CONTENTS

1.1 Introduction & Definition of Cloud Computing

Cloud computing is the delivery of different services through the Internet. These resources include tools and applications like data storage, servers, databases, networking, and software.

Rather than keeping files on a proprietary hard drive or local storage device, cloud-based storage makes it possible to save them to a remote database. As long as an electronic device has access to the web, it has access to the data and the software programs to run it.

Cloud computing is a popular option for people and businesses for a number of reasons including cost savings, increased productivity, speed and efficiency, performance, and security.

Cloud computing is named as such because the information being accessed is found remotely in the cloud or a virtual space. Companies that provide cloud services enable users to store files and applications on remote servers and then access all the data via the Internet. This means the user is not required to be in a specific place to gain access to it, allowing the user to work remotely.

Cloud computing takes all the heavy lifting involved in crunching and processing data away from the device you carry around or sit and work at. It also moves all of that work to huge computer clusters far away in cyberspace. The Internet becomes the cloud. Your data, work, and applications are available from any device with which you can connect to the Internet, anywhere in the world.

The U.S. National Institute of Standards and Technology (NIST) defines cloud computing as:

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

1.2 Characteristics of Cloud Computing

NIST identifies five essential characteristics of cloud computing

On-demand self service

Cloud computing resources can be provisioned on-demand by the users, without requiring interactions with the cloud service provider. The process of provisioning resources is automated.

Broad network access

Cloud computing resources can be accessed over the network using standard access mechanisms that provide platform-independent access through the use of heterogeneous client platforms such as workstations, laptops, tablets and smartphones.

Resource pooling

The computing and storage resources provided by cloud service providers are pooled to serve multiple users using multi-tenancy. Multi-tenant aspects of the cloud allow multiple users to be served by the same physical hardware.

Rapid elasticity

Cloud computing resources can be provisioned rapidly and elastically. Cloud resources can be rapidly scaled up or down based on demand.

Measured service

Cloud computing resources are provided to users on a pay-per-use model. The usage of the cloud resources is measured and the user is charged based on some specific metric.

In addition to these 5 essential characteristics of cloud computing, other characteristics that again highlight savings in cost include:

Performance

Cloud computing provides improved performance for applications since the resources available to the applications can be scaled up or down based on the dynamic application workloads.

Reduced costs

Cloud computing provides cost benefits for applications as only as much computing and storage resources as required can be provisioned dynamically, and upfront investment in purchase of computing assets to cover worst case requirements is avoided.

Outsourced Management

Cloud computing allows the users (individuals, large organizations, small and medium enterprises and governments) to outsource the IT infrastructure requirements to external cloud providers.

Reliability

Applications deployed in cloud computing environments generally have a higher reliability since the underlying IT infrastructure is professionally managed by the cloud service.

Multi-tenancy

The multi-tenanted approach of the cloud allows multiple users to make use of the same shared resources.

In virtual multi-tenancy, computing and storage resources are shared among multiple users.

In organic multi-tenancy every component in the system architecture is shared among multiple tenants

1.3 Cloud Service Models

Cloud computing is not a single piece of technology like a microchip or a cell phone. Rather, it's a system primarily comprised of three services: software-as-a-service (SaaS), infrastructure-as-a-service (IaaS), and platform-as-a-service (PaaS).

Software-as-a-service (SaaS)

Applications, management and user interfaces provided over a network. It involves the licensure of a software application to customers. Licenses are typically provided through a pay-as-you-go model or on-demand. This type of system can be found in Microsoft Office's 365.

Software/Interface

SaaS provides the users a complete software application or the user interface to the application itself.

Outsourced Management

The cloud service provider manages the underlying cloud infrastructure including servers, network, operating systems, storage and application software, and the user is unaware of the underlying architecture of the cloud.

Thin client interfaces

Applications are provided to the user through a thin client interface (e.g., a browser). SaaS applications are platform independent and can be accessed from various client devices such as workstations, laptop, tablets and smartphones, running different operating systems.

Ubiquitous Access

Since the cloud service provider manages both the application and data, the users are able to access the applications from anywhere.

Characteristics

Multi-tenancy
On-demand Software
Open Integration Protocols
Social Network Integration

Adoption

Individual Users: High
Small /Medium Enterprises: High
Large Organizations: High
Government: Medium

Benefits

Lower costs
No infrastructure required
Seamless upgrades
Guaranteed performance
Automated Backups
Easy Data Recovery
Secure
High adoption
On-the-move access

Examples

Google Apps
Salesforce.com
Facebook
Zoho
Dropbox
Taleo
Microsoft office 365
Linkedin
Slideshare
CareCloud

Platform-as-a-service (PaaS)

Application development frameworks, operating systems and deployment frameworks are outsourced. It is considered the most complex of the three layers of cloud-based computing. PaaS shares some similarities with SaaS, the primary difference being that instead of delivering software online, it is actually a platform for creating software that is delivered via the Internet. This model includes platforms like Force.com and Heroku

Development & Deployment

PaaS provides the users the capability to develop and deploy application in the cloud using the development tools, application programming interfaces (APIs), software libraries and services provided by the cloud service provider.

Provider Manages Infrastructure

The cloud service provider manages the underlying cloud infrastructure including servers, network, operating systems and storage.

User Manages Application

The users, themselves, are responsible for developing, deploying, configuring and managing applications on the cloud infrastructure.

Characteristics

Multi-tenancy
Open Integration Protocols
App Development Tools and SDKs
Analytics

Adoption

Individual Users: Low
Small /Medium Enterprises: Medium
Large Organizations: High
Government: Medium

Benefits

Lower upfront and operational costs
No Infrastructure Management Costs
Improved Scalability
Higher performance
Secure Access
Quick and easy development
Seamless Integration

Examples

Google App Engine
Windows Azure Platform
Force.com
RightScale
Heorku
Github
Gigaspaces
AppScale
OpenStack
LongJump

Infrastructure-as-a-service (IaaS)

Virtual computing, Storage and Network resource can be provisioned on demand. It involves a method for delivering everything from operating systems to servers and storage through IP-based connectivity as part of an on-demand service. Clients can avoid the need to purchase software or servers, and instead procure these resources in an outsourced, on-demand service. Popular examples of the IaaS system include IBM Cloud and Microsoft Azure.

Resource Provisioning

Provides the users the capability to provision computing and storage resources.

Virtual Machines

These resources are provided to the users as virtual machine instances and virtual storage. Users can start, stop, configure and manage the virtual machine instances and virtual storage.

Provider Managers Infrastructure

The cloud service provider manages the underlying infrastructure.

Pay-per-use/Pay-as-you-go

Virtual resources provisioned by the users are billed based on a pay-per-use/pay-as you-go paradigm.

Characteristics

Multi-tenancy
Virtualized Hardware
Management and Monitoring Tools
Disaster Recovery

Adoption

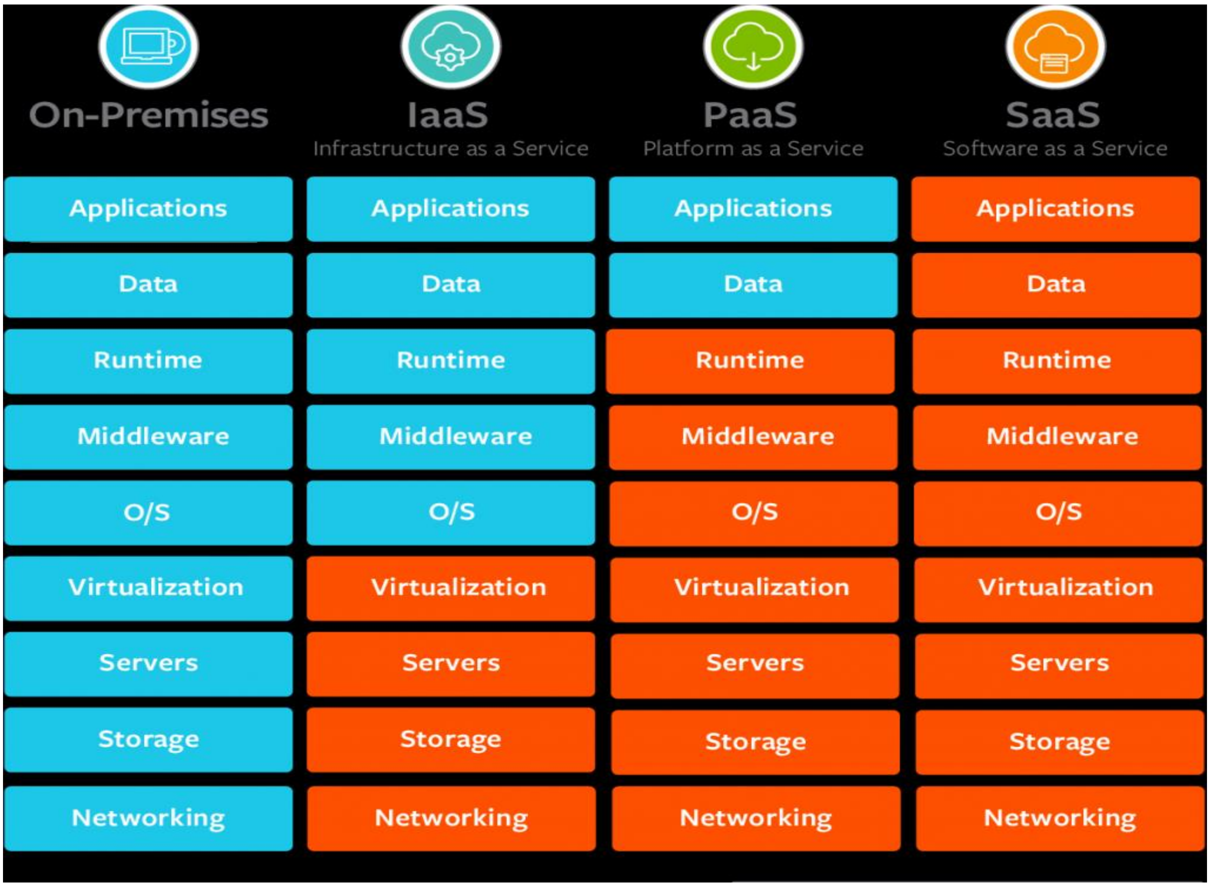
Individual Users: Low
Small /Medium Enterprises: Medium
Large Organizations: High
Government: High

Benefits

Less need for IT management activities
No Infrastructure Management Costs
Pay-per-use
Guaranteed performance
Dynamic scaling
Secure Access
Enterprise Grade infrastructure
Green IT adoption

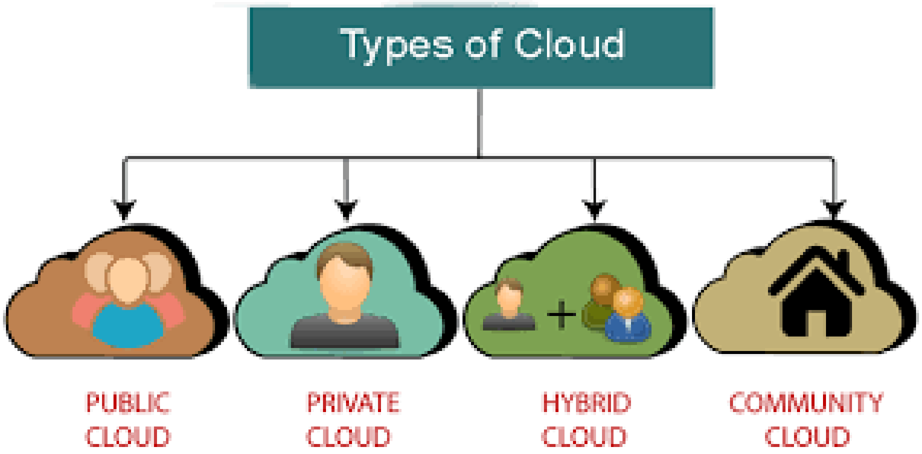
Examples

Amazon EC2
Google Compute
Rackspace
GoGrid
Eucalyptus
Joyent
Terremark
OpSource
Savvis
Nimbula
Enamoly



On-Premises	IaaS Infrastructure as a Service	PaaS Platform as a Service	SaaS Software as a Service
Applications	Applications	Applications	Applications
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware
O/S	O/S	O/S	O/S
Virtualization	Virtualization	Virtualization	Virtualization
Servers	Servers	Servers	Servers
Storage	Storage	Storage	Storage
Networking	Networking	Networking	Networking

1.4 Cloud Deployment Models



Public Cloud

Public clouds are provided by a designated service provider for general public under a utility based pay-per-use consumption model. The cloud resources are hosted generally on the service provider‘s premises. Popular examples of public clouds are Amazon‘s AWS (EC2, S3 etc.), Rackspace Cloud Suite, and Microsoft‘s Azure Service Platform.

Private Cloud

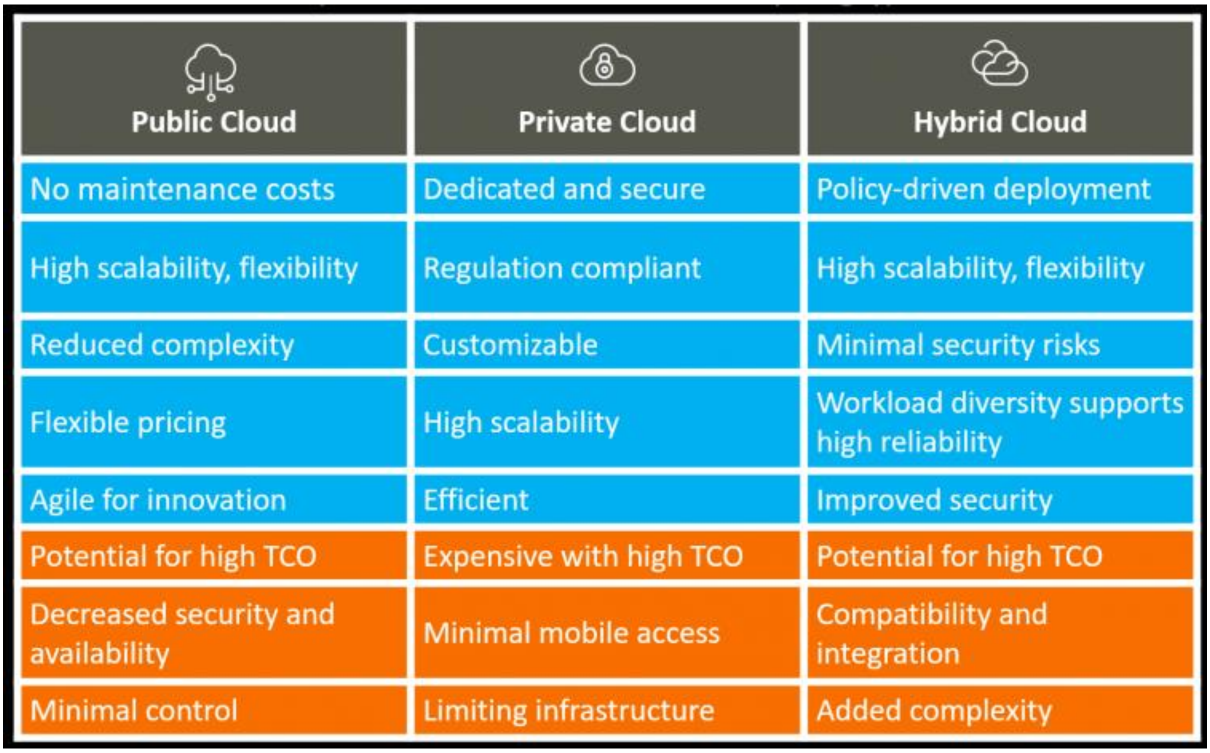
Private clouds are built, operated, and managed by an organization for its internal use only to support its business operations exclusively. Public,private, and government organizations worldwide are adopting this model to exploit the cloud benefits like flexibility, cost reduction, agility and so on..

Community Cloud

Community clouds are shared by several organizations and support a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). They may be managed by the organizations or a third party and may exist on premise or off premise. One example of this is OpenCirrus formed by HP,Intel, Yahoo, and others.

Hybrid Cloud

Hybrid clouds are a composition of two or more clouds (private, community,or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application.



