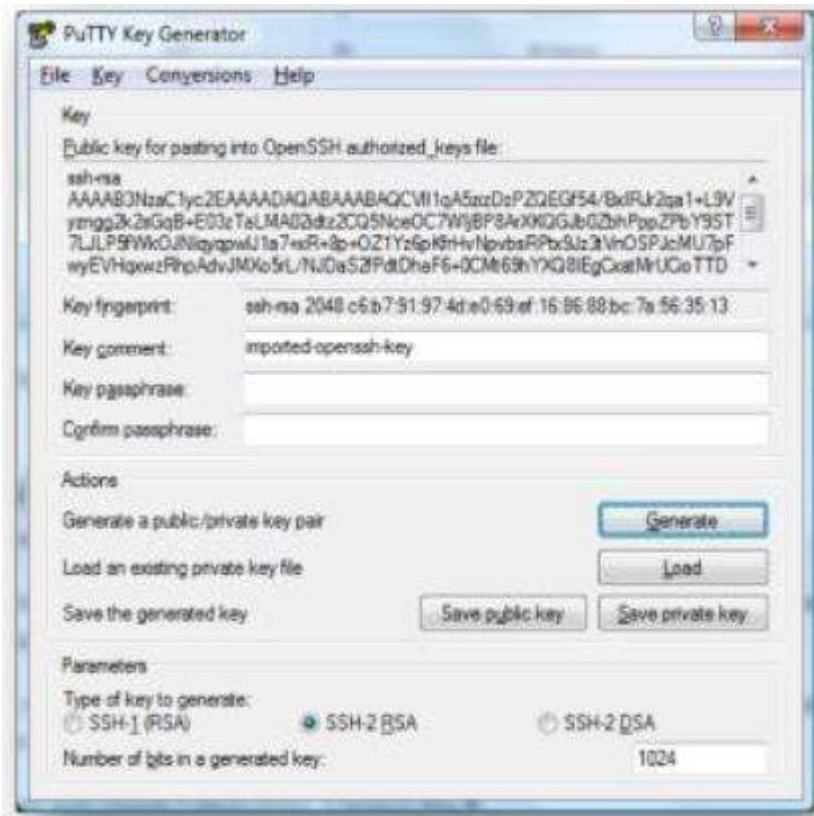
2. Use your private key when connecting to the instance. You will reference your private key file and the default user name which is `ec2-user`. The format of the ssh client is as follows: `$ ssh -i IoT-GettingStarted-Key.pem ec2-user@<IP Address of EC2 Host>`

3. Type "Yes" to accept the fingerprint. You should now be connected to your instance.

Windows (PuTTY)

This is a Windows-only step, because other operating systems have SSH built in. Download and install PuTTY. The single word "putty" in Google will return a list of download sites. Be certain that you install both PuTTY and PuTTYGen

1. Launch PuTTYGen and choose Conversions -> Import Key. Browse for the downloaded pem file (e.g., IoT-GettingStarted-Key.pem) and import the key. The result will look similar to this:

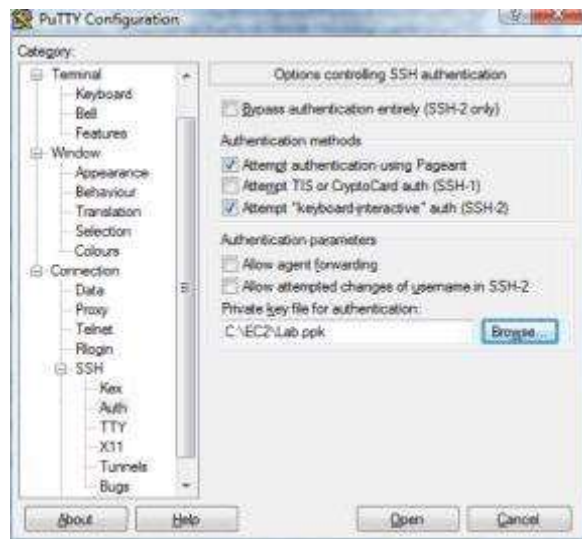


2. Save the key as the same file name with a .ppk extension. Click File -> Save as Private Key. Ignore the dialog that asks if you want to do this without a passphrase.

3. Close PuTTYGen.

4. Open PuTTY.

5. On the left menu expand Connection -> SSH and select the Auth sub-menu. Click Browse and select your PPK file from the previous step.



6. Select Connection and configure the keepalive to 60. This will prevent your SSH session from timing out.



7. Select Session on the left. In the Host Name box, enter ec2-user@ followed by the IP address of your Simulator EC2 instance. (e.g. ec2-user@ 50.17.175.10).

8. Click Yes to confirm the fingerprint.



Note: The SSH fingerprint will eventually show up in the System Log and you can take that and compare it to protect against a man in the middle attack.

9. You should now be connected to your instance.

Step 2: Set Up AWS IoT

2.1 AWS IoT Overview

AWS IoT consists of the following components: -

- **Message Broker** — Provides a secure mechanism for things and AWS IoT applications to publish and receive messages from each other. You can use either the MQTT protocol directly or MQTT over WebSockets to publish and subscribe. You can use the HTTP REST interface to publish.
- **Rules Engine** — Provides message processing and integration with other AWS services. You can use a SQL-based language to select data from message payloads, process the data, and send the data to other services, such as Amazon S3, Amazon DynamoDB, and AWS Lambda. You can also use the message broker to republish messages to other subscribers.
- **Thing Registry** — Sometimes referred to as the Device Registry. Organizes the resources associated with each thing. You register your things and associate up to three custom attributes with each thing. You can also associate certificates and MQTT client IDs with each thing to improve your ability to manage and troubleshoot your things.
- **Thing Shadows Service** — Provides persistent representations of your things in the AWS cloud. You can publish updated state information to a thing shadow, and your thing can synchronize its state when it connects. Your things can also publish their current state to a thing shadow for use by applications or devices.
- **Thing Shadow** — Sometimes referred to as a device shadow. A JSON document used to store and retrieve current state information for a thing (device, app, and so on).
- **Device Gateway** — Enables devices to securely and efficiently communicate with AWS IoT. Security and Identity service—Provides shared responsibility for security in the AWS cloud. Your things must keep their credentials safe in order to send data securely to the message broker. The message broker and rules engine use AWS security features to send data securely to devices or other AWS services.

2.2 Create the AWS IoT Resources

Now you will create the resources needed in the AWS IoT console. There are 4 components that will need to be created: -

- **Thing** – A logical representation of a device stored in IoT's Registry. Supports attributes, as well as Device Shadows, which can be used to store device state & define desired state.
- **Policy** – Attached to Certificates to dictate what that Certificate (or rather, a Thing using that certificate) is entitled to do on AWS IoT.

- **Certificate** – Things can communicate with AWS IoT via MQTT, MQTT over WebSockets or HTTPS. MQTT is a machine-to-machine pub-sub protocol well-suited for IoT use cases given its low overhead and low resource requirements. MQTT
- transmission to your AWS IoT gateway is encrypted using TLS and authenticated using certs you will create.
- **Rule** – Leverages AWS IoT's Rules Engine to dictate how messages sent from Things to AWS IoT are handled. You will configure rules that send data published to an MQTT topic to a variety of AWS Services.

