

InfraStructure as a Code IN AWS

Using Ansible



[Date]

[Company name]

[Company address]

***Infrastructure as a code***

***Table of contents***

**1.Abstract ………………………………………………………………………………………………………………………………………… 2**

**2.What is cloud ……………………………………………………………………………………………………………………………….. 3**

**3.History of cloud computing……………………………………………………………………………………………………………. 7**

**4.Cloud computing planning…………………………………………………………………………………………………………….. 11**

**5.cloud computing technologies………………………………………………………………………………………………………. 14**

**6.Cloud computing architecture……………………………………………………………………………………………………….. 17**

**7.Cloud computing infrastructure…………………………………………………………………………………………………….. 20**

**8.Cloud deployment Models…………………………………………………………………………………………………………….. 22**

**9.Private cloud model……………………………………………………………………………………………………………………….. 25**

**10.Hybride Clod Model……………………………………………………………………………………………………………………... 28**

**11. Disadvantages…………………………………………………………………………………………………………………………….. 29**

**12.comminity cloud model……………………………………………………………………………………………………………...... 30**

**13.Cloud service models…………………………………………………………………………………………………………………… 32**

**14.Cloud Computing Platform as a Service (PaaS) ……………………………………………………………………………. 36**

**15.Cloud Computing Software as a Service (SaaS)……………………………………………………………………………...41**

**16.Cloud Computing Identity as a Service (IDaaS) ………………………………………………………………………….... 45**

**17.Cloud Computing Network as a Service (Naas)……………………………………………………………………………… 49**

**18.Cloud Computing Data Storage……………………………………………………………………………………………………… 52**

**19. Creating EC2 Instance…………………………………………………………………………………………………………………… 56**

**20. Connect to your Amazon EC2 instance…………………………………………………………………………………………… 61**

*Abstract:*

In this Project we are going to use aws cloud as an infrastructure as a cloud and creating the runtime instance by ansible playbook.

We are going to create an instance in aws console with EC2 storage and spin up. And then we must install the ansible to do required node in the ansible node we have to install boto3 packages to connect aws api console and from aws side also we have do user creation and create secure identity key and key it ready for the script to interact with aws.

Once we have executed the play book on node of ansible if it success then we can have running instance on the aws console so from these we can create infrastructure for an application .Once infrastructure ready we can go ahead and deploy our app o any sample app we can test multiple aws functionalities aws auto scale other functionalities as well

We have inventory file in ansible node from there we can change the required server details in aws instances. we can create according to our need’s; with this we can automate the complete infrastructure of aws and use it simple for app deployment process.

**Cloud computing**

Cloud Computing provides us means by which we can access the applications as utilities over the internet. It allows us to create, configure, and customize the business applications online.

This tutorial will take you through a step-by-step approach while learning Cloud Computing concepts

Cloud Computing provides us means of accessing the applications as utilities over the Internet. It allows us to create, configure, and customize the applications online.

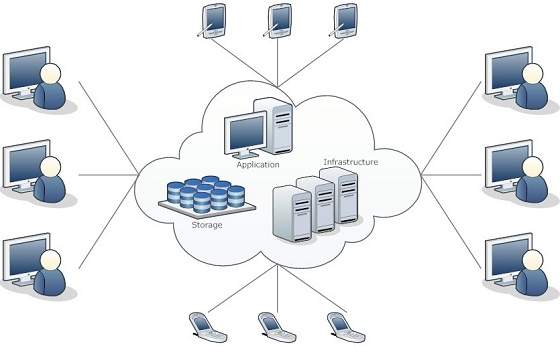
**What is Cloud:**

The term Cloud refers to a Network or Internet. In other words, we can say that Cloud is something, which is present at remote location. Cloud can provide services over public and private networks, i.e., WAN, LAN, VPN.

Applications such as e-mail, web conferencing, customer relationship management (CRM) execute on cloud.

**What is Cloud Computing:**

Cloud Computing refers to manipulating, configuring, and accessing the hardware and software resources remotely. It offers online data storage, infrastructure, and application.



Cloud computing offers platform independency, as the software is not required to be installed locally on the PC. Hence, the Cloud Computing is making our business applications mobile and collaborative

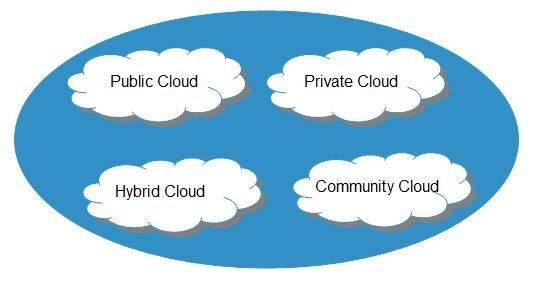
## Basic Concepts

There are certain services and models working behind the scene making the cloud computing feasible and accessible to end users. Following are the working models for cloud computing:

* Deployment Models
* Service Models

### Deployment Models

Deployment models define the type of access to the cloud, i.e., how the cloud is located? Cloud can have any of the four types of access: Public, Private, Hybrid, and Community.