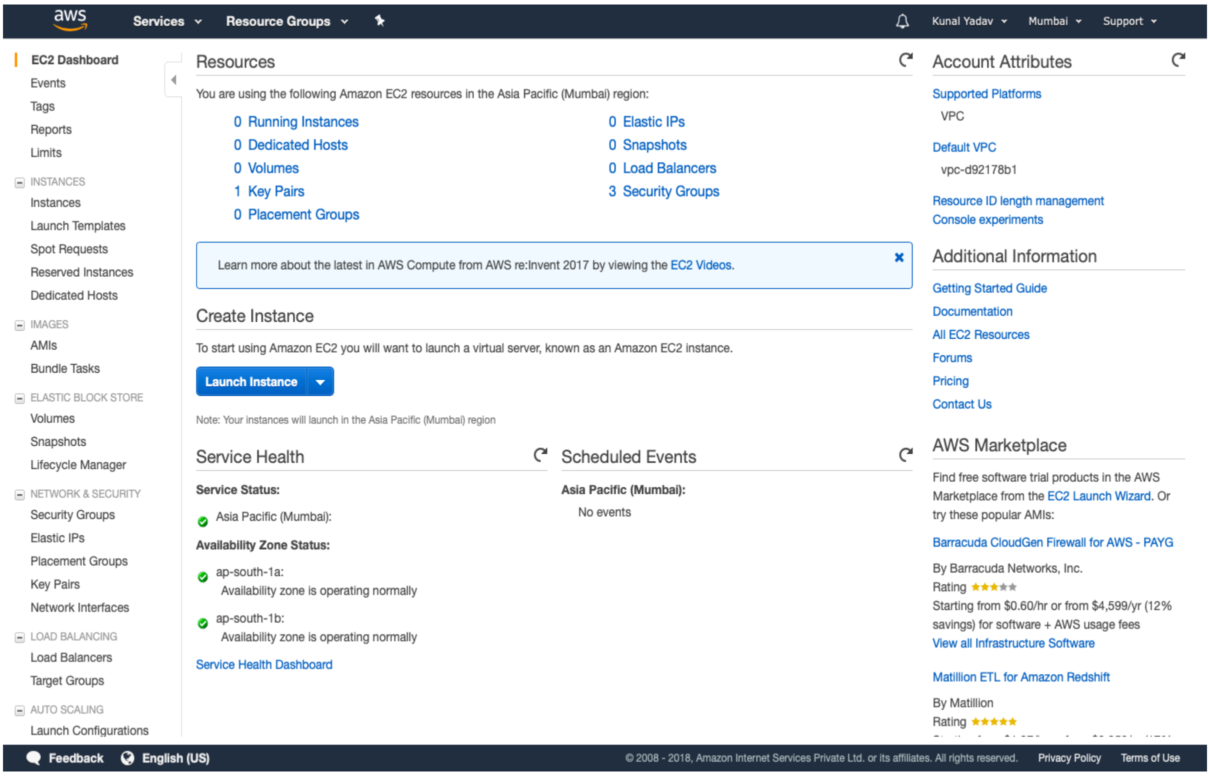
Public Cloud	Private Cloud	Hybrid Cloud
No maintenance costs	Dedicated and secure	Policy-driven deployment
High scalability, flexibility	Regulation compliant	High scalability, flexibility
Reduced complexity	Customizable	Minimal security risks
Flexible pricing	High scalability	Workload diversity supports high reliability
Agile for innovation	Efficient	Improved security
Potential for high TCO	Expensive with high TCO	Potential for high TCO
Decreased security and availability	Minimal mobile access	Compatibility and integration
Minimal control	Limiting infrastructure	Added complexity

1.5 Cloud Services Examples and Applications

Cloud Services Examples

IaaS: Amazon EC2, Google Compute Engine, Windows Azure VMs

Amazon EC2 is an IaaS offers from Amazon.com. Ec2 is a web service that provides computing capacity in the form of virtual machines that are launched in Amazon's cloud computing environment. EC2 allows users to launch instances on demand using a simple web-based interface. Amazon provides a pre-configured Amazon's Machine Images(AMI's) which are templates of cloud instances. EC2 allows one to launch machines with variety of operating systems. Users can load their applications on running instances and rapidly and easily increase or decrease the capacity to meet the dynamic application performance requirements. It is pay-per-use model.



Resources

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

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

Learn more about the latest in AWS Compute from AWS re:Invent 2017 by viewing the EC2 Videos.

Create Instance

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

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

Service Health

Service Status:

Asia Pacific (Mumbai):

Availability Zone Status:

ap-south-1a:
Availability zone is operating normally

ap-south-1b:
Availability zone is operating normally

Service Health Dashboard

Scheduled Events

Asia Pacific (Mumbai):
No events

Account Attributes

Supported Platforms
VPC

Default VPC
vpc-d92178b1

Resource ID length management
Console experiments

Additional Information

Getting Started Guide
Documentation
All EC2 Resources
Forums
Pricing
Contact Us

AWS Marketplace

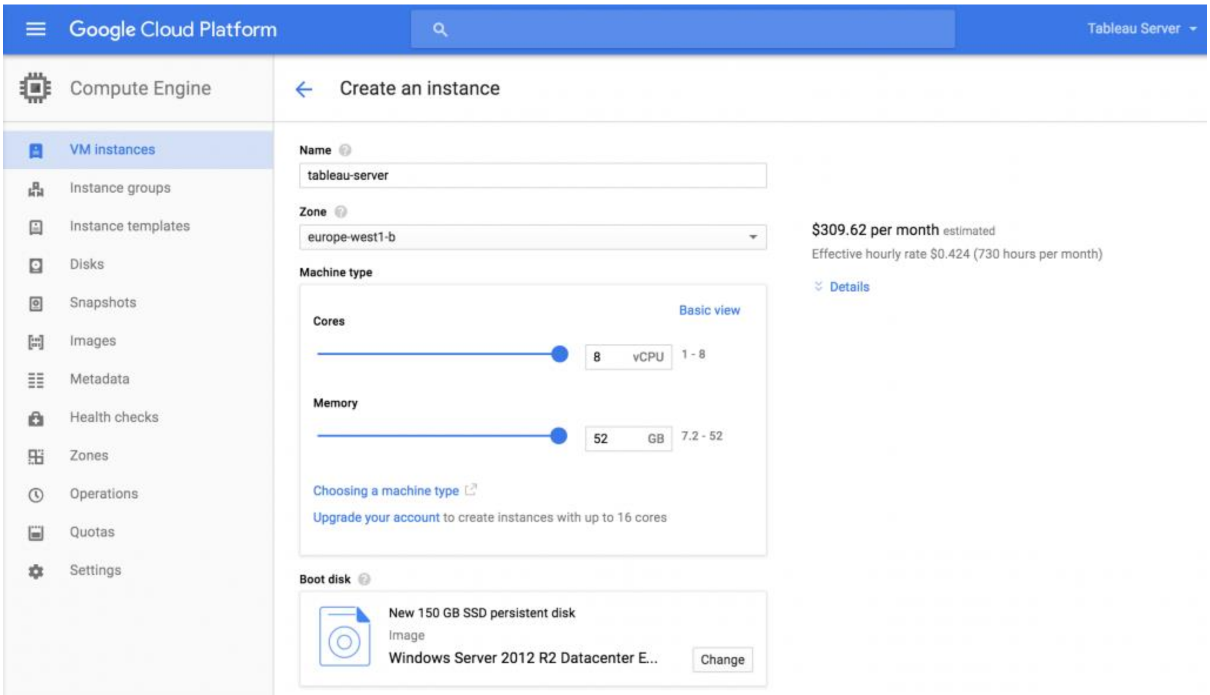
Find free software trial products in the AWS Marketplace from the EC2 Launch Wizard. Or try these popular AMIs:

Barracuda CloudGen Firewall for AWS - PAYG
By Barracuda Networks, Inc.
Rating ★★★★★
Starting from $0.60/hr or from $4,599/yr (12% savings) for software + AWS usage fees
View all Infrastructure Software

Matillion ETL for Amazon Redshift
By Matillion
Rating ★★★★★

© 2008 - 2018, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

Google compute engine is an IaaS offering from Google. GCE provides virtual machines of various computing capacities ranging from small instances to high memory machine types.



Create an instance

Name
tableau-server

Zone
europe-west1-b

Machine type

Cores
8 vCPU 1 - 8

Memory
52 GB 7.2 - 52

Choosing a machine type

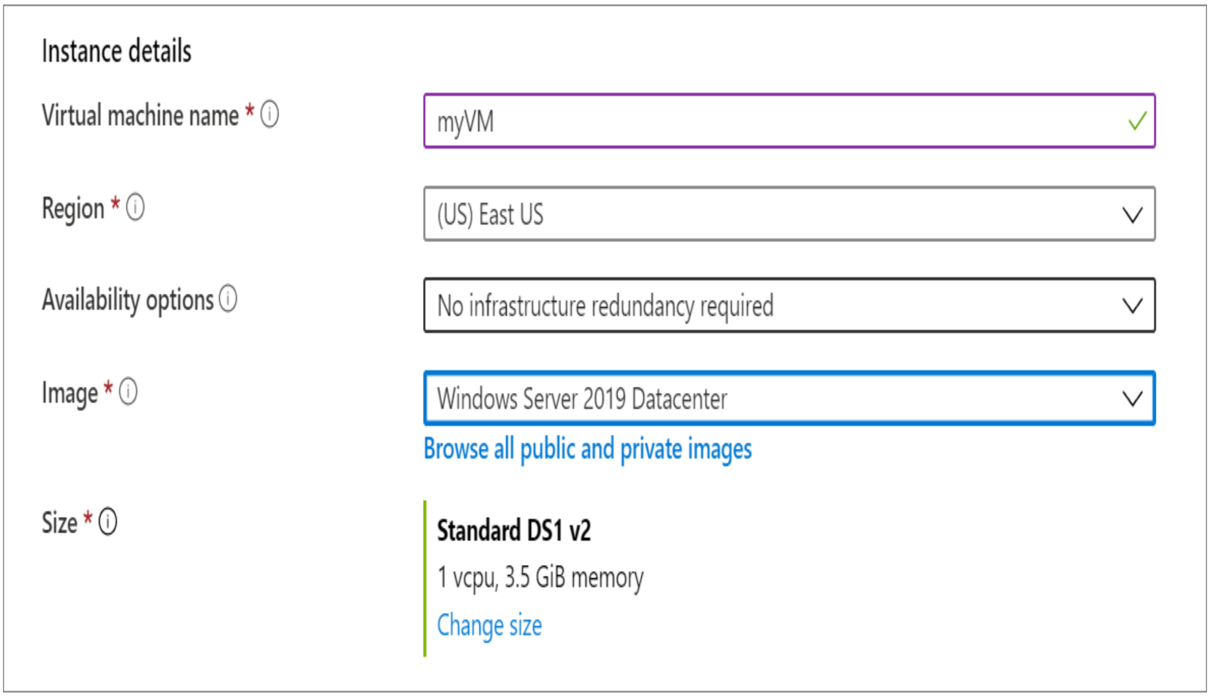
Upgrade your account to create instances with up to 16 cores

Boot disk

New 150 GB SSD persistent disk
Image
Windows Server 2012 R2 Datacenter E...

$309.62 per month estimated
Effective hourly rate $0.424 (730 hours per month)
Details

Windows azure VM is IaaS offering from Microsoft. Azure VMs provides virtual machines of various computing capacities ranging from small instances to memory intensive machine types.



Instance details

Virtual machine name * ⓘ: myVM ✓

Region * ⓘ: (US) East US

Availability options ⓘ: No infrastructure redundancy required

Image * ⓘ: Windows Server 2019 Datacenter

Size * ⓘ: Standard DS1 v2

1 vcpu, 3.5 GiB memory

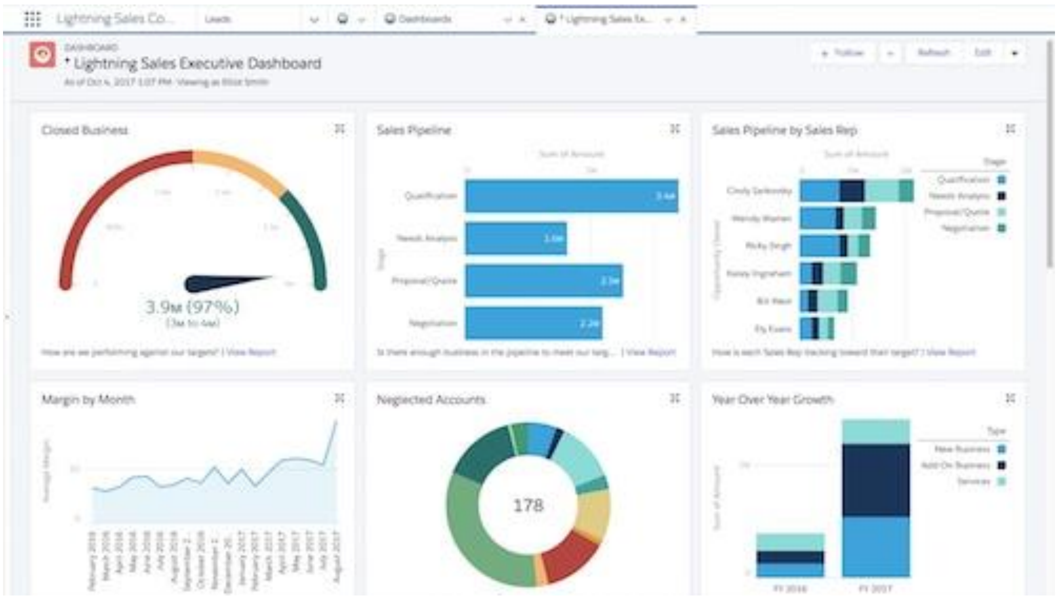
PaaS: Google App Engine

Google App Engine (often referred to as GAE or simply App Engine) is a Platform as a Service and cloud computing platform for developing and hosting web applications in Google-managed data centers. Applications are sandboxed and run across multiple servers. App Engine offers automatic scaling for web applications—as the number of requests increases for an application, App Engine automatically allocates more resources for the web application to handle the additional demand.

Google App Engine primarily supports Go, PHP, Java, Python, Node.js, .NET, and Ruby applications, although it can also support other languages via "custom runtimes". The service is free up to a certain level of consumed resources and only in standard environment but not in flexible environment. Fees are charged for additional storage, bandwidth, or instance hours required by the application.

SaaS: Salesforce

The Salesforce Sales Cloud is an on-demand CRM platform offering applications for small, midsize, and enterprise organizations, with a focus on sales and customer relationships. This Salesforce platform has CRM capabilities that include lead management, marketing automation, and contact management. These applications help organizations to manage customer accounts, track sales leads, conduct and monitor marketing campaigns and provide service post-sale. Salesforce Lightning Experience is upgraded CRM version of Sales Cloud that streamlines the sales process by allowing any sales rep to generate and follow-up on leads, manage account activity, track the status of each campaign, and create reports without hassle.



Equivalent Services from Leading Market Players

	AWS	Google	Azure
Compute	EC2	Compute Engine	VM
Storage	S3, EBS	Cloud Storage	Blob Storage
Networking	VPC	Google VPC	Virtual Network
Databases	RDS, Dynamo DB	Cloud SQL, Firestore, Bigtable	SQL, MySQL, Cosmos DB
PaaS	Beanstalk	App Engine	App Service
SaaS	Workdocs	Google Docs	Office 365

Machine Learning	Sagemaker	Cloud ML Engine	Machine Learning
IoT	IoT core	Cloud IoT Core	IoT Hub

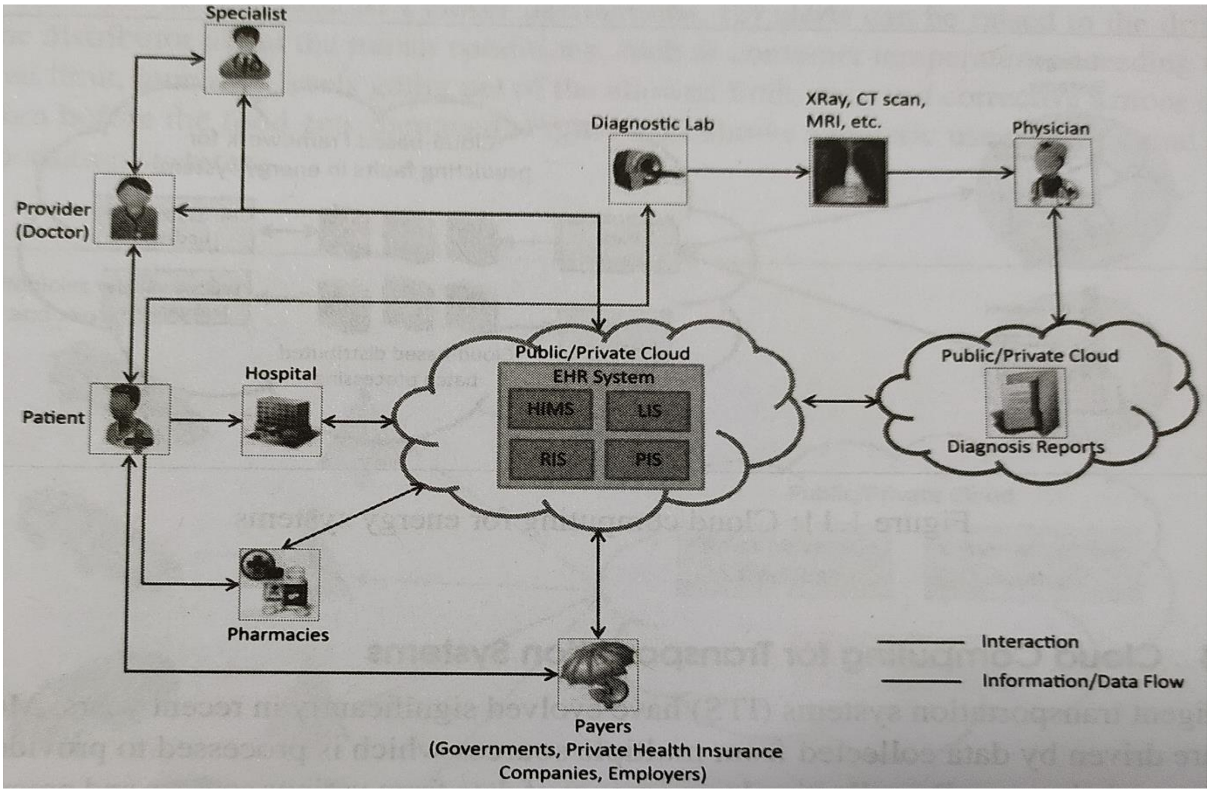
Cloud-based Services & Applications

Some of the cloud applications are:

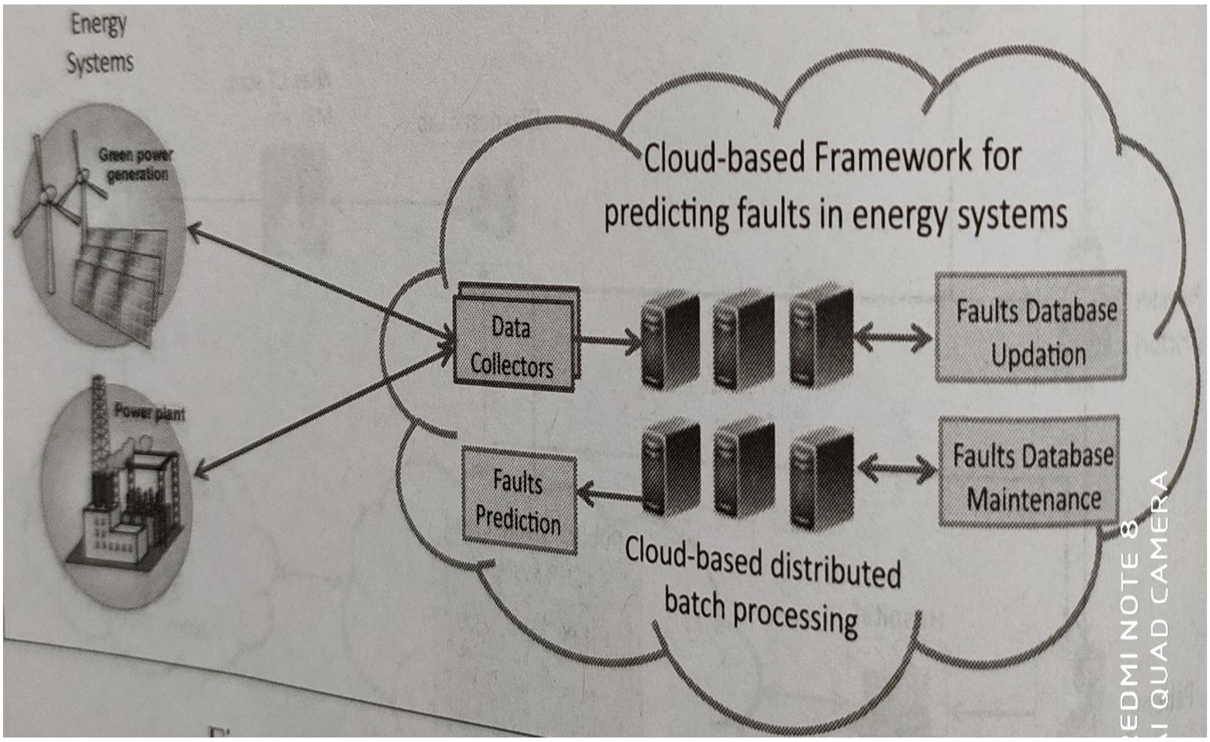
- Banking & Financial Apps
- E-Commerce Apps
- Social Networking
- Healthcare Systems
- Energy Systems
- Intelligent Transportation Systems
- E-Governance
- Education
- Mobile Communications

Let us see in detail of the some of the use-cases for cloud -services for few examples

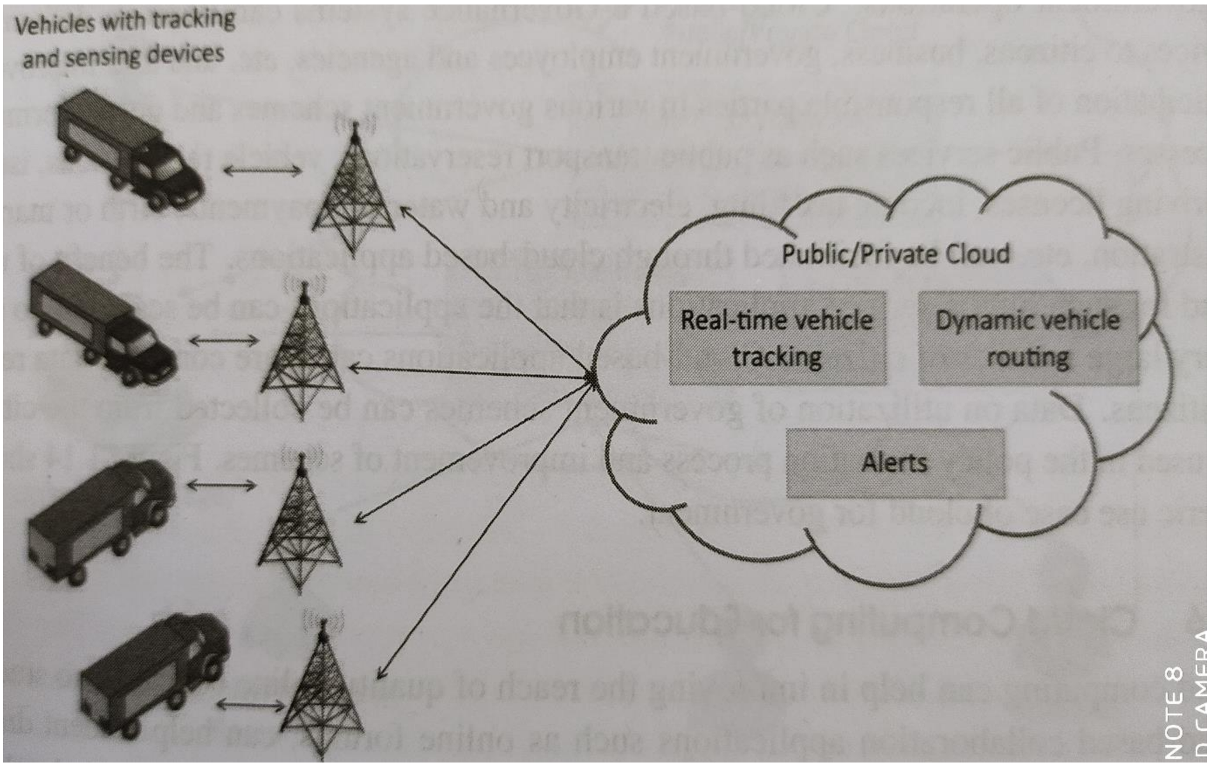
Cloud Computing for Healthcare



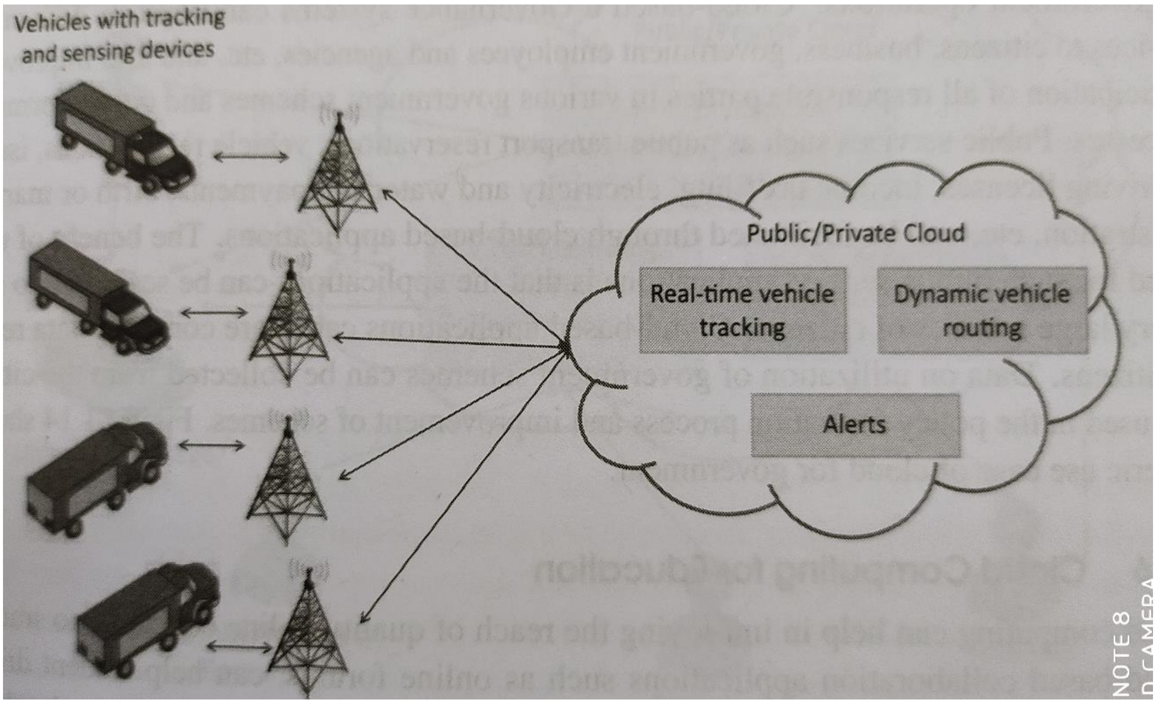
Cloud Computing in Energy Systems



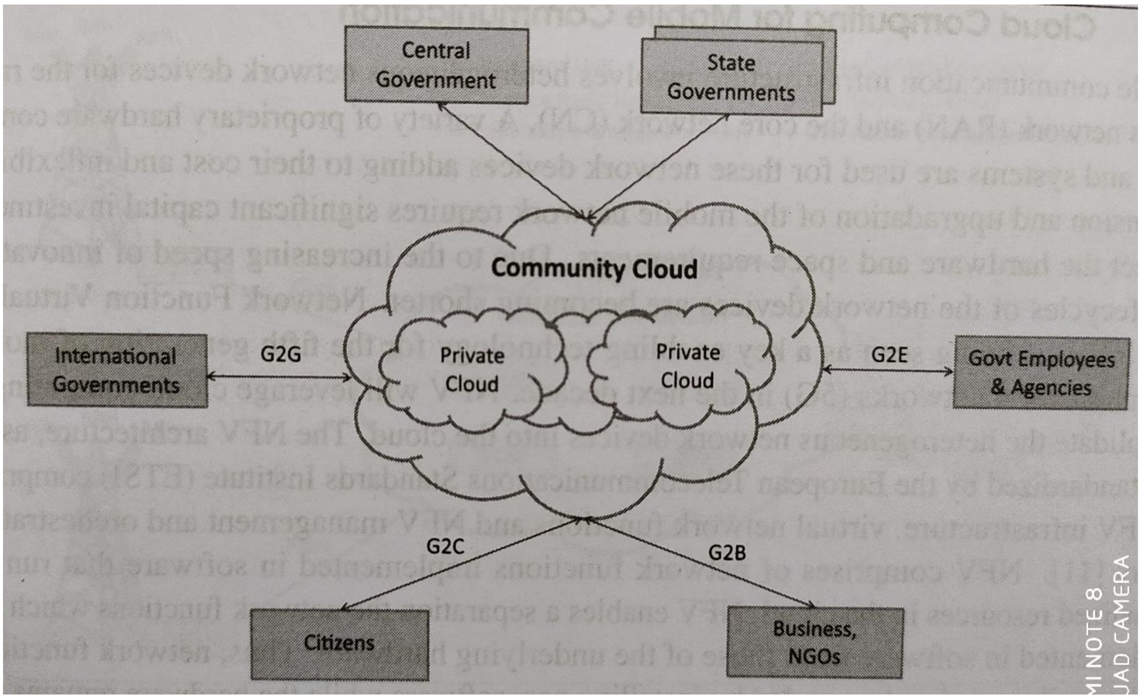
Cloud Computing in Transportation Systems


Vehicles with tracking and sensing devices
Public/Private Cloud
Real-time vehicle tracking
Dynamic vehicle routing
Alerts

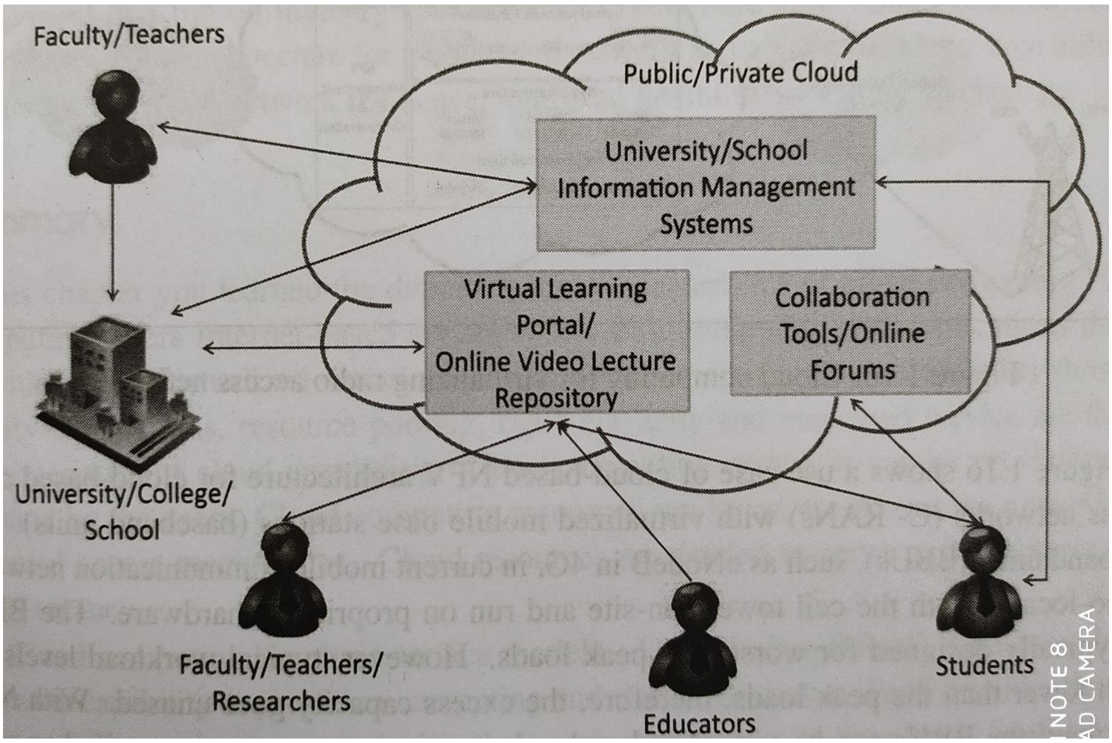
Cloud Computing in Manufacturing Industry


Vehicles with tracking and sensing devices
Public/Private Cloud
Real-time vehicle tracking
Dynamic vehicle routing
Alerts

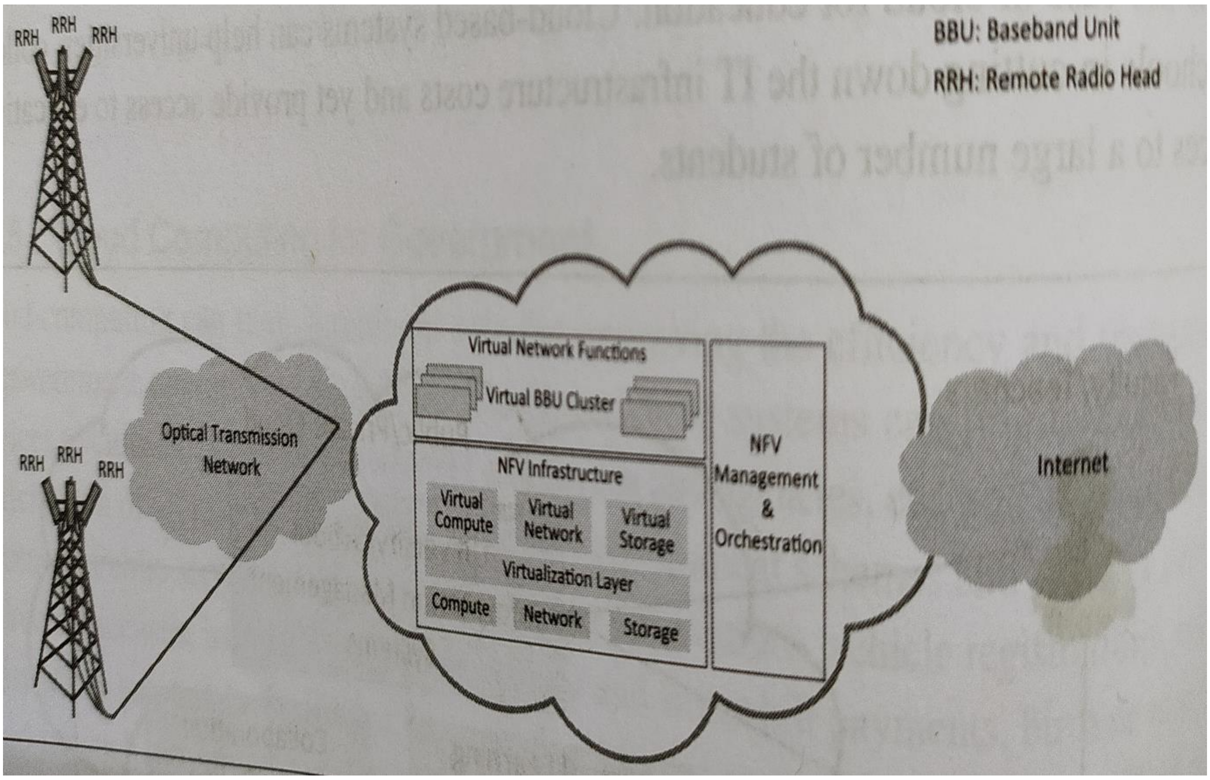
Cloud Computing for Government



Cloud Computing for Education


Faculty/Teachers
Public/Private Cloud
University/School Information Management Systems
Virtual Learning Portal/ Online Video Lecture Repository
Collaboration Tools/Online Forums
University/College/ School
Faculty/Teachers/ Researchers
Educators
Students
NOTE 8
QUAD CAMERA

Cloud computing for Mobile Communication



1.6 Cloud Computing Architecture

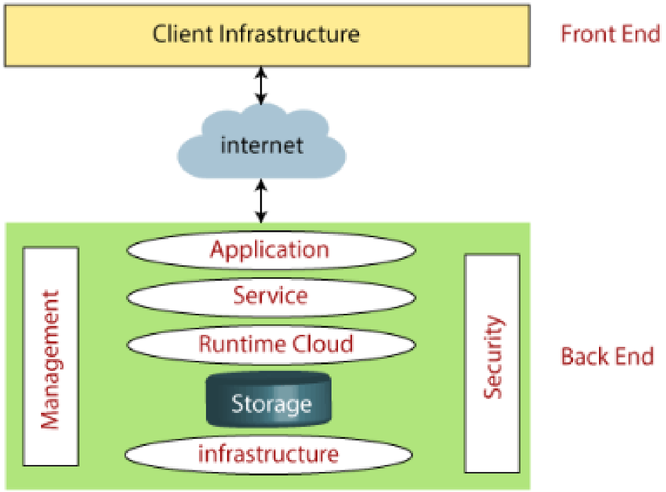
Cloud Computing technology is used by both small and large organizations to store the information in cloud and **access** it from anywhere at any time using the internet connection.