2.3 Create an IoT Thing

1. Sign in to the AWS IoT console.
2. On the left side of the console, click on **Registry**, then click **Things**.



Dashboard



Connect



Registry

Things

Types



Security

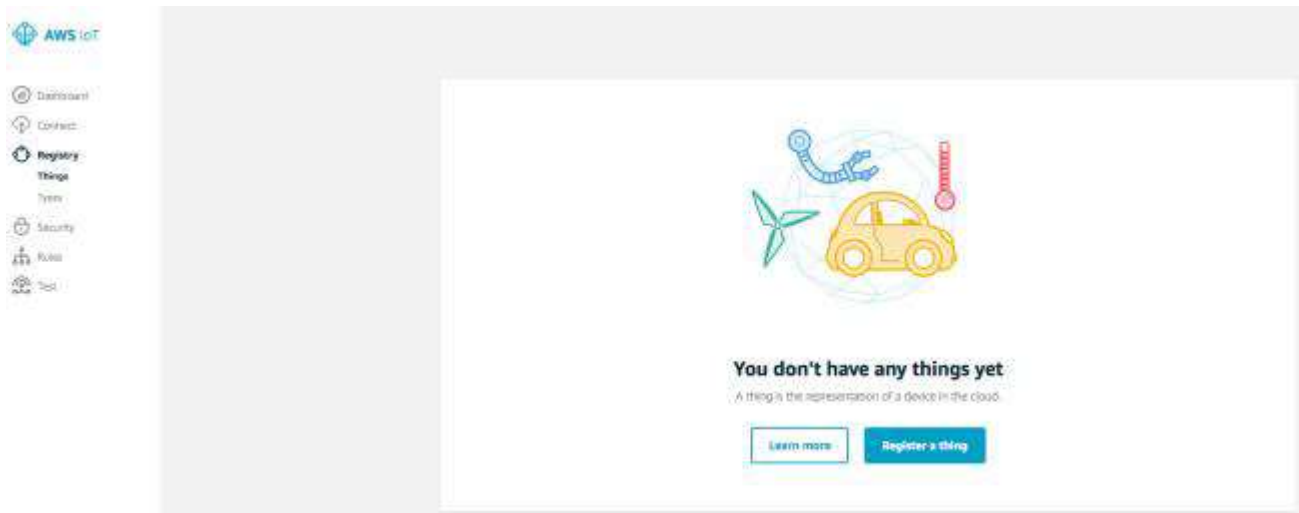


Rules

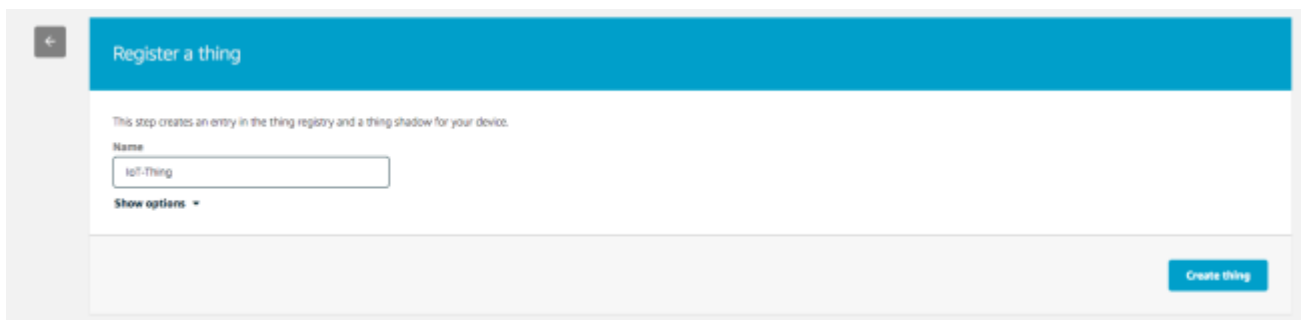


Test

3. If you have never used the service before, then click [Register a thing](#). Otherwise, [Create](#), will be in the top right corner



4. Provide a name for the Thing and click [Create thing](#).

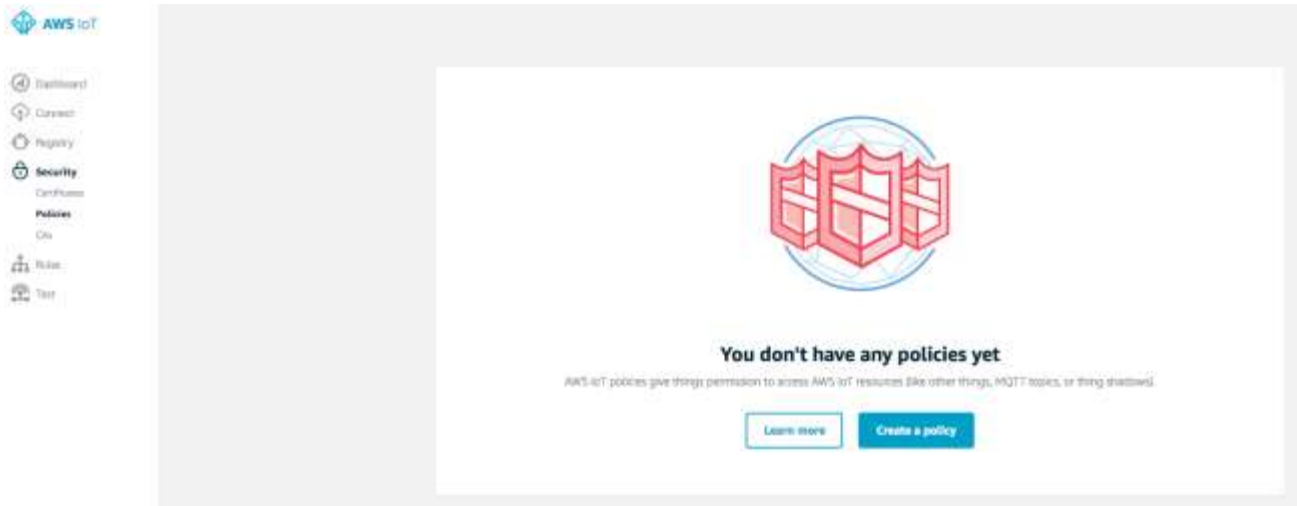


5. On the Thing Detail page, click on [Interact](#) in the left side menu. Capture the Rest API Endpoint (e.g. a2ipckzivgv00u.iot.us-west-2.amazonaws.com) listed under HTTPS. You will need this host name to configure the Device Simulator.



2.4 Create an IoT Policy

1. From the AWS IoT Console, select **Security**, and then **Policies**. If you have never used the service before, then click on **Create a Policy**. If you have never, used the service before, then click. If a previous policy exists, then you will click **Create** on the top right.



2. Give the Policy a **Name**.

3. Replace the **Action** with *iot:**

4. For the **Resource ARN**, replace the statement with ***.

5. The **Create** button should turn blue. Click it to complete the policy creation.

A screenshot of the 'Create a policy' form in the AWS IoT console. The form has a blue header bar with the text 'Create a policy'. Below the header, there is a description: 'Create a policy to define a set of authorized actions. You can authorize actions on one or more resources (things, topics, topic filters)'. The 'Name' field contains the text 'iot-policy'. Below this is the 'Add statements' section, which includes a sub-header 'Policy statements define the types of actions that can be performed by a resource.' and a toggle for 'Advanced mode'. The 'Action' field contains 'iot:*'. The 'Resource ARN' field contains '*'. The 'Effect' section has two radio buttons: 'Allow' (selected) and 'Deny'. At the bottom right of the 'Add statements' section is a 'Remove' button. At the bottom left of the form is an 'Add statement' button. At the bottom right of the entire form is a blue 'Create' button.