#### **Public Cloud**

The public cloud allows systems and services to be easily accessible to the public. Public cloud may be less secure because of its openness.

#### **Private Cloud**

The private cloud allows systems and services to be accessible within an organization. It is more secured because of its private nature.

#### **Community Cloud**

The community cloud allows systems and services to be accessible by a group of organizations.

#### **Hybrid Cloud**

The hybrid cloud is a mixture of public and private cloud, in which the critical activities are performed using private cloud while the non-critical activities are performed using public cloud.

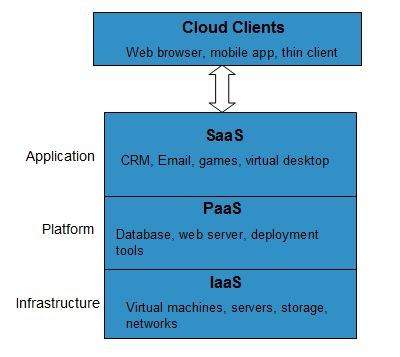
### **Service Models**

Cloud computing is based on service models. These are categorized into three basic service models which are -

* Infrastructure-as–a-Service (IaaS)
* Platform-as-a-Service (PaaS)
* Software-as-a-Service (SaaS)

Anything-as-a-Service (XaaS) is yet another service model, which includes Network-as-a-Service, Business-as-a-Service, Identity-as-a-Service, Database-as-a-Service or Strategy-as-a-Service.

The Infrastructure-as-a-Service (IaaS) is the most basic level of service. Each of the service models inherit the security and management mechanism from the underlying model, as shown in the following diagram:



#### **Infrastructure-as-a-Service (IaaS)**

**IaaS** provides access to fundamental resources such as physical machines, virtual machines, virtual storage, etc.

#### **Platform-as-a-Service (PaaS)**

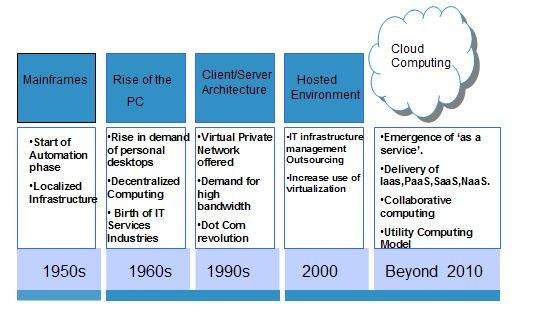
**PaaS** provides the runtime environment for applications, development, and deployment tools, etc.

#### **Software-as-a-Service (SaaS)**

**SaaS** model allows to use software applications as a service to end-users.

## History of Cloud Computing

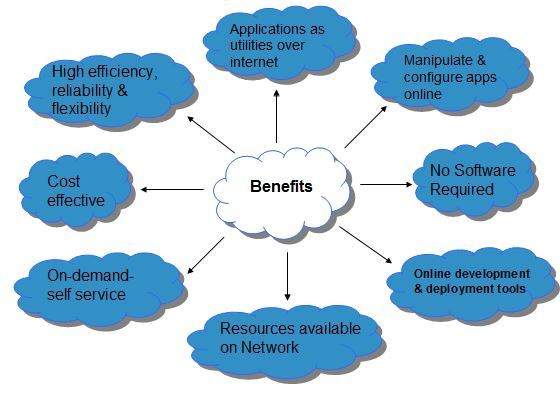
The concept of Cloud Computing came into existence in the year 1950 with implementation of mainframe computers, accessible via thin/static clients. Since then, cloud computing has been evolved from static clients to dynamic ones and from software to services. The following diagram explains the evolution of cloud computing:



## Benefits

Cloud Computing has numerous advantages. Some of them are listed below -

* One can access applications as utilities, over the Internet.
* One can manipulate and configure the applications online at any time.
* It does not require to install a software to access or manipulate cloud application.
* Cloud Computing offers online development and deployment tools, programming runtime environment through PaaS model.
* Cloud resources are available over the network in a manner that provide platform independent access to any type of clients.
* Cloud Computing offers on-demand self-service. The resources can be used without interaction with cloud service provider.
* Cloud Computing is highly cost
* effective because it operates at high efficiency with optimum utilization. It just requires an Internet connection
* Cloud Computing offers load balancing that makes it more reliable.



## Risks related to Cloud Computing

Although cloud Computing is a promising innovation with various benefits in the world of computing, it comes with risks. Some of them are discussed below:

### Security and Privacy

It is the biggest concern about cloud computing. Since data management and infrastructure management in cloud is provided by third-party, it is always a risk to handover the sensitive information to cloud service providers.

Although the cloud computing vendors ensure highly secured password protected accounts, any sign of security breach may result in loss of customers and businesses.

### Lock In

It is very difficult for the customers to switch from one Cloud Service Provider (CSP) to another. It results in dependency on a particular CSP for service.

### Isolation Failure

This risk involves the failure of isolation mechanism that separates storage, memory, and routing between the different tenants.