Cloud computing architecture is a combination of **service-oriented architecture** and **event-driven architecture**.

Cloud computing architecture is divided into two parts:

- Front End
- Back End

The below diagram shows the architecture of cloud computing -



Front End

The front end is used by the client. It contains client-side interfaces and applications that are required to access the cloud computing platforms. The front end includes web servers (including Chrome, Firefox, internet explorer, etc.), thin & fat clients, tablets, and mobile devices.

Back End

The back end is used by the service provider. It manages all the resources that are required to provide cloud computing services. It includes a huge amount of data storage, security mechanism, virtual machines, deploying models, servers, traffic control mechanisms, etc.

Components of Cloud Computing Architecture

There are the following components of cloud computing architecture -

1. Client Infrastructure

Client Infrastructure is a Front end component. It provides GUI (Graphical User Interface) to interact with the cloud.

2. Application

The application may be any software or platform that a client wants to access.

3. Service

A Cloud Services manages that which type of service you access according to the client's requirement.

Cloud computing offers the following three type of services:

i. Software as a Service (SaaS) – It is also known as **cloud application services**. Mostly, SaaS applications run directly through the web browser means we do not require to download and install these applications. Some important example of SaaS is given below –

Example: Google Apps, Salesforce Dropbox, Slack, Hubspot, Cisco WebEx.

ii. Platform as a Service (PaaS) – It is also known as **cloud platform services**. It is quite similar to SaaS, but the difference is that PaaS provides a platform for software creation, but using SaaS, we can access software over the internet without the need of any platform.

Example: Windows Azure, Force.com, Magento Commerce Cloud, OpenShift.

iii. Infrastructure as a Service (IaaS) – It is also known as **cloud infrastructure services**. It is responsible for managing applications data, middleware, and runtime environments.

Example: Amazon Web Services (AWS) EC2, Google Compute Engine (GCE), Cisco Metapod.

4. Runtime Cloud

Runtime Cloud provides the **execution and runtime environment** to the virtual machines.

5. Storage

Storage is one of the most important components of cloud computing. It provides a huge amount of storage capacity in the cloud to store and manage data.

6. Infrastructure

It provides services on the **host level**, **application level**, and **network level**. Cloud infrastructure includes hardware and software components such as servers, storage, network devices, virtualization software, and other storage resources that are needed to support the cloud computing model.

7. Management

Management is used to manage components such as application, service, runtime cloud, storage, infrastructure, and other security issues in the backend and establish coordination between them.

8. Security

Security is an in-built back end component of cloud computing. It implements a security mechanism in the back end.

9. Internet

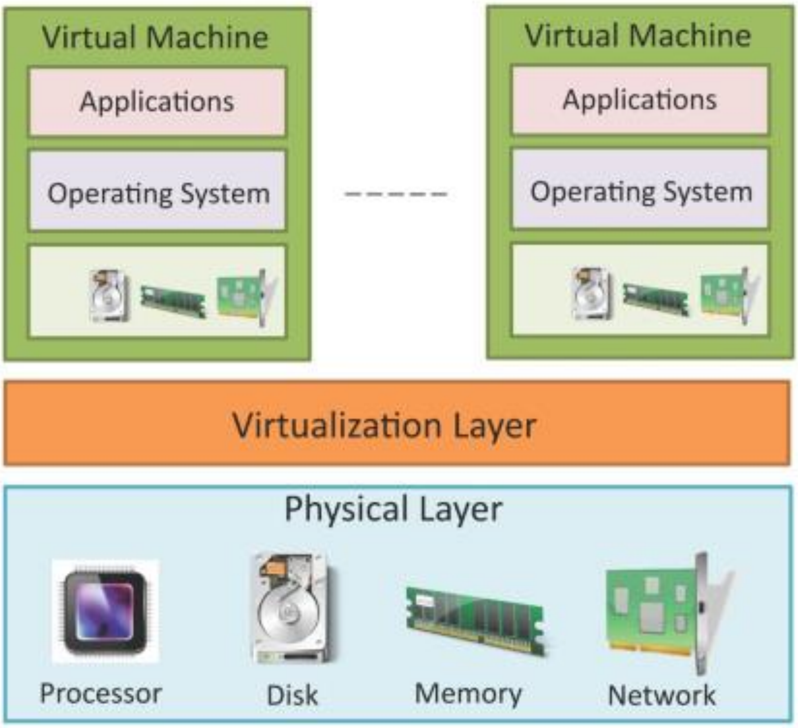
The Internet is medium through which front end and back end can interact and communicate with each other.

1.7 Cloud Concepts and Technologies

Virtualization

Virtualization is the "creation of a virtual (rather than actual) version of something, such as a server, a desktop, a storage device, an operating system or network resources".

In other words, Virtualization is a technique, which allows to share a single physical instance of a resource or an application among multiple customers and organizations. It does by assigning a logical name to a physical storage and providing a pointer to that physical resource when demanded. It allows multiple od to run on the same underlying resource.



Hypervisor

Hypervisor is a form of virtualization software used in Cloud hosting to divide and allocate the resources on various pieces of hardware. The program which provide partitioning, isolation or abstraction is called virtualization hypervisor. Hypervisor is a hardware virtualization technique that allows multiple guest operating systems (OS) to run on a single host system at the same time. A hypervisor is sometimes also called a virtual machine manager(VMM).

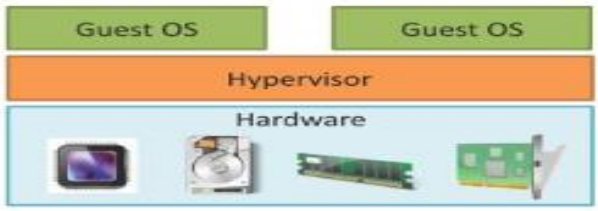
Types of Hypervisor

TYPE-1 Hypervisor

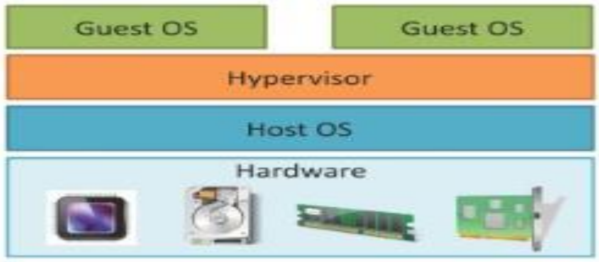
Hypervisor runs directly on underlying host system. It is also known as “Native Hypervisor” or “Bare metal hypervisor”. It does not require any base server operating system. It has direct access to hardware resources. Examples of Type 1 hypervisors include VMware ESXi, Citrix XenServer and Microsoft Hyper-V hypervisor.

TYPE-2 Hypervisor

A Host operating system runs on underlying host system. It is also known as ‘Hosted Hypervisor”. Basically a software installed on an operating system. Hypervisor asks operating system to make hardware calls. Example of Type 2 hypervisor include VMware Player or Parallels Desktop. Hosted hypervisors are often found on endpoints like PCs.



Type-1 Hypervisor



Type-2 Hypervisor

Examples

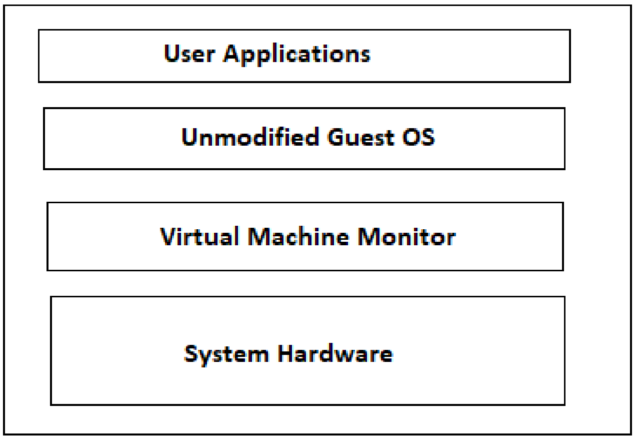
Load Balancer	**Type**
NginX	Software
HA Proxy	Software
Pound	Software
Varish	Software
Cisco Systems Catalyst 6500	Hardware
Coyote Point Equalizer	Hardware
F5 Networks BIG-IP LTM	Hardware
Barracuda Load Balancer	Hardware

Types of Virtualization

- **Full Virtualization**
- **Para Virtualization**
- **Hardware Virtualization**

Full Virtualization

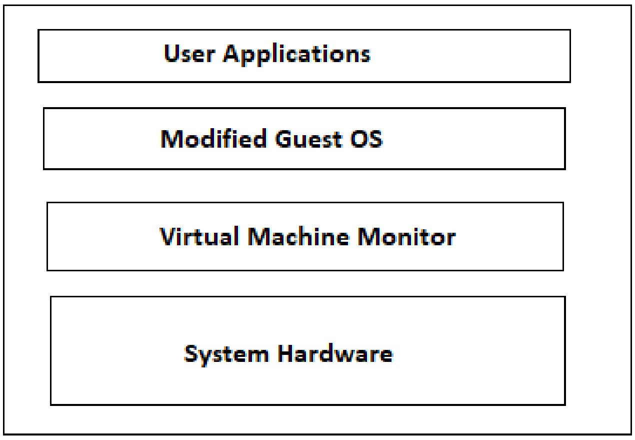
- Decouples the guest OS from underlying hardware.
- Guest OS doesn’t require any modification and doesn’t aware that it is being virtualized
- Direct execution of user requests and binary translation of OS requests



(a) Full Virtualization

Para Virtualization

- Guest OS kernel is modified to enable communication with the hypervisor to improve performance and efficiency.
- Hyper calls communicate with the virtualization layer hypervisor



(b) Para Virtualization

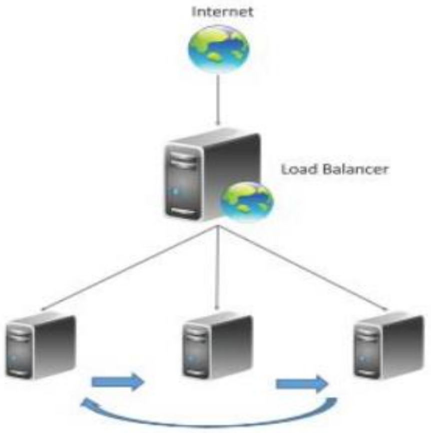
Hardware Virtualization

- Hardware assisted virtualization is enabled by hardware features such as Intel's Virtualization Technology (VT-x) and AMD's AMD-V
- privileged and sensitive calls are set to automatically trap to the hypervisor
- No need for Binary translations or para virtualizations

Load Balancing

Cloud Computing resources can be scaled up on-demand. Load balancing distributes workloads across multiple servers to meet the application workloads. The goals of Load balancing are

- Achieve maximum utilization of resources
- Minimizing the response times
- Maximizing throughput



This can be achieved by implementing load balancing algorithms. There are variety of load balancing algorithms. Let us have a look on different load balancing algorithms.

Round Robin Load Balancing

Servers are selected one by one to serve incoming requests in a Non-hierarchical circular fashion with no priority.

Weighted Round Robin Load Balancing

Servers are assigned some weights. The Incoming requests are proportionally routed using static or dynamic ratio of respective weights.

Low Latency Load Balancing

Load balancer monitors the latency of each server. Each Incoming request is routed to low latency server.

Least Connections Load Balancing

In Least connections load balancing, the incoming requests are routed to server with least number of connections.

Priority Load Balancing

Each server is assigned a priority. The incoming requests are routed to server with highest priority as long as the server is available. When the highest priority server fails, then incoming traffic is routed to next priority server.

Overflow Load Balancing

Overflow Load balancing is similar to priority load balancing, when the higher priority server overflows it is routed to low priority server.

For session based applications, an important issue to handle during load balancing is the persistence of multiple requests from a particular user session. Since load balancing can route successive requests from a user session to different servers, maintaining the state or the information of the session is important. Following are the different persistence handling approaches.

Sticky Sessions

In this approach **a**ll requests belonging to a user is routed to same server. These sessions are called as sticky sessions. The benefit of this approach is that it makes session management simple. However, a drawback of this approach is that if a server fails, all the sessions belonging to that server are lost, since there is no automatic failover possible.

Session Database

In this approach, all the session information is stored separately in a session database which is often replicated to avoid single point failure. This approach involves additional overhead of storing the session information, however, unlike the sticky session approach, this approach allows automatic failover.

Browser Cookies

In this approach, the session information is stored in the client side in the form of browser cookies. The benefit of this approach is that it makes the session management easy and has the least amount of overhead for the load balancer.

URL re-writing

In this approach, a URL re-write engine stores the session information by modifying the URL's on the client side. Though this approach avoids overhead on the load balancer, a drawback is that the amount of session information that can be stored is limited. For applications that require larger amounts of session information, this approach does not work.

Examples

Load balancing can be implemented in software or hardware. Software based load balancers run on standard operating systems and the load balancers can be virtualized like any other cloud resources. Hardware based load balancers implement load balancing algorithms in the Application Specific Integrated Circuits (ASIC's). In the hardware load balancer, the incoming user request are routed to the underlying servers based on some pre-configured load balancing strategy and the response from the servers are sent back either directly to the user or back to the load balancer.

Load Balancer	Type
NginX	Software
HA Proxy	Software
Pound	Software
Varish	Software
Cisco Systems Catalyst 6500	Hardware
Coyote Point Equalizer	Hardware
F5 Networks BIG-IP LTM	Hardware
Barracuda Load Balancer	Hardware

Scalability & Elasticity

Multi-tier applications such as e-Commerce, social networking, business-to-business, etc. can experience rapid changes in their traffic. Capacity planning involves determining the right sizing of each tier of the deployment of an application in terms of the number of resources and the capacity of each resource. Capacity planning may be for computing, storage, memory or network resources.

Vertical Scaling/Scaling up

Involves upgrading the hardware resources (adding additional computing, memory, storage or network resources).

Horizontal scaling /Scaling out

Involves addition of more resources of the same type

Vertical
Horizontal

Deployment

Cloud application deployment design is an iterative process that involves:

Deployment Design

- The variables in this step include the number of servers in each tier, computing, memory and storage capacities of severs, server interconnection, load balancing and replication strategies.

Performance Evaluation

- To verify whether the application meets the performance requirements with the deployment.
- Involves monitoring the workload on the application and measuring various workload parameters such as response time and throughput.
- Utilization of servers (CPU, memory, disk, I/O, etc.) in each tier is also monitored.

Deployment Refinement

- Various alternatives can exist in this step such as vertical scaling (or scaling up), horizontal scaling (or scaling out), alternative server interconnections, alternative load balancing and replication strategies, for instance.

Examples

Cloud Deployment Management tool	**Features**
Right Scale	Design, deploy and manage cloud deployments across multiple public or private clouds
Scalr	Provides tools to automate the management of servers, monitor servers, replaces servers that fail, provides auto scaling and backups
Kaavo	Deploying applications across multiple clouds, managing distributed applications and automating high availability
Cloudstack	Allows simple and cost effective deployment management and configuration of cloud computing environments

Replication

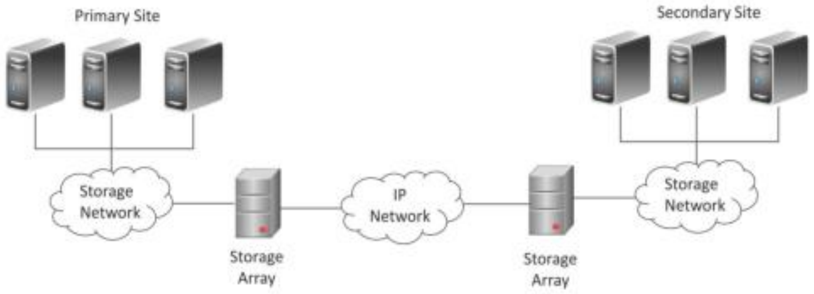
Replication is used to create and maintain multiple copies of the data in the cloud.

To provide solution for disaster recovery

Organization can plan for disaster recovery without having capital expenditure on purchasing configuring and managing secondary site locations.

Array Based

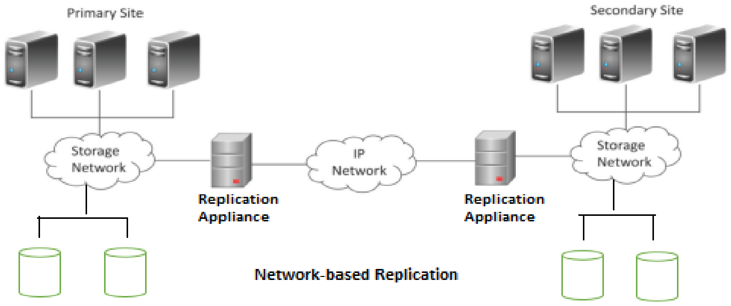
It copies data from a local storage array to remote storage array, It can work in heterogeneous environments with different OS. It uses SAN (storage area network) or NAS (Network attached storage) to replicate. Similar arrays are required in local and remote locations. Setting up this environment is costlier.