2.5 Create an IoT Certificate

While it is possible to create the device certificates in the AWS Management Console, we have created these during the CloudFormation stack creation via a script that runs on your EC2 device simulator instance.

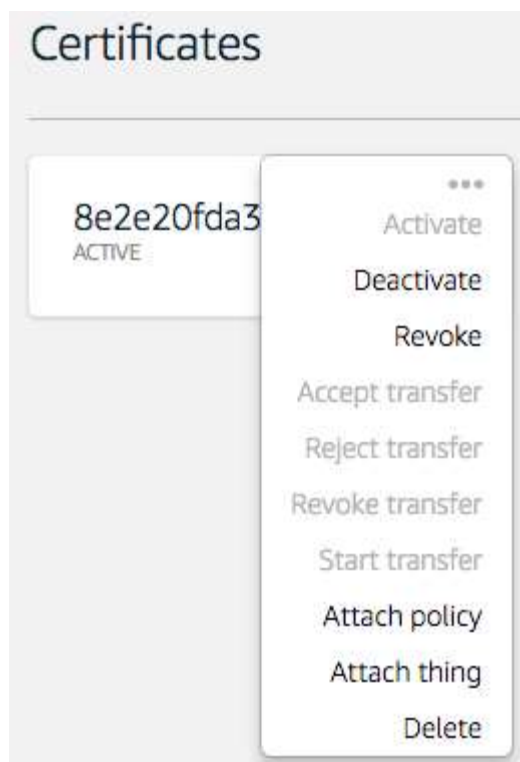
1. SSH into the EC2 Instance.

2. Type `ls ~/certs` to view the certificates were created. You should have 3 files in the directory

- certificate.pem.crt
- private.pem.key
- root-ca.pem

3. In the AWS IoT Console window, click on [Security](#) and [Certificates](#). You should see your certificate. Confirm that it says [ACTIVE](#).

4. Now the certificate must be associated with Thing and Policy that were created previously. Click on Options (...) on the top right of the certificate and click on [Attach policy](#).



5. Select the policy you just created and click [Attach](#).



6. Repeat the process, selecting [Attach a thing](#). Select the Thing you created earlier.

2.6 Configure and Run the Device Simulator

An example script is provided that will send messages containing current battery charge, simulated GPS location data, as well as other telemetry data. The AWS IoT Service will process these messages and send to the appropriate AWS services based on the rule actions that you will configure throughout the workshop modules.

1. SSH into the EC2 instance.

2. Open the file *settings.py* in the editor of your choice. We will be using nano in this example. `$ nano ~/settings.py`

3. Replace the **HOST_NAME** with the host name REST API Endpoint of your Thing (e.g., *a2ipckzivgv00u.iot.us-east-1.amazonaws.com*).

4. Save the file. In nano, press **CTRL-X**, Type **Y** to save changes, and press **enter** to save the file as *settings.py*.

5. Start the device simulator.

```
$ nohup python app.py &
```

2.7 Create an IoT Rule and Action

IoT Rule Actions give your devices the ability to interact with AWS services. Rules are analyzed and actions are performed based on the MQTT topic stream. The simulated IoT devices report current battery charge percentage which decreases over time. We will create a rule action that will monitor the reported battery charge and publish a message to a new topic when it is time to recharge. The device is subscribed to this topic and will "take action" to recharge.

1. In the AWS IoT console, click on **Rules** on the left and then click **Create a Rule**.
2. Configure the rule as follows: -

Field	Value
Name	gsRecharge
Description	leave blank
Attribute	*
Topic Filter	device/+/devicePayload
Condition	batteryCharge <=0

Create a rule

Create a rule to evaluate messages sent by your things and specify what to do when a message is received (for example, write data to a DynamoDB table or invoke a Lambda function).

Name

Description

Message source

Indicate the source of the messages you want to process with this rule.

Using SQL version [?](#)

2016-05-28

Rule query statement

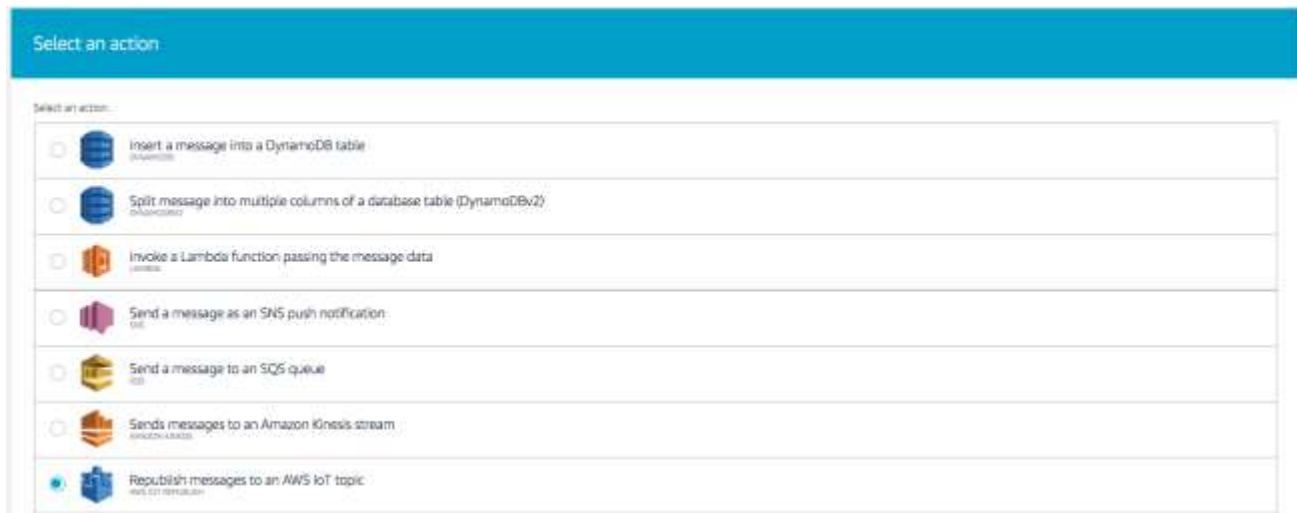
```
SELECT * FROM "device/+/devicePayload" WHERE batteryCharge <=0
```

Attribute

Topic filter

Condition

3. Click **Add action** and select **Republish messages to an AWS IoT topic**. Click on **Configure action**.



4. In the **Topic** dialog box, type *device/\${topic(2)}/rechargeAlert*.

5. Click the **IAM role name** dropdown box and select the role that begins with the stack name you configured followed by **AwsIotRepublishRole**.

6. Click **Add action**.

7. Click **Create rule**.

2.8 Confirmation: View Device Messages with the AWS IoT MQTT Client

Devices publish MQTT messages on topics. You can use the AWS IoT MQTT client to subscribe to these topics to see the content of these messages. We will now use the AWS IoT MQTT client to confirm that the IoT messages are being sent back and forth between the devices and the AWS IoT Device Gateway.

1. In the AWS IoT console, click on **Test**.



2. In the **Subscription topic** box, type the wildcard character **#** and click **Subscribe to topic**.

3. Click on the **#** symbol on the left pane under **Subscriptions**.

4. If the devices are successfully configured, you will see MQTT messages scrolling on the as pictured below.

Step 3: Process and Visualize Streaming Data

3.1 Dashboard Overview

DynamoDB is a fast and flexible NoSQL database service for all applications that need consistent, single-digit millisecond latency at any scale. It is a fully managed cloud database and supports both document and key-value store models. You will create a set of AWS IoT Rule Actions to write device messages to your DynamoDB tables.

Amazon API Gateway: Amazon API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale. For this module, your device data API has been created for you by CloudFormation, but you will have an opportunity to interact with your API configuration.

AWS Lambda: AWS Lambda lets you run code without provisioning or managing servers. With Lambda, you can run code for virtually any type of application or backend service - all with zero administration. Just upload your code and Lambda takes care of everything required to run and scale your code with high availability. For this module, the Lambda function has already been created for you and integrated with API Gateway, but in Module 5 you will create a Lambda function of your own.

Note on architecture: In this section you'll be building a dashboard that renders messages from your devices by pulling data from an Amazon DynamoDB table via a serverless API. An alternative pattern for this would include subscribing the dashboard to MQTT topics via WebSockets. This tutorial uses Amazon DynamoDB to illustrate both AWS IoT and serverless architectures.

In this section, you will create additional IoT rule actions to send device data to Amazon DynamoDB. There are 3 IoT devices in our setup, the devices are named: turing, hopper, and knuth. We will also set up a static website using Amazon S3 that will serve as a real-time dashboard allowing visualization of the device payload data. Each device sends the following JSON Payload to the AWS IoT Gateway every 5 seconds:

```
{
  "Items": [
    {
      "payload": {
        "timestampIso": "2016-09-10T22:35:06.732142",
        "batteryDischargeRate": 1.8513595745796365,
        "location": {
          "lon": 99.13799750347447,
```

```
"lat": 41.79078293335809
},
"timestampEpoch": 1473546906732,
"numVal": 5,
"deviceId": "Hopper",
"batteryCharge": 96.29728085084074
},
"deviceId": "Hopper"
},
{
"payload":
{
"timestampIso": "2016-09-10T22:35:06.732247",
"batteryDischargeRate": 1.5474383816808281,
"location": {
"lon": 114.60811541199776,
"lat": 41.15078293335809
},
"timestampEpoch": 1473546906732,
"numVal": 10,
"deviceId": "Turing",
"batteryCharge": 10.248573862511861
},
"deviceId": "Turing"
},
{
"payload":
{
"timestampIso": "2016-09-10T22:35:06.732010",
"batteryDischargeRate": 1.5634519980188246,
"location": {
"lon": 95.74692230611322,
"lat": 46.82078293335809
},
"timestampEpoch": 1473546906732,
"numVal": 6,
"deviceId": "Knuth",
"batteryCharge": 90.61928801188708
},
"deviceId": "Knuth"
}
],
"Count": 3,
"ScannedCount": 3
}
```

3.2 Create the IoT Rules and Actions

We will create a rule with two actions, to query the incoming messages and capture the payload section. The first rule will write time series data from devices to DynamoDB table called [IoT Dynamo Time Series Table](#). The second rule will write the latest received messages to a DynamoDB table called [IoT Dynamo Device Status Table](#).

Note: The actual DynamoDB table names will be prefixed by the name you chose for your CloudFormation stack, e.g. *IoTGS-DynamoTimeSeriesTable*.

1. In the AWS IoT console, click on [Rules](#) on the left and then click [Create](#).
2. Enter the following parameters. This rule includes a query statement that will capture the payload section from the incoming messages.

Field	Value
Name	IoTToDynamo
Description	leave blank
Attribute	*
Topic Filter	device/+ /devicePayload

Create a rule

Create a rule to evaluate messages sent by your things, and specify what to do when a message is received (for example, write data to a DynamoDB table or invoke a Lambda function).

Name
IoTToDynamo

Description

Message source
Indicate the source of the messages you want to process with this rule.
Using SQL version 2016-03-23

Rule query statement
SELECT * FROM 'device/+ /devicePayload'

Attribute
*

Topic filter
device/+ /devicePayload

Condition
e.g. temperature > 75

3. Click [Add action](#).
4. Select [Insert a message into a DynamoDB table](#) and then click [Configure action](#).
5. Click the [Table name](#) field and select the table whose name contains [Timeseries Table](#).

6. Enter the following parameters. This action will write the payload to DynamoDB table using the timestamp as a range key value.


Field	Value
Hash key value	<code>\${topic(2)}</code>
Range key value	<code>\${timestampEpoch}</code>
Role name	<code><stack-name>-AwsIoTToDynamoRole-<random-number></code>

The screenshot shows the 'Insert a message into a DynamoDB table' configuration page in the AWS IoT console. The page has a header with the title and a 'previous' link. Below the header, a message states 'The table must contain Hash and Range keys.' There are two main sections: 'Table name' and 'IAM role name'. The 'Table name' section includes a dropdown menu with the value 'iot-GS-IoTDynamoTimestampTable-WITHAW4CZDS' and a 'Create a new resource' button. The 'IAM role name' section includes a dropdown menu with the value 'iot-GS-AwsIoTToDynamoRole-UTDOLKXZVDM1' and a 'Create a new role' button. Below these sections, there are input fields for 'Hash key', 'Hash key type', 'Hash key value', 'Range key', 'Range key type', and 'Range key value'. The 'Hash key' field contains 'deviceId', 'Hash key type' contains 'STRING', and 'Hash key value' contains '\$deviceId'. The 'Range key' field contains 'payloadTimestamp', 'Range key type' contains 'NUMBER', and 'Range key value' contains '\$timestampEpoch'. There is also a text area for 'Write message data to this value'. At the bottom right, there is an 'Add action' button.

7. Click **Add action**.

8. We will also be creating a table of connected devices using this same IoT rule that reports the last reported value from the devices. We will create an additional action to accomplish this. Click **Add action** and repeat the process with the following values.

Field	Value
Table name	<code><stack-name>-IoTDynamoDeviceStatusTable</code>
Hash key value	<code>\${topic(2)}</code>
Range key value	leave empty
Role name	<code><stack-name>-AwsIoTToDynamoRole-<random-number></code>

 Insert a message into a DynamoDB table

The table must contain Hash and Range keys.

*Table name
 [Create a new resource](#)

*Hash key <input type="text" value="deviceId"/>	*Hash key type <input type="text" value="STRING"/>	*Hash key value <input type="text" value="\$topicGZI"/>
*Range key <input type="text" value="Optional field does not exist"/>	*Range key type <input type="text" value="Optional field does not exist"/>	*Range key value <input type="text"/>

Write message data in this column:

Choose or create a role to grant AWS IoT access to the DynamoDB resource to perform this action.