Array-based Replication

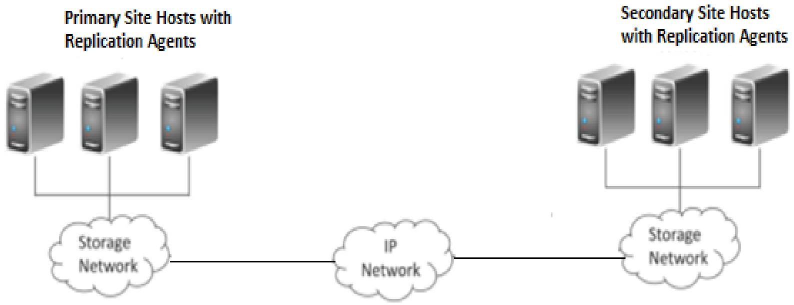
Network Based

It uses an appliance that sits between the network and intercepts the packets sent from host and storage arrays. The intercepted packets are replicated to a secondary location. It Supports heterogeneous environments and requires a single point management. It involves higher initial costs due to replication hardware and software.



Host-Based

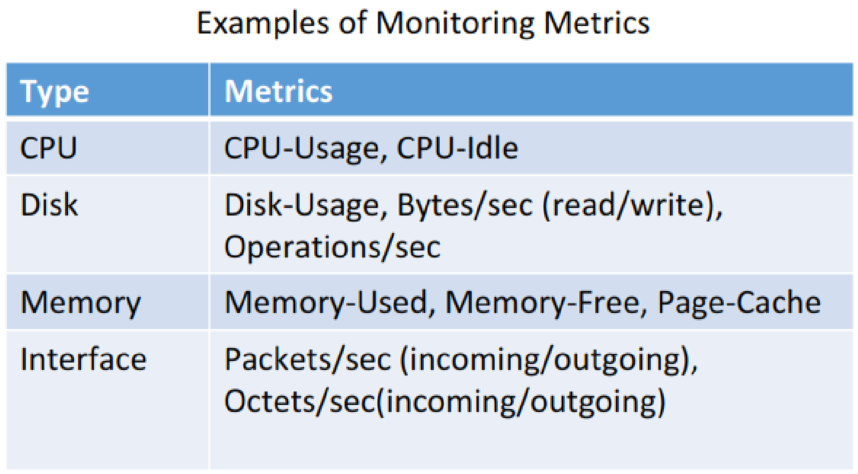
It runs on standard servers and uses software to transfer data from a local to remote location. Host will act as replication control mechanism. An agent is installed on the host that communicate with the agents on the other hosts. Replication can be block based or file based. Entire VM’s can replicated in real-time


Primary Site Hosts with Replication Agents
Secondary Site Hosts with Replication Agents
Storage Network
IP Network
Storage Network

Host-based Replication

Monitoring

Cloud resources can be monitored by monitoring services provided by the cloud service providers. Monitoring services allow cloud users to collect and analyze the data on various monitoring metrics. A monitoring service collects data on various system and application metrics from the cloud computing instances. Monitoring of cloud resources is important because it allows the users to keep track of the health of applications and services deployed in the cloud. For example, an organization which has its website hosted in the cloud can monitor the performance of the website and also the website traffic. With the monitoring, the data available at run-time users can make operational decisions such as scaling up or scaling down cloud resources.



Examples of Monitoring Metrics

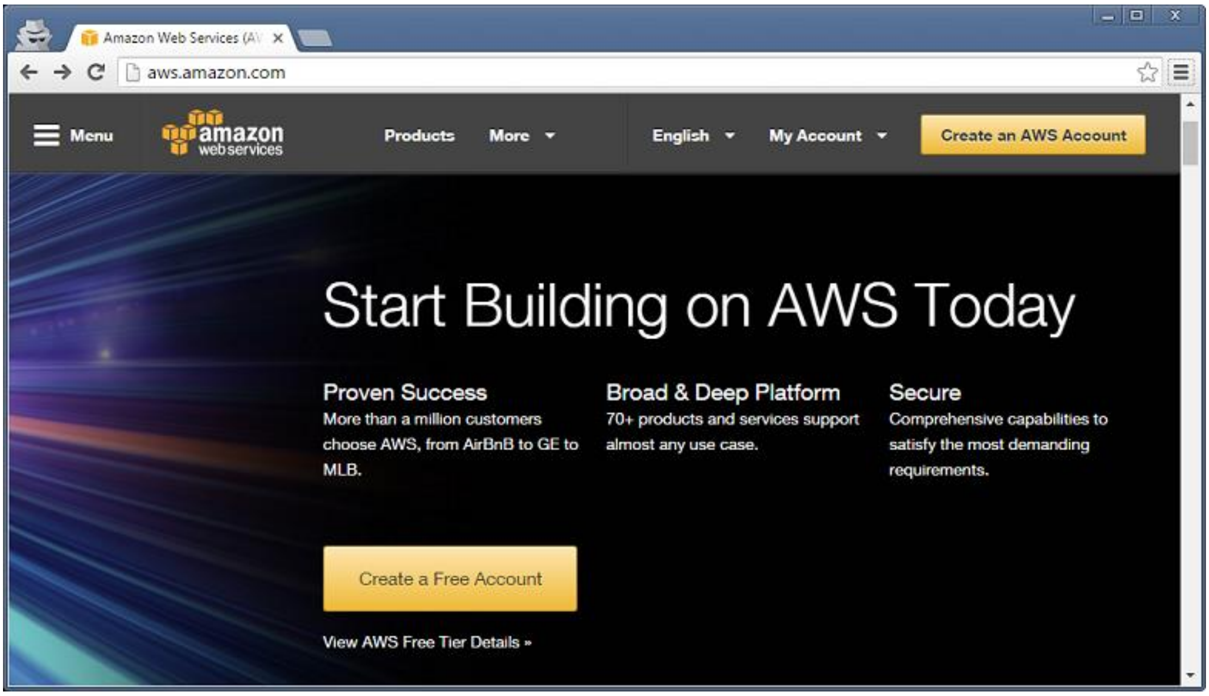
Type	Metrics
CPU	CPU-Usage, CPU-Idle
Disk	Disk-Usage, Bytes/sec (read/write), Operations/sec
Memory	Memory-Used, Memory-Free, Page-Cache
Interface	Packets/sec (incoming/outgoing), Octets/sec(incoming/outgoing)

Identity Access Management

Identity and Access Management (IDAM) for cloud describes the authentication and authorization of users to provide secure access to cloud resources. Organizations with multiple users can use IDAM services provided by the cloud service provider for management of user identifiers and user permissions. IDAM services allow organizations to centrally manage users, access permissions, security credentials and access keys. Organizations can enable role-based access control to cloud resources and applications using the IDAM services. IDAM services allow creation of user groups where all the users in a group have the same access permissions. Identity and Access Management is enabled by a number of technologies such as OpenAuth, Role-based Access Control (RBAC), Digital Identities, Security Tokens, Identity Providers, etc.

1.8 Setting up a Cloud account

Step 1: Go to aws.amazon.com and click *Create a Free Account*



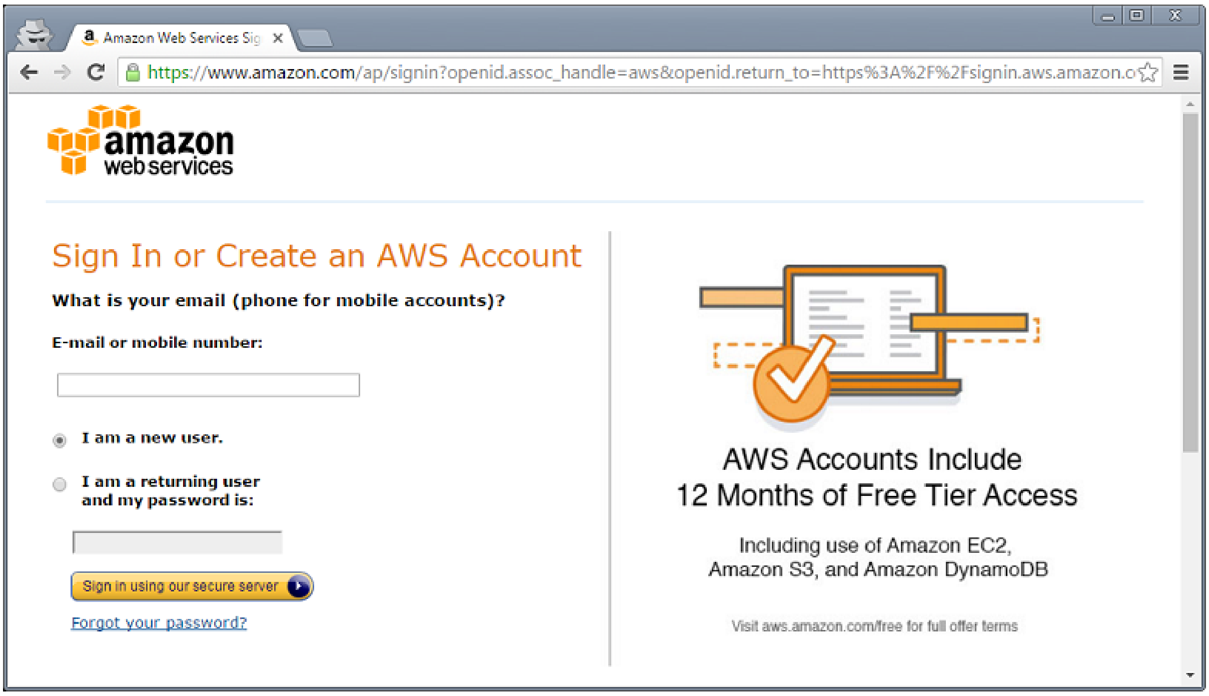
Start Building on AWS Today

Proven Success
More than a million customers choose AWS, from AirBnB to GE to MLB.

Broad & Deep Platform
70+ products and services support almost any use case.

Secure
Comprehensive capabilities to satisfy the most demanding requirements.

Step 2: Enter your email address. You may use an existing Amazon account (i.e. an account you use for shopping at amazon.com) if you have one or select *I am a new user*.



https://www.amazon.com/ap/signin?openid.assoc_handle=aws&openid.return_to=https%3A%2F%2Fsignin.aws.amazon.c

Sign In or Create an AWS Account

What is your email (phone for mobile accounts)?

E-mail or mobile number:

● I am a new user.

○ I am a returning user and my password is:

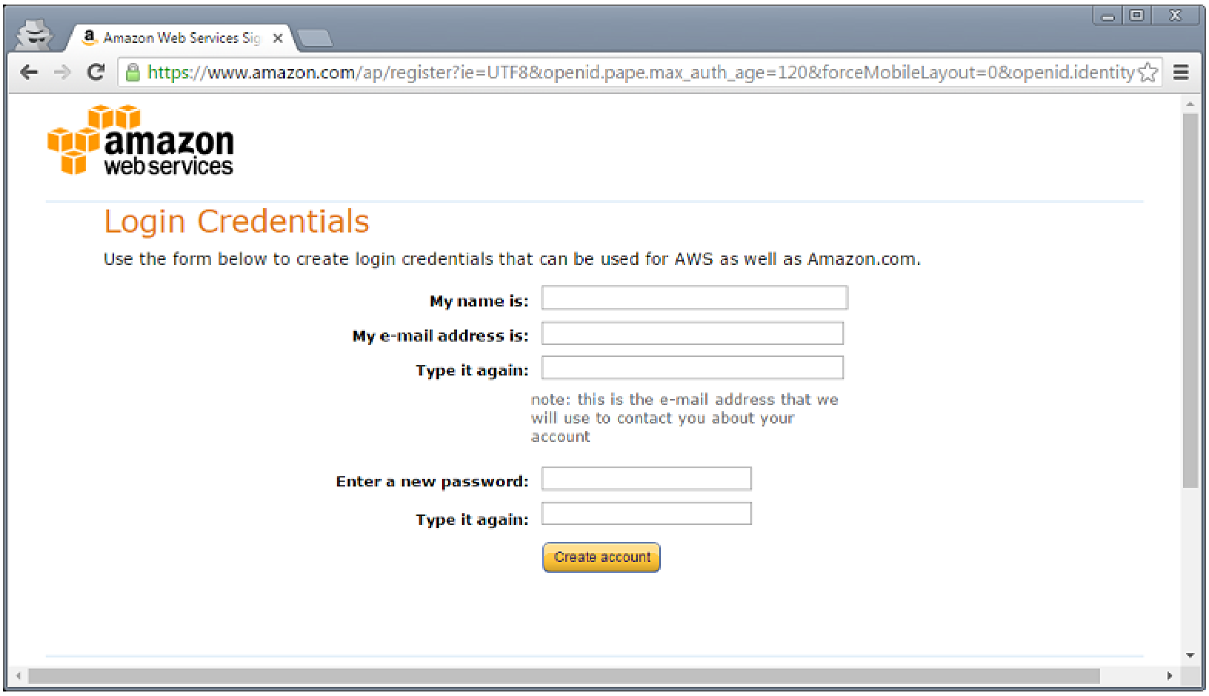
Forgot your password?

AWS Accounts Include
12 Months of Free Tier Access

Including use of Amazon EC2,
Amazon S3, and Amazon DynamoDB

Visit aws.amazon.com/free for full offer terms

Step 3: Enter the requested information and click *Create Account*.



https://www.amazon.com/ap/register?ie=UTF8&openid.pape.max_auth_age=120&forceMobileLayout=0&openid.identity

Login Credentials

Use the form below to create login credentials that can be used for AWS as well as Amazon.com.

My name is:

My e-mail address is:

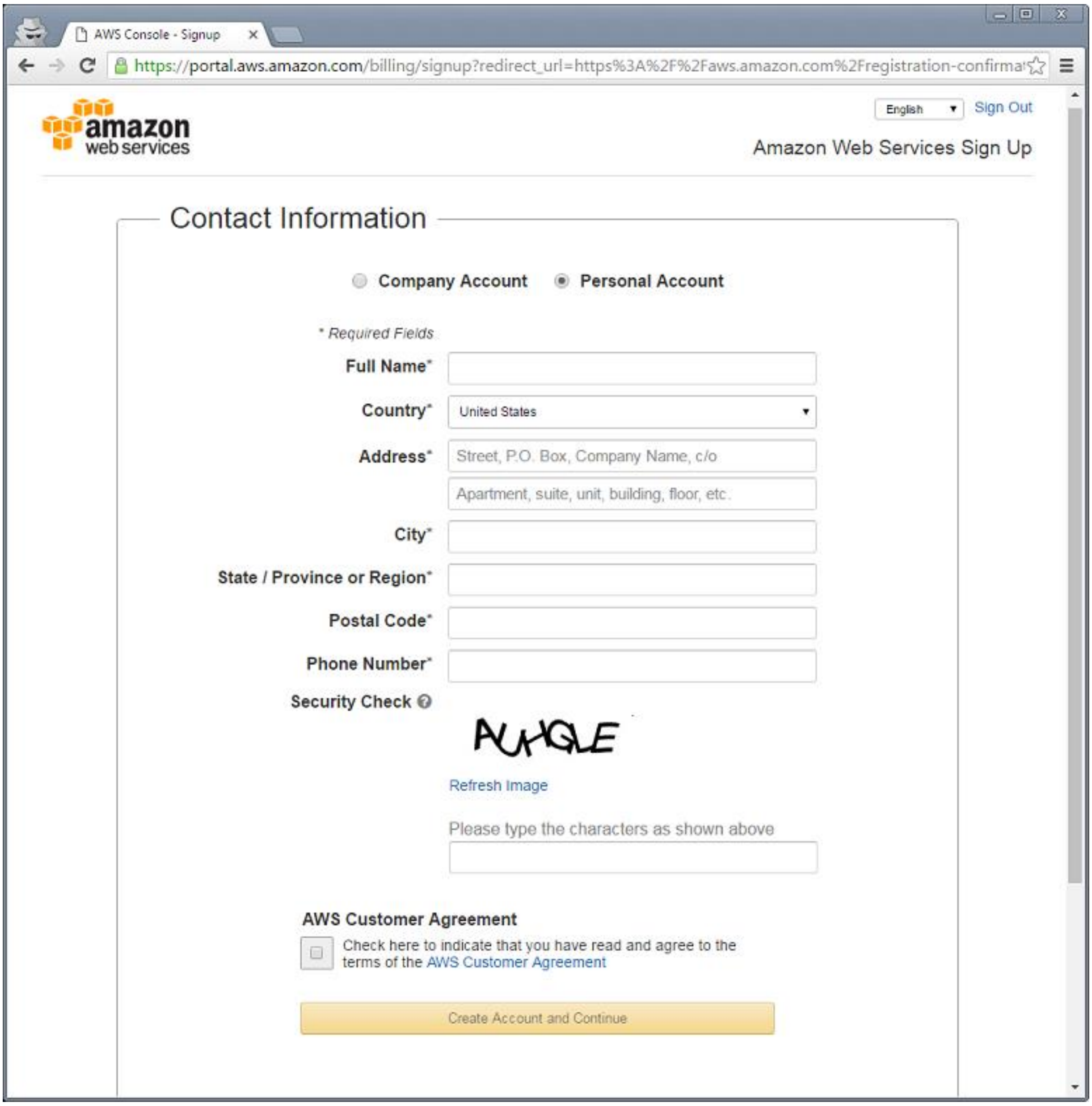
Type it again:

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

Enter a new password:

Type it again:

Step 4: Select *Personal Account*, enter your contact information, and complete the security check. Click *Create Account and Continue* when finished.



AWS Console - Signup

https://portal.aws.amazon.com/billing/signup?redirect_url=https%3A%2F%2Faws.amazon.com%2Fregistration-confirma

English ▾ Sign Out

Amazon Web Services Sign Up

Contact Information

○ Company Account ● Personal Account

* Required Fields

Full Name*

Country* United States ▾

Address* Street, P.O. Box, Company Name, c/o

Apartment, suite, unit, building, floor, etc.