*IAM role name
 [Create a new role](#)

Give AWS IoT permission to send a message to the selected resource: [Update role](#)

[Add action](#)

9. Click [Add action](#).

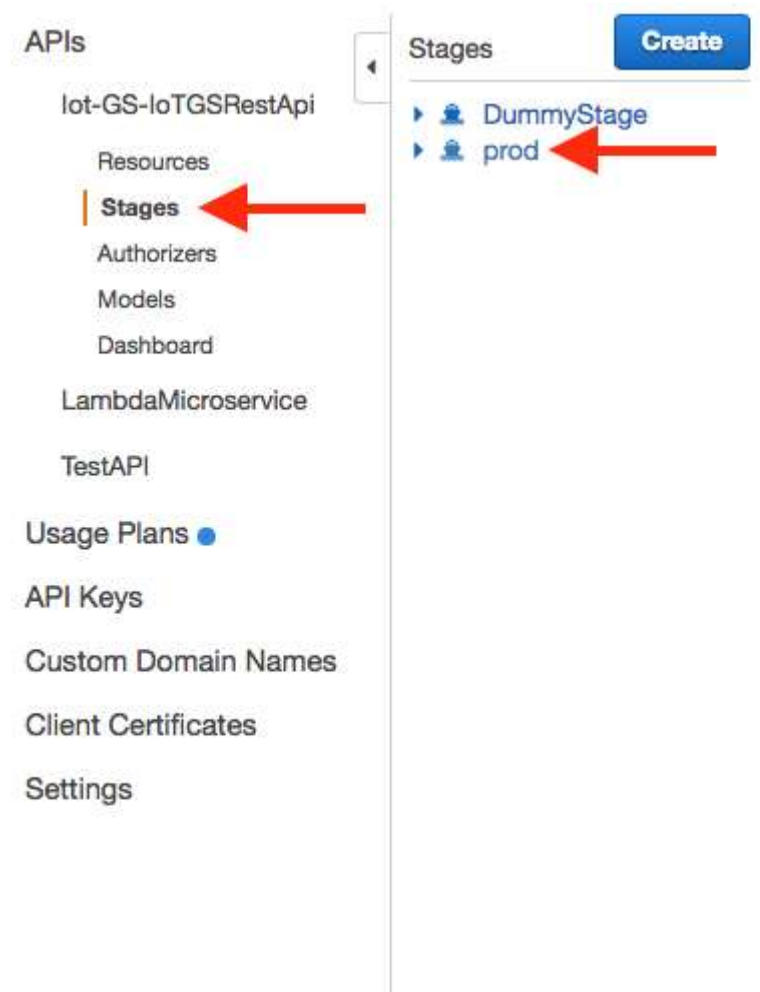
10. Click [Create rule](#).

Rule and actions are now configured; in the next step you'll enable the APIs that read the DynamoDB table and return devices data in an API GET method.

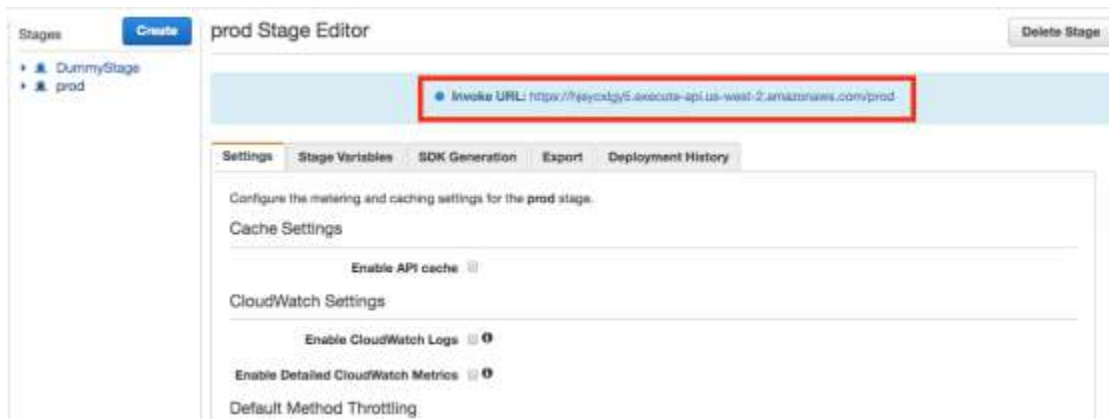
3.3 Test the APIs

In this section you will test that the API works, you will use a command line to read the data via HTTP. The API definition and the backing AWS Lambda function that support the API were configured for you when you provisioned the CloudFormation template. In the next section, you will "hook" the APIs into a dashboard to visualize the data in the website.

1. In Amazon API Gateway, a stage defines the path through which an API deployment is accessible. The CloudFormation template already configured a production stage called 'prod'. In the API Gateway console click **Stages** and then **prod**, and review the current API configuration.



2. Click on the link next to Invoke URL:



You should see devices data in a JSON format, refresh the page every few seconds and notice that data changes

3. Save the URL endpoint, you will need it in the next section.

3.4 Deploy the Real-Time Dashboard

In this section, you will visualize device data in a dashboard. The dashboard will place the devices in a map based on Geolocation (lon,lat), Battery Charge and Battery Discharge Rate will be displayed in a line chart. First, you will download the dashboard code and update API endpoint, and then you will setup a static website on S3 to host your dashboard.

1. SSH to the EC2 instance

2. Open app.js for editing in nano. `$ nano ~/dashboard/app.js`.

At the top of the file set the `devices_endpoint_url` to the API endpoint you'd created in the previous stage.

```
app.js
1  /* Enter Device Status Endpoint URL here */
2
3  var devices_endpoint_url = 'https://ff30e024i3c.execute-api.us-west-2.amazonaws.com/prod';
4
```

3. Save the file. In nano, press **CTRL-X**, Type **Y** to save changes, and press **enter** to save the file.

3.5 Host a Static Website on Amazon S3

In this step, you will configure a static website on S3 bucket. To host your static website, you configure an Amazon S3 bucket for website hosting and then upload your website content to the bucket. The website is then available at the region-specific website endpoint of the bucket.

1. From the AWS console, select [Services](#) and then [S3](#).
2. The CloudFormation template already created a bucket to hold the Dashboard, the bucket name is: [<CloudFormation Template Name>-iotgss3bucket-<Random Number>](#). Click on the bucket name, then click on [Properties](#).
3. Select [Permissions](#) and [Add bucket policy](#).
4. Paste the section below in the Bucket Policy Editor. Replace *<bucket-name>* with the name of your S3 bucket with the following policy enables anyone to read the bucket (execute GET HTTP command):

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow Public Access to All Objects",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::<bucket-name>/*"
    }
  ]
}
```
5. Click [Save](#).
6. Now you can enable static website hosting on the bucket, select the [Static Website Hosting](#) and check the [Enable website hosting](#) radio box. In the [Index Document](#) field enter *index.html* and click [Save](#).
7. Your dashboard will be available on the bucket's [Endpoint](#). Save this endpoint - you will use it shortly.
8. Copy the dashboard code to the S3 bucket. In a terminal window, make sure you are in the Dashboard directory. Copy the content of the directory to the S3 bucket, you will use the AWS CLI for this. type the following command: `aws s3 sync ~/dashboard s3://<bucket-name>`
9. In a browser, paste the S3 bucket endpoint to access your dashboard.

Devices Geolocation



turing

Battery Charge



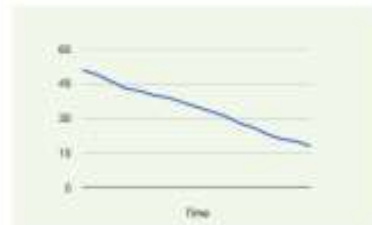
Battery Discharge



Sensor Data

hopper

Battery Charge



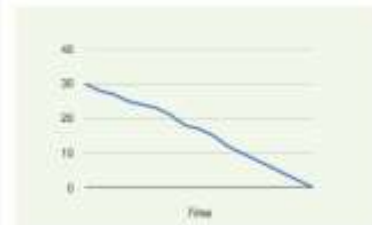
Battery Discharge



Sensor Data

knuth

Battery Charge



Battery Discharge



Sensor Data

Step 4: Clean Up the Environment

We will now delete all of the AWS resources that were used during this session.

4.1 Clean up IOT Resources

1. Sign in to the AWS IoT console.
2. Click on [Rules](#).
3. For each rule, Click ... and select [Delete](#).
4. In the confirmation window, click [Yes, continue with delete](#).
5. Click on [Security](#) and then [Policies](#)
6. Click ... and select [Delete](#).
7. In the confirmation window, click [Yes, continue with delete](#).
8. Under [Security](#), click [Certificates](#).
9. Click ... and select [Delete](#).
10. In the confirmation window, click [Yes, continue with delete](#).
11. Under [Registry](#) click [Things](#).
12. Click ... and select [Delete](#).
13. In the confirmation window, click [Yes, continue with delete](#).

4.2 Clean up the S3 bucket

1. Open the AWS S3 Console.
2. Right-click on the IoT bucket that we have been using and click [Empty Bucket](#).
3. You will need to type the name of bucket into the confirmation window. You can cut and paste for convenience. Click [Empty Bucket](#).

4.3 Delete the CloudFormation Stack

1. Open the AWS CloudFormation Console.
2. Check the box next to your IoT stack.
3. Under [Actions](#), click [Delete stack](#) and confirm. This may take up to 5 minutes to complete.

What is AWS IoT?

AWS IoT provides secure, bi-directional communication between Internet-connected devices such as sensors, actuators, embedded micro-controllers, or smart appliances and the AWS Cloud. This enables you to collect telemetry data from multiple devices, and store and analyze the data. You can also create applications that enable your users to control these devices from their phones or tablets.

What are the AWS IoT Components?

AWS IoT consists of the following components: -

Device gateway

Enables devices to securely and efficiently communicate with AWS IoT.

Message broker

Provides a secure mechanism for devices and AWS IoT applications to publish and receive messages from each other. You can use either the MQTT protocol directly or MQTT over WebSocket to publish and subscribe. You can use the HTTP REST interface to publish.

Rules engine

Provides message processing and integration with other AWS services. You can use an SQL-based language to select data from message payloads, and then process and send the data to other services, such as Amazon S3, Amazon DynamoDB, and AWS Lambda. You can also use the message broker to republish messages to other subscribers.

Security and Identity service

Provides shared responsibility for security in the AWS Cloud. Your devices must keep their credentials safe in order to securely send data to the message broker. The message broker and rules engine use AWS security features to send data securely to devices or other AWS services.

Registry

Organizes the resources associated with each device in the AWS Cloud. You register your devices and associate up to three custom attributes with each one. You can also associate certificates and MQTT client IDs with each device to improve your ability to manage and troubleshoot them.

Group registry

Groups allow you to manage several devices at once by categorizing them into groups. Groups can also contain groups—you can build a hierarchy of groups. Any action you perform on a parent group will apply to its child groups, and to all the devices in it and in all of its child groups as well.

Permissions given to a group will apply to all devices in the group and in all of its child groups.

Device shadow

A JSON document used to store and retrieve current state information for a device.

Device Shadow service

Provides persistent representations of your devices in the AWS Cloud. You can publish updated state information to a device's shadow, and your device can synchronize its state when it connects. Your devices can also publish their current state to a shadow for use by applications or other devices.

Device Provisioning service

Allows you to provision devices using a template that describes the resources required for your device: a thing, a certificate, and one or more policies. A thing is an entry in the registry that contains attributes that describe a device. Devices use certificates to authenticate with AWS IoT. Policies determine which operations a device can perform in AWS IoT.

The templates contain variables that are replaced by values in a dictionary (map). You can use the same template to provision multiple devices just by passing in different values for the template variables in the dictionary.

Custom Authentication service

You can define custom authorizers that allow you to manage your own authentication and authorization strategy using a custom authentication service and a Lambda function. Custom authorizers allow AWS IoT to authenticate your devices and authorize operations using bearer token authentication and authorization strategies.

Custom authorizers can implement various authentication strategies (for example: JWT verification, OAuth provider call out, and so on) and must return policy documents which are used by the device gateway to authorize MQTT operations.

Jobs Service

Allows you to define a set of remote operations that are sent to and executed on one or more devices connected to AWS IoT. For example, you can define a job that instructs a set of devices to download and install application or firmware updates, reboot, rotate certificates, or perform remote troubleshooting operations.

To create a job, you specify a description of the remote operations to be performed and a list of targets that should perform them. The targets can be individual devices, groups or both.

How AWS IoT Works?

AWS IoT enables Internet-connected devices to connect to the AWS Cloud and lets applications in the cloud interact with Internet-connected devices. Common IoT applications either collect and process telemetry from devices or enable users to control a device remotely.

Devices report their state by publishing messages, in JSON format, on MQTT topics. Each MQTT topic has a hierarchical name that identifies the device whose state is being updated. When a message is published on an MQTT topic, the message is sent to the AWS IoT MQTT message broker, which is responsible for sending all messages published on an MQTT topic to all clients subscribed to that topic.

Communication between a device and AWS IoT is protected through the use of X.509 certificates. AWS IoT can generate a certificate for you or you can use your own. In either case, the certificate must be registered and activated with AWS IoT, and then copied onto your device. When your device communicates with AWS IoT, it presents the certificate to AWS IoT as a credential.

We recommend that all devices that connect to AWS IoT have an entry in the registry. The registry stores information about a device and the certificates that are used by the device to secure communication with AWS IoT.

You can create rules that define one or more actions to perform based on the data in a message. For example, you can insert, update, or query a DynamoDB table or invoke a Lambda function. Rules use expressions to filter messages. When a rule matches a message, the rules engine invokes the action using the selected properties. Rules also contain an IAM role that grants AWS IoT permission to the AWS resources used to perform the action.

Each device has a shadow that stores and retrieves state information. Each item in the state information has two entries: the state last reported by the device and the desired state requested by an application. An application can request the current state information for a device.