City*

State / Province or Region*

Postal Code*

Phone Number*

Security Check

Refresh Image

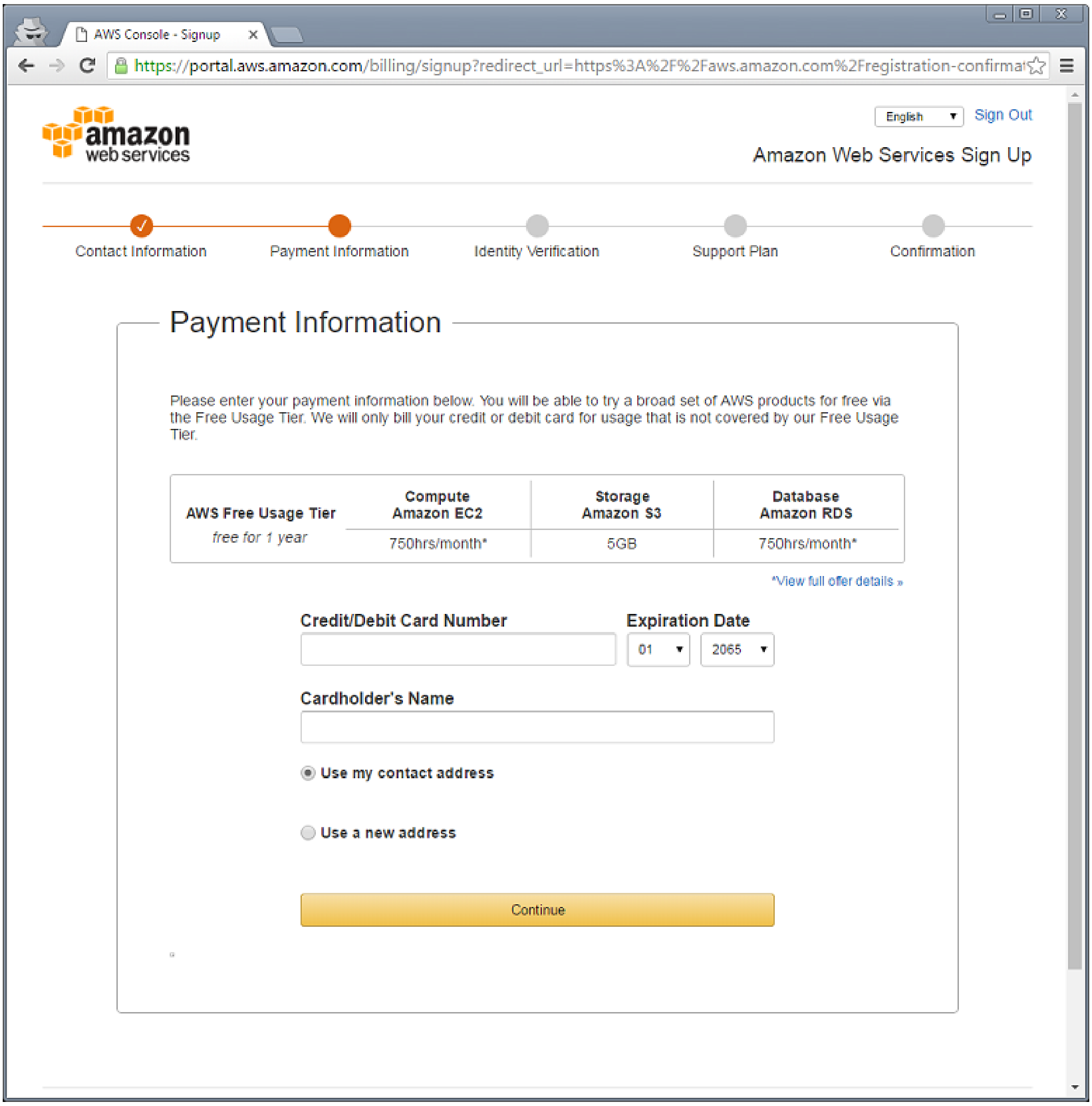
Please type the characters as shown above

AWS Customer Agreement

□ Check here to indicate that you have read and agree to the terms of the AWS Customer Agreement

Create Account and Continue

Step 5: Enter payment information and click *Continue*. Your credit card will not be charged until you begin using services above and beyond the Free Tier (if applicable) and any AWS credit codes entered into your account. (Note that there are a few AWS services that are not eligible to be covered by AWS credits. Review the FAQs after accessing the AWS Educate portal for any exclusions.)



AWS Console - Signup
https://portal.aws.amazon.com/billing/signup?redirect_url=https%3A%2F%2Faws.amazon.com%2Fregistration-confirmat
amazon web services
English
Sign Out

Amazon Web Services Sign Up

Payment Information

Please enter your payment information below. You will be able to try a broad set of AWS products for free via the Free Usage Tier. We will only bill your credit or debit card for usage that is not covered by our Free Usage Tier.

AWS Free Usage Tier	Compute Amazon EC2	Storage Amazon S3	Database Amazon RDS
free for 1 year	750hrs/month*	5GB	750hrs/month*

*View full offer details »

Credit/Debit Card Number

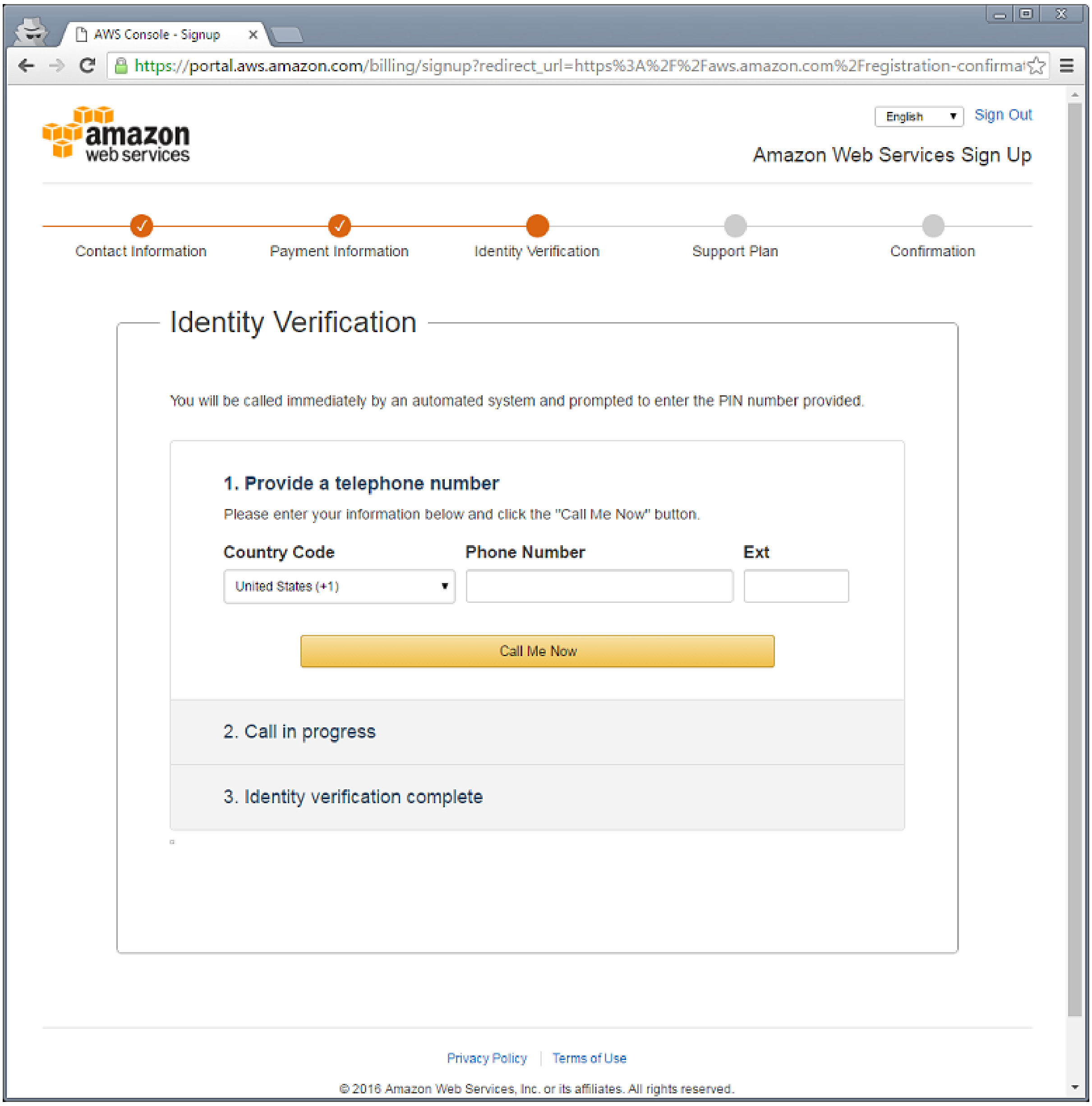
Expiration Date: 01 2065

Cardholder's Name

● Use my contact address

○ Use a new address

Step 6: Provide a phone number for Identity Verification and click *Call Me Now*. You must be able to receive a voice call from Amazon's automated identity verification system. A PIN will be displayed on your screen. Enter the PIN number when prompted by the identity verification system.



Amazon Web Services Sign Up

Identity Verification

You will be called immediately by an automated system and prompted to enter the PIN number provided.

1. Provide a telephone number

Please enter your information below and click the "Call Me Now" button.

Country Code: United States (+1)

Phone Number:

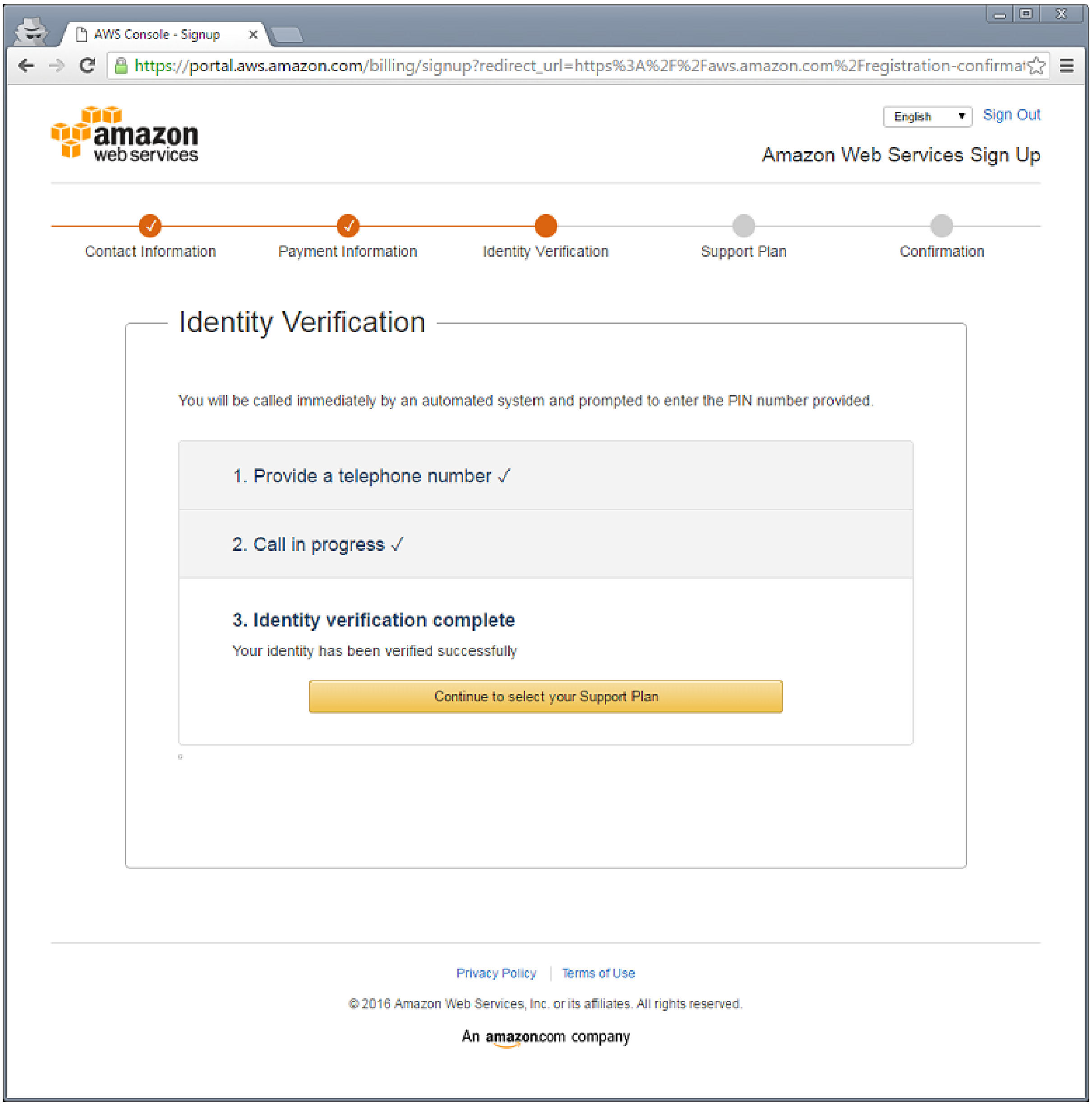
Ext:

2. Call in progress

3. Identity verification complete

© 2016 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Step 7: Once this process is completed, you will be able to continue the registration process.



AWS Console - Signup
https://portal.aws.amazon.com/billing/signup?redirect_url=https%3A%2F%2Faws.amazon.com%2Fregistration-confirmat
amazon web services
English
Sign Out

Amazon Web Services Sign Up

Identity Verification

You will be called immediately by an automated system and prompted to enter the PIN number provided.

1. Provide a telephone number ✓

2. Call in progress ✓

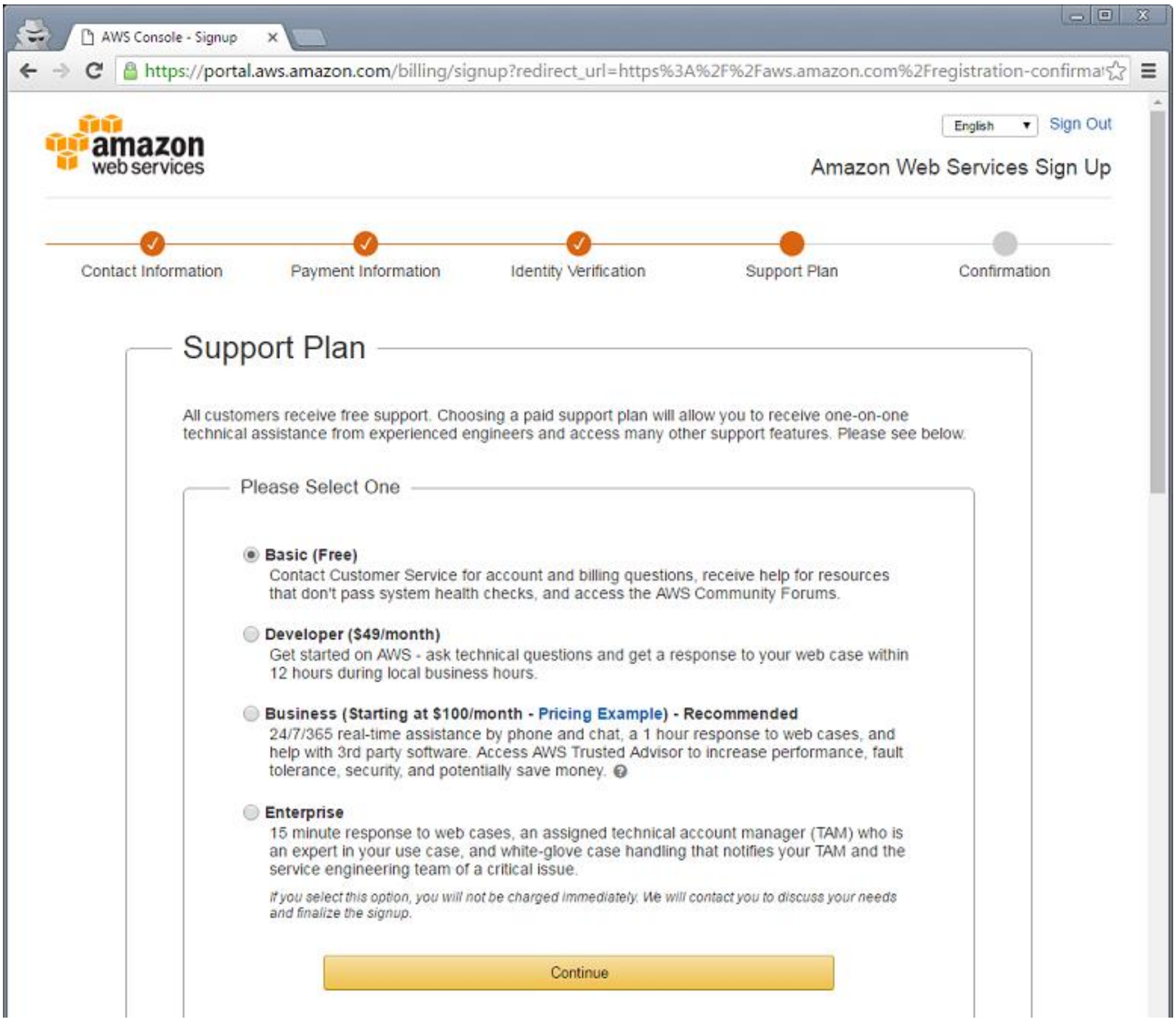
3. Identity verification complete

Your identity has been verified successfully

© 2016 Amazon Web Services, Inc. or its affiliates. All rights reserved.

An amazon.com company

Step 8: Select a Support Plan and click *Continue*.



AWS Console - Signup
https://portal.aws.amazon.com/billing/signup?redirect_url=https%3A%2F%2Faws.amazon.com%2Fregistration-confirma
amazon web services
English
Sign Out
Amazon Web Services Sign Up
Contact Information
Payment Information
Identity Verification
Support Plan
Confirmation

Support Plan

All customers receive free support. Choosing a paid support plan will allow you to receive one-on-one technical assistance from experienced engineers and access many other support features. Please see below.

Please Select One

● **Basic (Free)**
Contact Customer Service for account and billing questions, receive help for resources that don't pass system health checks, and access the AWS Community Forums.

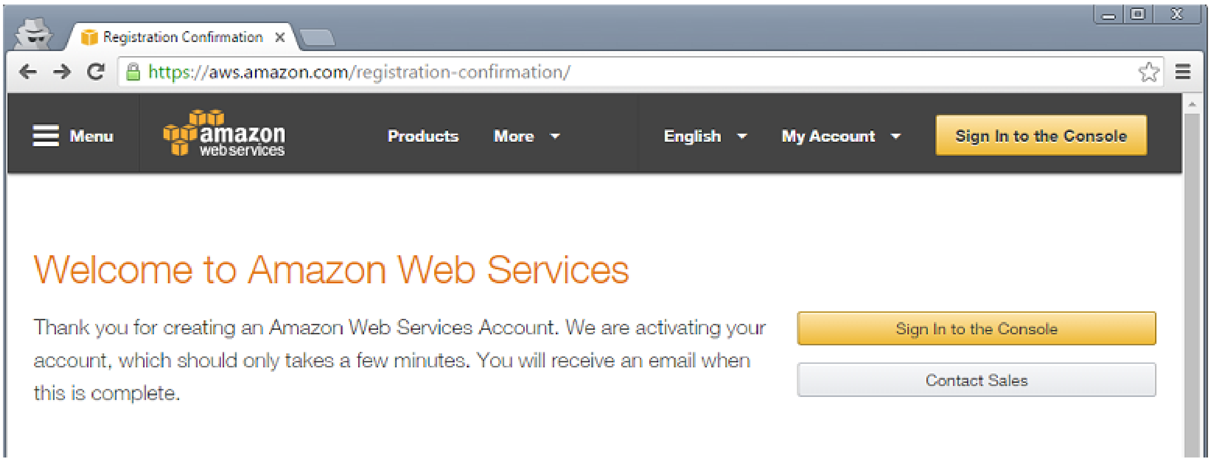
○ **Developer ($49/month)**
Get started on AWS - ask technical questions and get a response to your web case within 12 hours during local business hours.

○ **Business (Starting at $100/month - Pricing Example) - Recommended**
24/7/365 real-time assistance by phone and chat, a 1 hour response to web cases, and help with 3rd party software. Access AWS Trusted Advisor to increase performance, fault tolerance, security, and potentially save money.

○ **Enterprise**
15 minute response to web cases, an assigned technical account manager (TAM) who is an expert in your use case, and white-glove case handling that notifies your TAM and the service engineering team of a critical issue.

If you select this option, you will not be charged immediately. We will contact you to discuss your needs and finalize the signup.

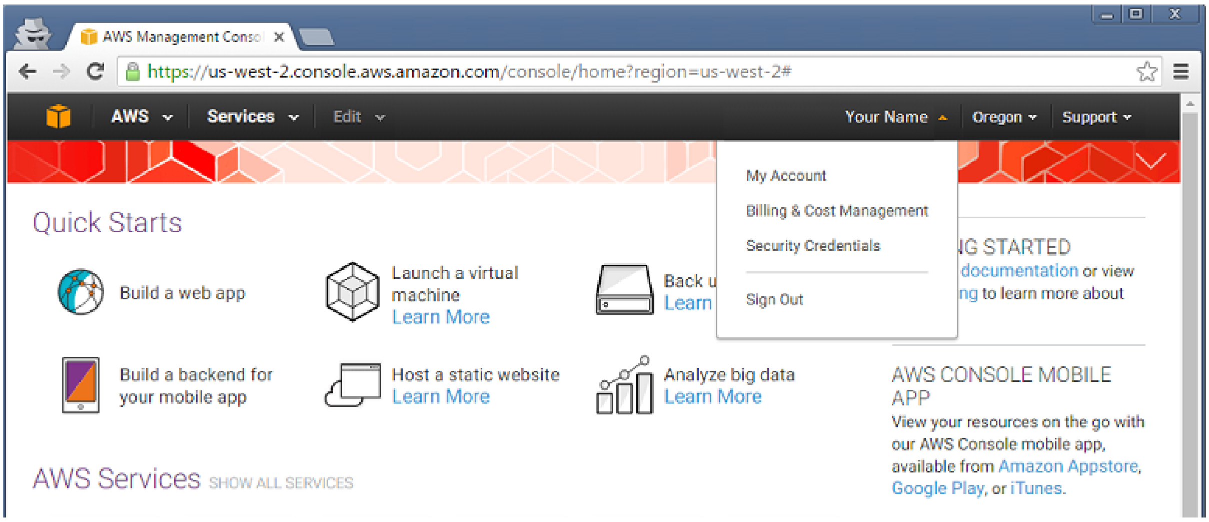
Step 9: You will be returned to the AWS login screen. Click *Sign in to the Console*.



Welcome to Amazon Web Services

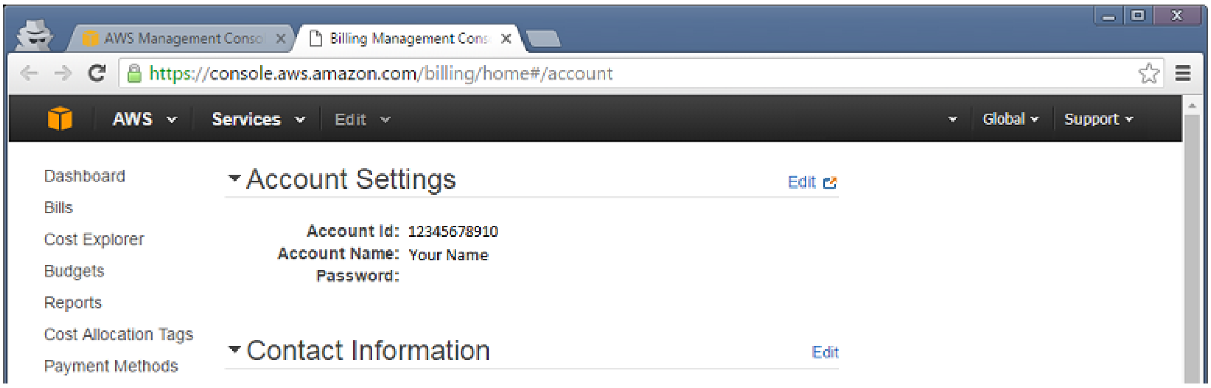
Thank you for creating an Amazon Web Services Account. We are activating your account, which should only takes a few minutes. You will receive an email when this is complete.

Step 10: After logging using the account you just created, click on *Your Name* in the upper right and then click *My Account*.



View your resources on the go with our AWS Console mobile app, available from Amazon Appstore, Google Play, or iTunes.

Step 11: Information about your account will be displayed.



Account Settings

Account Id: 12345678910
Account Name: Your Name
Password:

Contact Information

1.9 Identity Access Management

Identity and access management (IAM) in enterprise IT is about defining and managing the roles and access privileges of individual network users and the circumstances in which users are granted (or denied) those privileges. Those users might be customers (customer identity management) or employees (employee identity management. The core objective of IAM systems is one digital identity per individual. Once that digital identity has been established, it must be maintained, modified and monitored throughout each user's "access lifecycle."

IAM systems provide administrators with the tools and technologies to change a user's role, track user activities, create reports on those activities, and enforce policies on an ongoing basis. These systems are designed to provide a means of administering user access across an entire enterprise and to ensure compliance with corporate policies and government regulations.

Identity and management technologies include (but aren't limited to) password-management tools, provisioning software, security-policy enforcement applications, reporting and monitoring apps and identity repositories

AWS IAM

AWS Identity and Access Management (IAM) helps you control access to your AWS resources.

When you first create an AWS Account, you have complete access to all AWS services and resources in that account. This identity is the root user for the account. An IAM identity provides access to an AWS account.

What is an IAM User?

An IAM user is an entity that represents a person or application that interacts with AWS services.

IAM users can log in to the AWS Management Console for interactive tasks, as well as the ability to make programmatic requests using the API & CLI. IAM Users can be assigned permissions directly or be part of a group with specific permissions.

An IAM user has a name and password that they use to log in to the AWS management console. Users can also create up to two access keys that they can use to for programmatic access to AWS.

What are IAM Groups?

An IAM group is a collection of users that share access control policies. The group members have permission to perform specified actions on objects within the group's scope. For example, if you grant read-only access to all of your EC2 instances, then any group member will be able to view information about those instances.

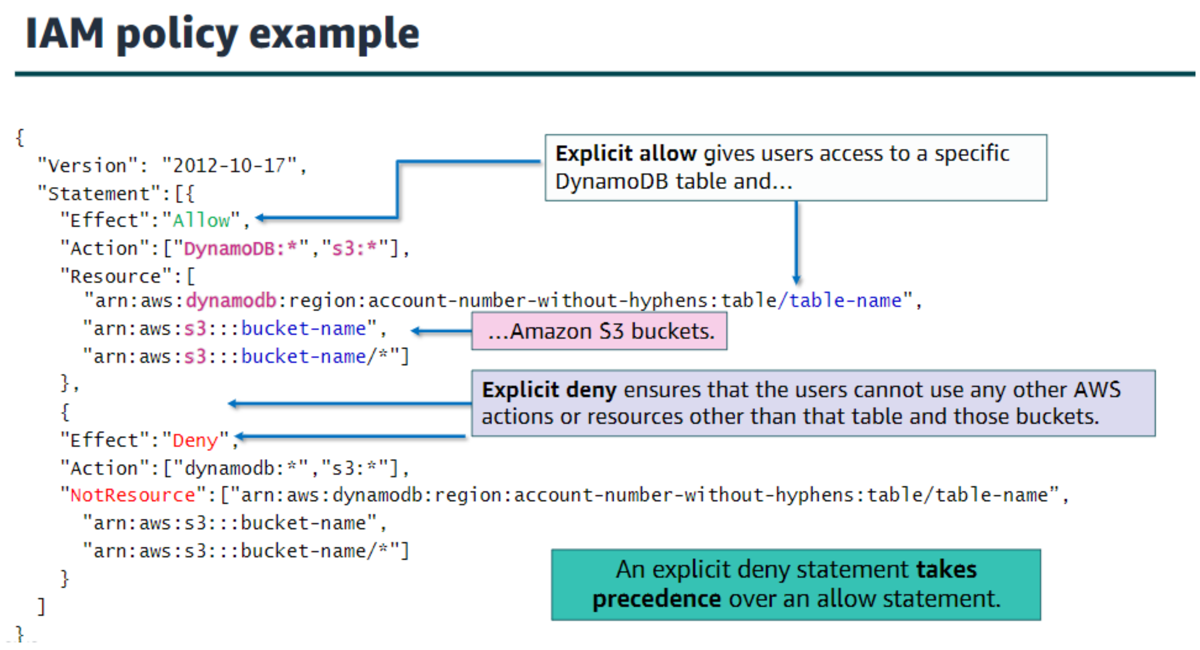
What is an AWS IAM Role?

IAM roles are identities you can create with specific permissions for short durations. You can assign IAM roles to entities you trust so that those entities can assume the role when needed. IAM Roles are primarily meant for internal use.

An IAM role has no associated credentials (password or access keys). The lack of credentials is one of the main differences between a User and a Role. A role can be temporarily assumed by a user, service, or application that has been granted permission to assume the role.

What is IAM Policy?

IAM policies define permissions for an action regardless of the method that you use to perform the operation. For example, if a policy allows the GetUser action, then a user with that policy can get user information from the AWS Management Console.



IAM policy example

```
{
  "Version": "2012-10-17",
  "Statement":[{
    "Effect":"Allow",
    "Action":["DynamoDB:*","s3:*"],
    "Resource":[
      "arn:aws:dynamodb:region:account-number-without-hyphens:table/table-name",
      "arn:aws:s3:::bucket-name",
      "arn:aws:s3:::bucket-name/*"]
    },
    {
    "Effect":"Deny",
    "Action":["dynamodb:*","s3:*"],
    "NotResource":["arn:aws:dynamodb:region:account-number-without-hyphens:table/table-name",
      "arn:aws:s3:::bucket-name",
      "arn:aws:s3:::bucket-name/*"]
    }
  ]
}
```

Explicit allow gives users access to a specific DynamoDB table and…

…Amazon S3 buckets.

Explicit deny ensures that the users cannot use any other AWS actions or resources other than that table and those buckets.

An explicit deny statement **takes precedence** over an allow statement.

When to use each Identity?

IAM User

IAM Users are for external access and are meant for people. You should create an IAM User for the following situations:

Creating your first IAM User: You should not use the root account. You should create a new IAM user with the required permissions to access the AWS Management console.

IAM users can be assigned broad permissions to access multiple AWS services.

However, IAM users should not be used for internal access. For example, an IAM user should not be used to make requests to AWS from an application running on EC2.

IAM Groups

You should use IAM Groups when you want to assign permissions to multiple IAM users. For example, you can create a group called developers and add all developers to that group. You can then assign permissions to the developers group. You can add additional permissions to the group later by modifying the group's policy.

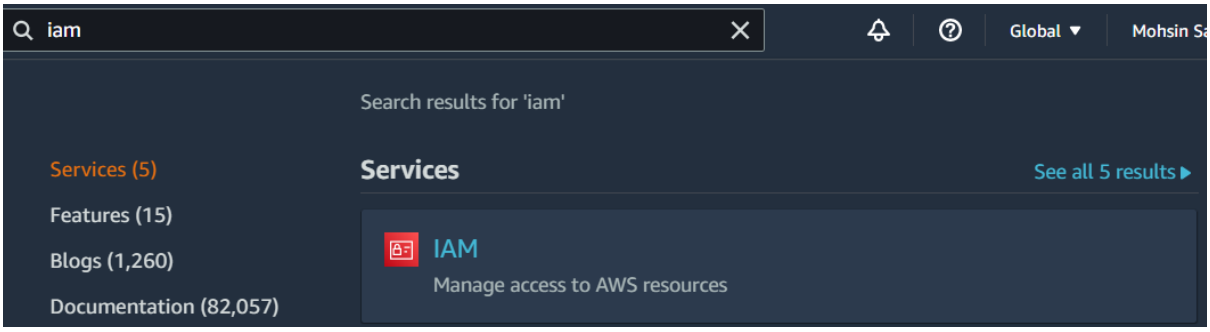
IAM Roles

IAM Roles are meant for internal access where you want to grant permissions to an entity that you trust. IAM Roles use temporary security credentials for accessing AWS services. For example, you can create an IAM Role that allows an EC2 instance to access an S3 bucket. You can then assign the IAM Role to the EC2 instance.

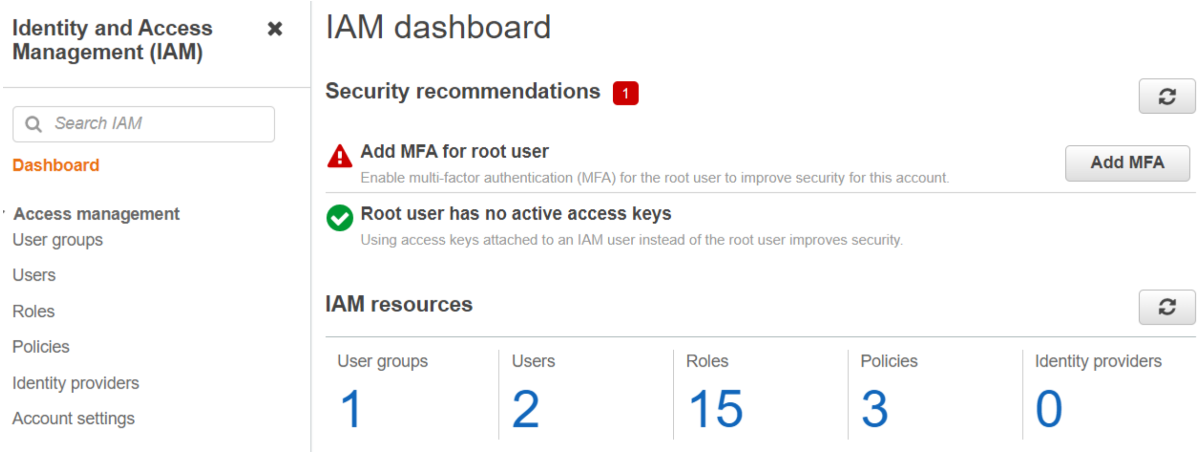
You should create an IAM Role when you are creating an application that runs in AWS and needs to make requests to other AWS Services.

AWS IAM

Sign in to your AWS account and in the top search bar, type IAM.