The shadow responds to the request by providing a JSON document with the state information (both reported and desired), metadata, and a version number. An application can control a device by requesting a change in its state. The shadow accepts the state change request, updates its state information, and sends a message to indicate the state information has been updated. The device receives the message, changes its state, and then reports its new state.



AWS Lab

AMAZON EC2 - LAB

- Create an AWS account.
- Login to AWS account and navigate to EC2 service.
- Launch on 64bit Linux instance based on Amazon Linux AMI.
- Generate key pair and define security group.
- Access the new instance using Putty or any other SSH client.
- Install PHP and Apache.
- Build new AMI of running instance.
- Transfer AMI in any other region.

AMAZON ELB- LAB

- Navigate to EC2 service.
- Make sure you have two EC2 instance running.
- Install PHP and Apache and create index.html as default page.
- Navigate to Load Balancers under “Network and Security”.
- Create a new load balancer and add both instances.
- Once active, it will generate a new ELB URL.
- Try to access the URL. You should be able to see output of index.html
- Now shutdown one instance and try to access the same URL again.
- You should be able to access index.html again.

AMAZON AUTOSCALING - LAB

- Download Autoscaling and CloudWatch tools.
- Setup the tools by exporting environment variables.
- For autoscaling:
 - Create launch configuration
 - Create auto scaling group
 - Define the scale up and scale down policies.
- Using CloudWatch tools:
 - Create an alarm to call scale up policy
 - Create an alarm to call scale down policy

AMAZON STORAGE - LAB

- Navigate to S3 service.
- Create an S3 bucket.
- Upload data to S3 bucket using AWS console.
- Install s3cmd utility on EC2 instance.
- Upload data to S3 bucket using s3cmd.
- Enable static website monitoring option.
- Try to access bucket content using S3 URL.
- Edit bucket life cycle so that data older than 60 days gets archived to Amazon Glacier.

AMAZON RDS- LAB

- Navigate to RDS Service.
- Create a MySQL Database.
- Specify instance size and credentials.
- Note down the DB end point.
- Try to access database using any MySQL client.

AMAZON CLOUDFRONT- LAB

- Navigate to CloudFront service.
- Create a new distribution of type 'download'.
- Select Amazon S3 bucket as an origin.
- Keep all values default.
- Create the distribution.
- Note down the dynamically generated CloudFront URL.
- Try to access S3 bucket objects using CloudFront URL.

AMAZON CLOUDWATCH- LAB

- Navigate to CloudWatch service.
- Create a new alarm with following parameters.
 - Select statistics for 5 minutes average.
 - Select CPU Utilization metric for an EC2 instance.
 - If alarm triggers, an email notification should be sent out.
- Try to generate some load on server so that alarm triggers.
- Check the status of alarm. If it turns RED, then you should get an email notification.

AMAZON IAM- LAB

- Navigate to IAM Service.
- Create a new group called 'developers'.
- Create a new user 'developer1' in the group of developers.
- Assign Read Only policy to 'developer1'
- Note down the IAM URL to login via console.
- Login with user 'developer1' .
- Try to create S3 bucket.
- You should get an error while trying to create bucket.

AMAZON VPC - LAB

- Navigate to VPC Service.
- Create a VPC with public subnet option only.
- Launch a new EC2 instance in VPC.
- Review following parameters:
 - Internet gateway
 - ACL
 - Routing Table

AMAZON ROUTE53- LAB

- Navigate to Route53 Service.
- Create a hosted zone file for your domain.
- Create one A record and for domain 'www1.yourdomain.com' and point it to the IP address of EC2 instance.
- Configure ELB in US East and Singapore region.
- Create a Failover record for both ELBs created in above step.

AMAZON SES- LAB

- Navigate to SES service.
- Create SMTP credentials.
- Verify sender email address.
- Try to send out a test email using SMTP details provided by Amazon

AMAZON SNS - LAB

- Navigate to SNS service.
- Create a new topic called 'mytopic'.
- Create a new subscriber which will use Email protocol.
- Confirm the subscription.
- Try to send a test message.

AMAZON ARCHITECTURE - LAB

Assignment background and details:

An enterprise is losing market share to innovative start-ups that offer new customer centric digital services. The company's board has realized that a big transformation of the organization is required to regain the competitiveness in the marketplace. They have hired a new CIO from a successful gaming company. The first decision of the new CIO has been to enforce a cloud-first strategy and embrace a DevOps culture.

You have been assigned with multiple tasks based on the new CIO formulated strategy. In the first phase the task is to migrate a pilot application from on-premises data center to a public cloud platform: AWS / Azure / Google. The current 3- tier application architecture is illustrated below. There are similar test and production environments of the application. To keep costs down the test environment has been built with smaller hosts than production.

In the first phase the customer would like to move to the cloud enabling them to gain the following benefits:

- Provide multiple test environments for developers and realistic performance testing setup while keeping costs in control at the same time.
- Ability to test and deploy changes multiple times per day, even during the peak hours, with solid roll back plan if something goes wrong.
- Ability to withstand high peak loads. Typical peak load can be 10 -20x of normal load and lasts typically, 1 2h. During the peak hours, users access only a small portion of the site.

In the second phase the customer would like to benefit from rapid innovation speed of global cloud platform

vendors and gain business velocity with new customer centric digital services:

- Recommend approaches which allow for experimentation and rapid development.
- Provide best practices for building new cloud native applications that can start small and scale out with successful business.
- Avoid cloud platform vendor lock-in, portability/extension of the application environment

between/to different cloud platforms should be possible.

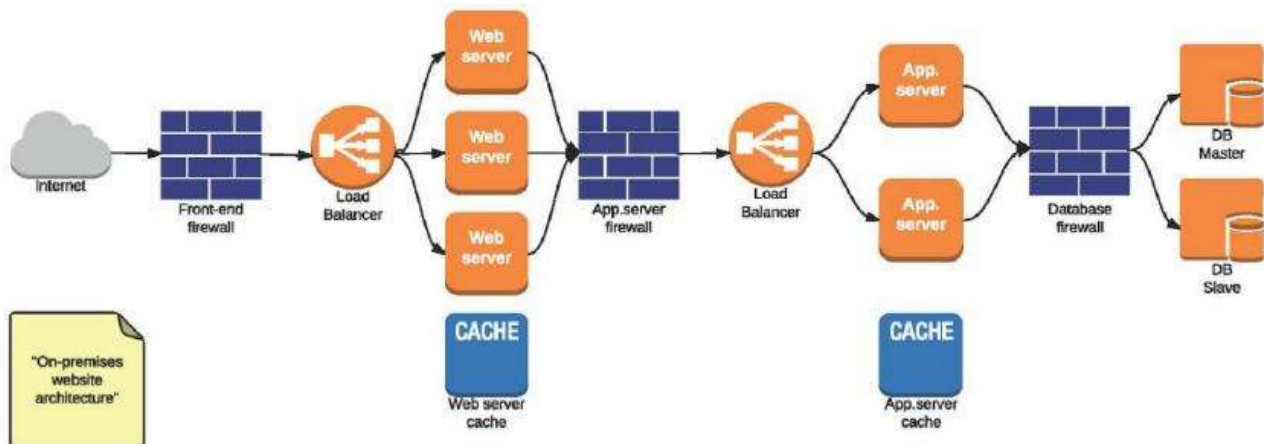
- Suggest tools that enable sharing of acquired knowledge in all steps of application lifecycle: development, deployment and operations.
- Provide generic guidance for refactoring existing business applications to allow them to take advantage of cloud technologies.

As the time is limited, you are not expected to provide very detailed plan. If needed, you can make assumption on technologies used for different tools and services, however ensure you document them.

Suggestions for preparing your solution:

- Please present your solution as a document that could be sent to the CIO of the company who has requested the information. Assume it will be used as the basis for a face to face meeting with the CIO.
- Prioritize the first phase and if you have time provide ideas also for the second phase.
- Ensure you include the name of the Cloud Platform & tooling that you have suggested and be prepared to justify your choices.
- Please ensure that your solution highlights best practices.

Customer's existing pilot application environment in an on-premises data center:





Sample Resume

Technical Manager – Cloud

Operations

- Direct daily operations of department, analyzing workflow, establishing priorities, developing standards and setting deadlines
- Facilitate the evaluation and selection of software product standards, as well as the design of standard software configurations
- Consult with application or infrastructure development projects to fit systems or infrastructure to architecture, and identify when it is necessary to modify the solution architecture to accommodate project needs
- Consult with users, management, vendors, and technicians to assess computing needs and system requirements
- Work with proposals to assess project feasibility and requirements
- Control operational budget and expenditures
- Work closely with the program management office (PMO) to ensure alignment of plans with what is being delivered.

Project Team

- Work with department heads, managers, supervisors, vendors, and team members, to solicit cooperation and resolve problems
- Stay abreast of advances in technology
- Understand the entire business process and gather business requirement from the business users for AWS, IoT implementation
- Analyze the requirement | change requests received and set up meeting with the customers to discuss implementation details
- Provide the impact analysis and design documents for business requirements to business users
- Estimate the efforts and develop plan for each phase for customer approvals
- Review the approved project plans to plan and coordinate project activity
- Review Architecture Assessment at project initiation time to ensure proper alignment
- Coordinate solution architecture implementation and modification activities time to time
- Review all solution architecture design and analysis work from various business users
- Assign and review the work of systems analysts, programmers, and architects and guide them to improve the quality of work
- Review and approve all systems charts and programs prior to their successful implementation
- Prepare and review operational reports or project progress reports
- Manage backup, security and user help systems

- Purchase necessary IoT Devices for On Premises to communicate with cloud
- Provide users with technical support for computer problems
- Problem Management – resolve recurring incidents, perform break fixes and implement preventive action items
- Incident Management – log, prioritize and resolve incidents and track them against various SLAs
- Provide 24*7 technical support for production related issues and get resolved as per SLAs
- Provide value add to customers by tuning applications to reduce the run time and improve the performance of the applications and create necessary user supporting documents which allow faster means to resolve any production issue

Cloud Architect: Sample Resume 1

Project Title: AWSIoTConnect

Environment: AWS (EC2, VPC, ELB, S3, EBS, RDS, Route53, ELB, Cloud Watch, CloudFormation, AWS Auto Scaling, Lambda, Elastic Bean Stalk), IOT, MySQL, SQL, AWS CLI, Unix/Linux, Shell scripting, Jenkins, Chef, Tomcat.

Description: The goal of AWSIoTConnect Project is to create a distributed IOT Platform for the Digital IOT World for different IOT Verticals. With AWSIoTConnect we can derive the Value provided by IOT Architecture, once we have the important information extracted we can create an IOT Data Brokerage Model to sell the important data to a Third-Party analytics Vendor or a Public Cloud Provider who provides IaaS. Providing IoT support to the different market segment such as Manufacturing -including infrastructure, awareness & safety, Energy/Utility including Oil & Gas, Transportation - including transportation systems, vehicles, and Non-vehicular, Smart City Applications, Retail IOT, HealthCare IoT, Finance-UBI & BFSI- Blockchain based IoT.

Responsibilities:

- Configured Windows & Linux environments in both public and private domains.
- Integrated Amazon Cloud Watch with Amazon EC2 instances for monitoring the log files and track metrics.
- Proficient in AWS services like VPC, EC2, S3, ELB, Auto Scaling Groups(ASG), EBS, RDS, IAM, CloudFormation, Route 53, CloudWatch, CloudFront, CloudTrail, Snowball, SES.
- Used security groups, network ACL's, internet gateways and route tables to ensure a secure zone for organization in AWS public cloud.
- Created S3 buckets in the AWS environment to store files, sometimes which are required to serve static content for a web application.

- Used IAM for creating roles, users, groups and also implemented MFA to provide additional security to AWS account and its resources.
- Written cloud formation templates in json to create custom VPC, subnets, NAT to ensure successful deployment of web applications.
- Maintained the monitoring and alerting of production and corporate servers using Cloud Watch service.
- Configured AWS Identity Access Management (IAM) Group and users for improved login authentication.
- Created AWS S3 buckets, performed folder management in each bucket, managed cloud trail logs and objects within each bucket.
- Created EBS volumes for storing application files for use with EC2 instances whenever they are mounted to them.
- Experienced in creating RDS instances to serve data through servers for responding to requests.
- Created snapshots to take backups of the volumes and also images to store launch configurations of the EC2 instances.
- Managed automated backups and created own backup snapshots when needed.
- Work with IOT and streaming protocols such as MQTT, LWM2M, SQS, AMQP, Kafka
- Develop AWS IoT Web based Application and Web Services that enables devices to connect to AWS services and other devices
- Work with Device Gateway/Device Registry to enable secure device connections and streaming of data
- Work with Message Broker / Rules Engine to filter, transform and act upon device data with business rules
- Deliver messages to other AWS services

Cloud Architect: Sample Resume 2

Project Title: AWS Support & Maintenance

Environment: AWS (EC2, VPC, ELB, S3, EBS, RDS, Route53, ELB, Cloud Watch, CloudFormation, AWS Auto Scaling, Lambda, Elastic Bean Stalk), IOT, MySQL, SQL, AWS CLI, Unix/Linux, Shell scripting, Jenkins, Chef, Tomcat.

Responsibilities:

- Responsible for architecting, designing, implementing and supporting of cloud-based infrastructure and its solutions.
- Created Highly Available Environments using Auto-Scaling, Load Balancers to spin up/down the servers and was responsible to send notifications through SNS for every activity occurred in the cloud environment and automated all configurations using Ansible.
- Developed Cloud Formation scripts to build on demand EC2 instance formation.
- Possess good knowledge in creating and launching EC2 instances using AMI's of Linux, Ubuntu, RHEL, and Windows and wrote shell scripts to bootstrap instance.
- Implemented Amazon RDS multi-AZ for automatic failover and high availability at the database tier.
- Configured and scheduled the scripts to automate the module installation in the environment.
- Created AWS S3 buckets, performed folder management in each bucket, managed cloud trail logs and objects within each bucket.
- Configured S3 to host Static Web content.
- Managing Amazon Web Services (AWS) infrastructure with automation and orchestration tools such as Chef, Ansible.
- Experienced in creating multiple VPC's and public, private subnets as per requirement and distributed them as groups into various availability zones of the VPC.
- Created NAT gateways and instances to allow communication from the private instances to the internet through bastion hosts.
- Created and configured elastic load balancers and auto scaling groups to distribute the traffic and to have a cost efficient, fault tolerant and highly available environment.
- Implemented domain name service (DNS) through Route 53 to have highly available and scalable applications.
- Written Templates for AWS infrastructure as a code using Ansible to build staging and production environments.
- Maintained edge location to cache data with CDN using Cloud Front to deliver data with less latency. Scaled distributed in-memory cache environment in the cloud using Elastic cache.

Cloud Architect: Sample Resume 3

Project Title: AWS Support & Maintenance

Environment: AWS (EC2, VPC, ELB, S3, EBS, RDS, Route53, ELB, Cloud Watch, CloudFormation, AWS Auto Scaling, Lambda, Elastic Bean Stalk), IOT, MySQL, SQL, AWS CLI, Unix/Linux, Shell scripting, Jenkins, Chef, Tomcat.

Responsibilities:

- Designing and deploying scalable, highly available, and fault tolerant systems on AWS
- Selecting the appropriate AWS service based on data, compute, database, or security requirements
- Lift and shift of an existing on-premises application to AWS Ingress and egress of data to and from AWS
- Identifying appropriate use of AWS architectural best practices
- Estimating AWS costs and identifying cost control mechanisms.
- Selecting appropriate AWS services to design and deploy an application based on given requirements
- Migrating complex, multi-tier applications on AWS
- Designing and deploying enterprise-wide scalable operations on AWS
- Implementing cost control strategies
- Picking the right AWS services for the application.
- Leveraging AWS SDKs to interact with AWS services from the application.
- Optimizing performance of AWS services used by the application.
- Code-level application security (IAM roles, credentials, encryption, etc.)
- Actively monitor, research and analyze ways in which the services in AWS can be improved.
- Manage and configure AWS services as per the business needs
- Creating and managing AMI and snapshots
- Upgrade and downgrade of AWS resources (CPU, Memory, EBS)
- Creating AWS instances
- Monitoring servers thorough Amazon Cloud Watch, SNS
- Creating and managing the S3 buckets
- Configuring IAM roles and security