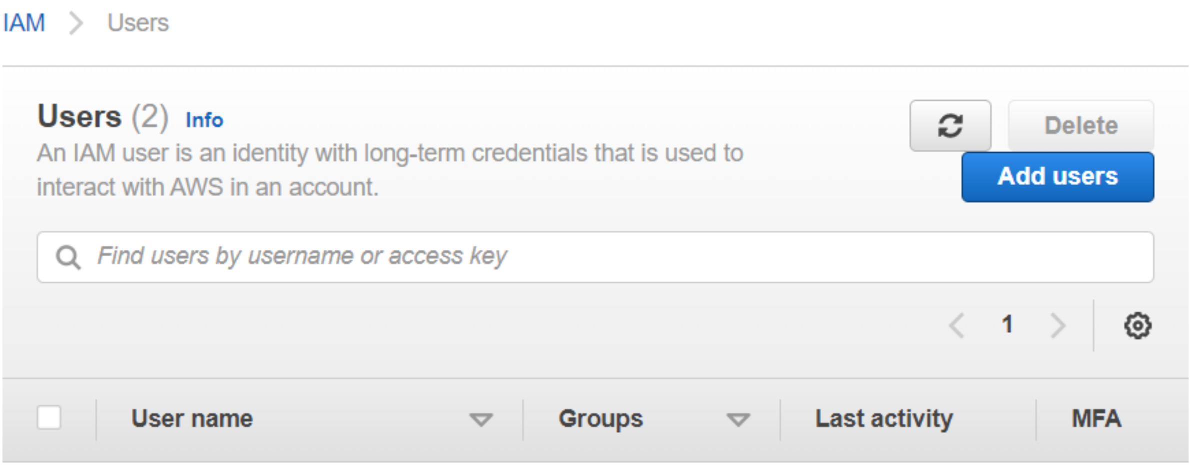


Select the IAM option down the search menu. This will take you to your IAM dashboard.


Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
IAM dashboard
Security recommendations 1
Add MFA for root user
Enable multi-factor authentication (MFA) for the root user to improve security for this account.
Add MFA
Root user has no active access keys
Using access keys attached to an IAM user instead of the root user improves security.
IAM resources

User groups	Users	Roles	Policies	Identity providers
1	2	15	3	0

From the left side panel, Click on Users tab where you will find the Add Users option.



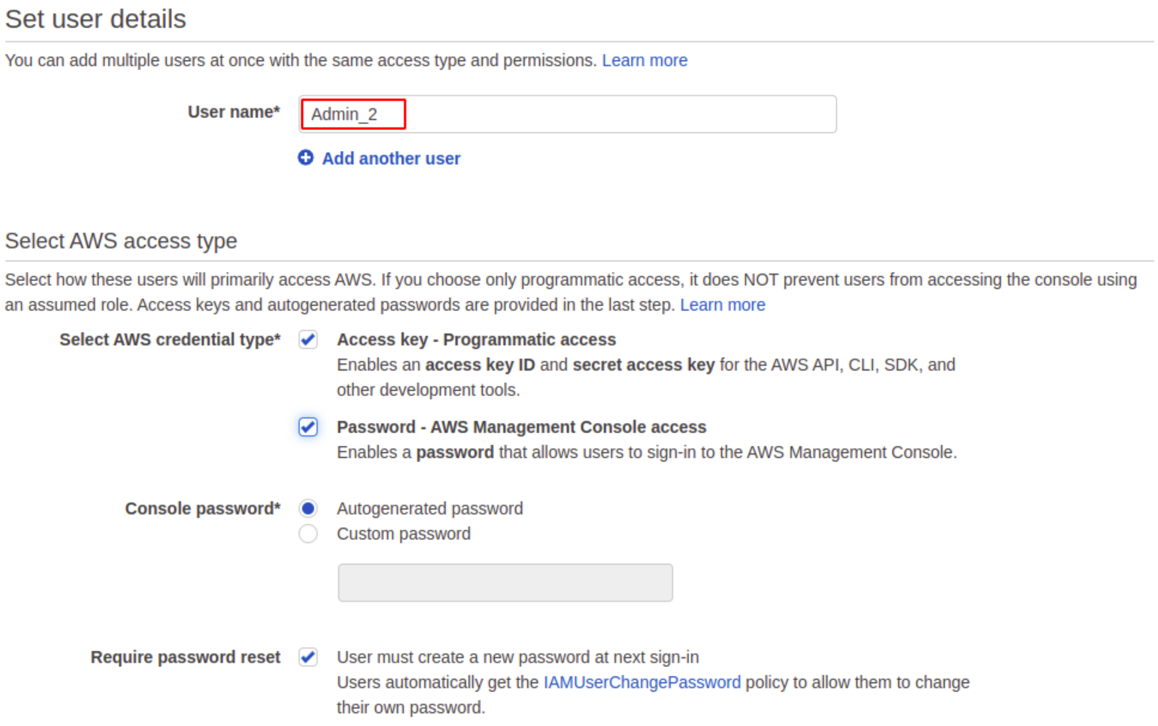
IAM > Users

Users (2) Info

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

□	User name	Groups	Last activity	MFA

To create a new user, you need to configure multiple settings. First, you need to give a username for an IAM user and choose your login credentials type. For logging in to your user account using the AWS management console, you will need to create a password (you can either auto generate a password or use a custom one) or if you want to access your user account from CLI or SDK, you will need to set up an access key which will provide you the access key ID and a secret access key.



Set user details

You can add multiple users at once with the same access type and permissions. Learn more

User name* Admin_2

Select AWS access type

Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. Learn more

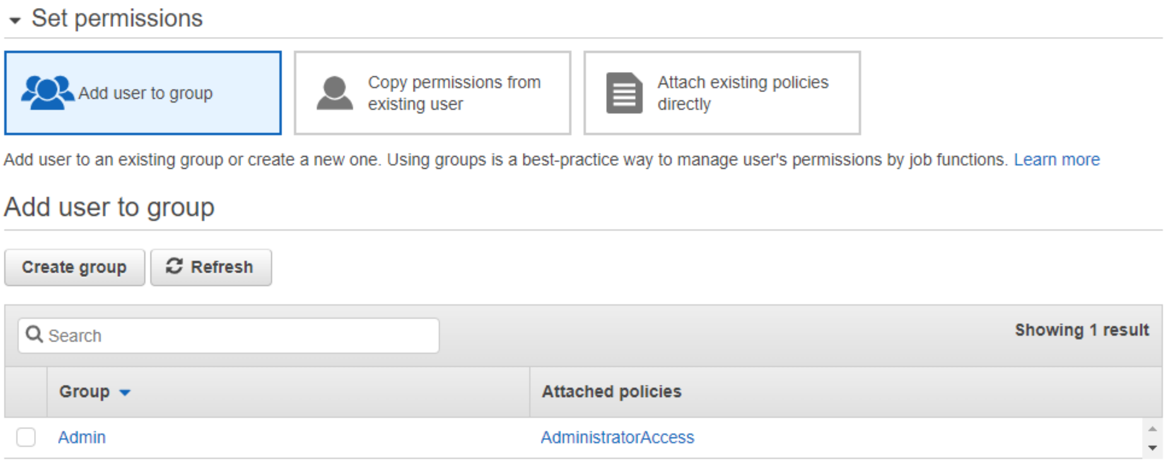
Select AWS credential type* ☑ **Access key - Programmatic access**
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

☑ **Password - AWS Management Console access**
Enables a **password** that allows users to sign-in to the AWS Management Console.

Console password* ● Autogenerated password
○ Custom 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.

In the next section, you will have to manage the permissions assigned to each IAM user in an AWS account. The better approach to give permissions is to create a user group which we will see in the next section but if you want, you can attach a permission policy directly to an IAM user.



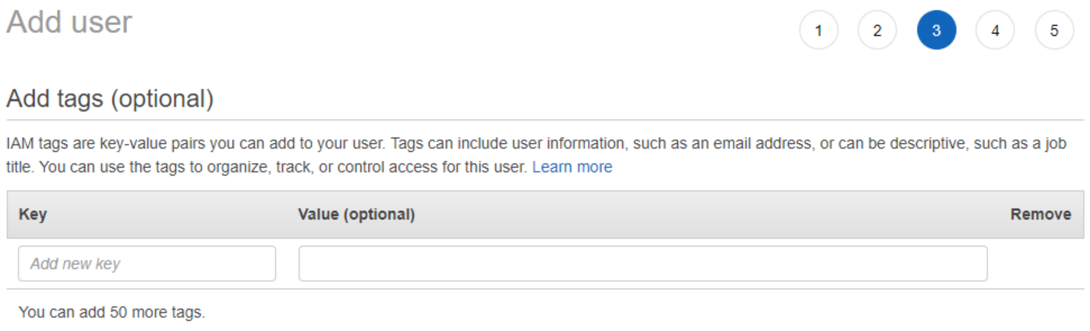
▾ Set permissions
Add user to group
Copy permissions from existing user
Attach existing policies directly

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

Add user to group

	Group ▾	Attached policies
□	Admin	AdministratorAccess

The last step you will find is to add tags which are simple keywords with description to trace all resources in your account related to that keyword. Tags are optional and you may skip them on your choice.



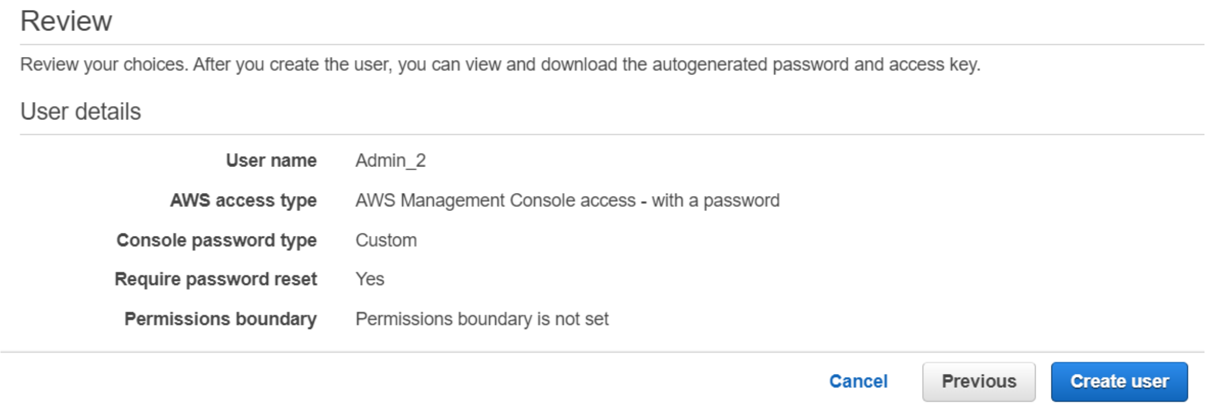
Add tags (optional)

IAM tags are key-value pairs you can add to your user. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this user. Learn more

Key	Value (optional)	Remove
Add new key		

You can add 50 more tags.

Finally, just review the details you have just given about that user and you are good to go to create an IAM user.



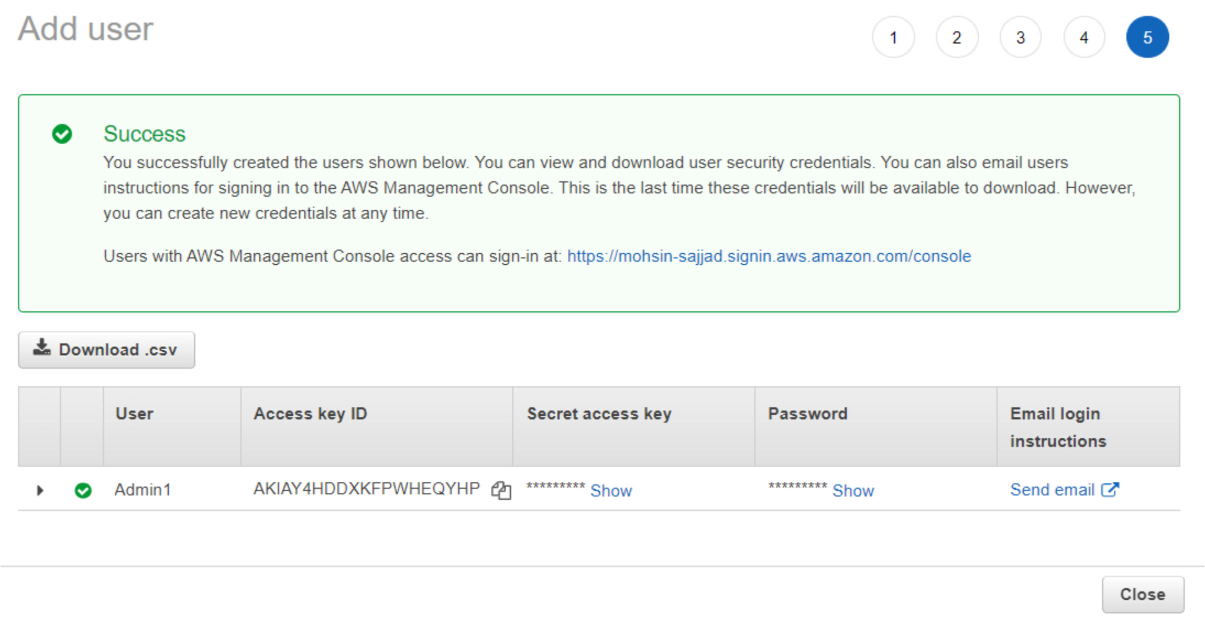
Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name	Admin_2
AWS access type	AWS Management Console access - with a password
Console password type	Custom
Require password reset	Yes
Permissions boundary	Permissions boundary is not set

When you click on create user, a new screen will appear where you will be able to download your user credentials in case you have enabled the access key. This is necessary to download this file as this is the only time you can get them otherwise you will have to create new credentials.



Add user

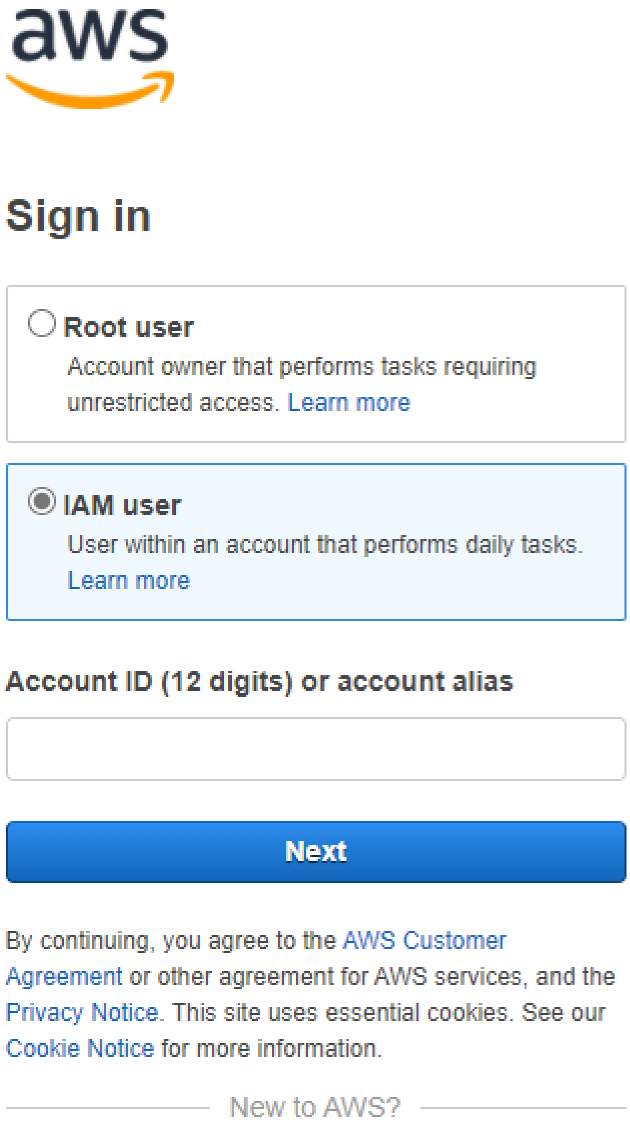
Success

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://mohsin-sajjad.signin.aws.amazon.com/console

		User	Access key ID	Secret access key	Password	Email login instructions
▸	✓	Admin1	AKIAY4HDDXKFPWHEQYHP	********* Show	********* Show	Send email

For logging in to your IAM user account using the management console, you just need to enter your account ID, username and password.



Sign in

○ Root user
Account owner that performs tasks requiring unrestricted access. Learn more

● IAM user
User within an account that performs daily tasks. Learn more

Account ID (12 digits) or account alias

By continuing, you agree to the AWS Customer Agreement or other agreement for AWS services, and the Privacy Notice. This site uses essential cookies. See our Cookie Notice for more information.

New to AWS?

1.10 AWS Key Pair

A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance. Amazon EC2 stores the public key, and you store the private key. You use the private key, instead of a password, to securely access your instances. Anyone who possesses your private keys can connect to your instances, so it's important that you store your private keys in a secure place.

When you launch an instance, you are prompted for a key pair. If you plan to connect to the instance using SSH, you must specify a key pair. You can choose an existing key pair or create a new one. When your instance boots for the first time, the content of the public key that you specified at launch is placed on your Linux instance in an entry within ~/.ssh/authorized_keys. When you connect to your Linux instance using SSH, to log in you must specify the private key that corresponds to the public key content.

The keys that Amazon EC2 uses are 2048-bit SSH-2 RSA keys. You can have up to 5,000 key pairs per Region.

Create a AWS- key pair

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/
2. In the navigation pane, under NETWORK & SECURITY, choose Key Pairs.
3. Choose Create key pair.
4. For Name, enter a descriptive name for the key pair. Amazon EC2 associates the public key with the name that you specify as the key name. A key name can include up to 255 ASCII characters. It can't include leading or trailing spaces.
5. For File format, choose the format in which to save the private key. To save the private key in a format that can be used with OpenSSH, choose pem. To save the private key in a format that can be used with PuTTY, choose ppk.
6. Choose Create key pair.
7. 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 determined by the file format you chose. Save the private key file in a safe place.
8. If you will use an SSH client on a macOS or Linux computer to connect to your Linux instance, use the following command to set the permissions of your private key file so that only you can read it.

```
chmod 400 my-key-pair.pem
```

If you do not set these permissions, then you cannot connect to your instance using this key pair.