### Management Interface Compromise

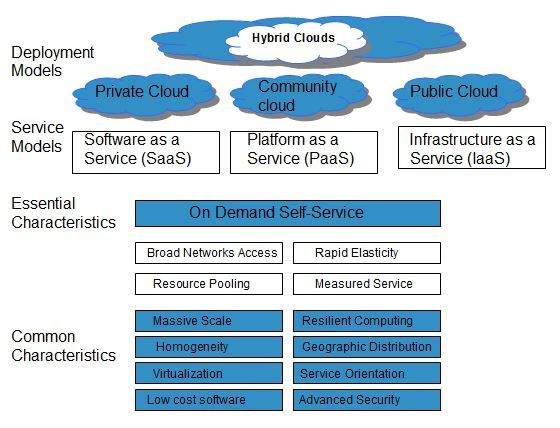
In case of public cloud provider, the customer management interfaces are accessible through the Internet.

### Insecure or Incomplete Data Deletion

It is possible that the data requested for deletion may not get deleted. It happens because either of the following reasons

* Extra copies of data are stored but are not available at the time of deletion
* Disk that stores data of multiple tenants is destroyed.

## Characteristics of Cloud Computing

There are four key characteristics of cloud computing. They are shown in the following diagram: 

### On Demand Self Service

Cloud Computing allows the users to use web services and resources on demand. One can logon to a website at any time and use them.

### Broad Network Access

Since cloud computing is completely web based, it can be accessed from anywhere and at any time.

### Resource Pooling

Cloud computing allows multiple tenants to share a pool of resources. One can share single physical instance of hardware, database, and basic infrastructure.

### Rapid Elasticity

It is very easy to scale the resources vertically or horizontally at any time. Scaling of resources means the ability of resources to deal with increasing or decreasing demand.

The resources being used by customers at any given point of time are automatically monitored.

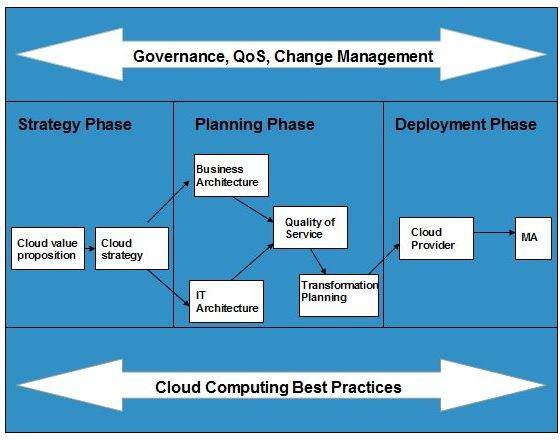
### Measured Service

In this service cloud provider controls and monitors all the aspects of cloud service. Resource optimization, billing, and capacity planning etc. depend on it.

# Cloud Computing Planning

Before deploying applications to cloud, it is necessary to consider your business requirements. Following are the issues one must consider:

* Data Security and Privacy Requirement
* Budget Requirements
* Type of cloud - public, private or hybrid
* Data backup requirements
* Training requirements
* Dashboard and reporting requirements
* Client access requirements
* Data export requirements

To meet all these requirements, it is necessary to have well-compiled planning. In this tutorial, we will discuss the various planning phases that must be practiced by an enterprise before migrating the entire business to cloud. 

## Strategy Phase

In this phase, we analyze the strategy problems that customer might face. There are two steps to perform this analysis:

* Cloud Computing Value Proposition
* Cloud Computing Strategy Planning

### Cloud Computing Value Proposition

In this, we analyze the factors influencing the customers when applying cloud computing mode and target the key problems they wish to solve. These key factors are:

* IT management simplification
* operation and maintenance cost reduction
* business mode innovation
* low cost outsourcing hosting
* high service quality outsourcing hosting.

All of the above analysis helps in decision making for future development.

### Cloud Computing Strategy Planning

The strategy establishment is based on the analysis result of the above step. In this step, a strategy document is prepared according to the conditions a customer might face when applying cloud computing mode.

## Planning Phase

This step performs analysis of problems and risks in the cloud application to ensure the customers that the cloud computing is successfully meeting their business goals. This phase involves the following planning steps:

* Business Architecture Development
* IT Architecture development
* Requirements on Quality of Service Development
* Transformation Plan development

### **Business Architecture Development**

In this step, we recognize the risks that might be caused by cloud computing application from a business perspective.

### **IT Architecture Development**

In this step, we identify the applications that support the business processes and the technologies required to support enterprise applications and data systems.

### **Requirements on Quality of Service Development**

Quality of service refers to the non-functional requirements such as reliability, security, disaster recovery, etc. The success of applying cloud computing mode depends on these non-functional factors.

### **Transformation Plan Development**

In this step, we formulate all kinds of plans that are required to transform current business to cloud computing modes.

## Deployment Phase

This phase focuses on both of the above two phases. It involves the following two steps:

* Selecting Cloud Computing Provider
* Maintenance and Technical Service

### **Selecting Cloud Computing Provider**

This step includes selecting a cloud provider on basis of Service Level Agreement (SLA), which defines the level of service the provider will meet.

### **Maintenance and Technical Service**

Maintenance and Technical services are provided by the cloud provider. They need to ensure the quality of services.

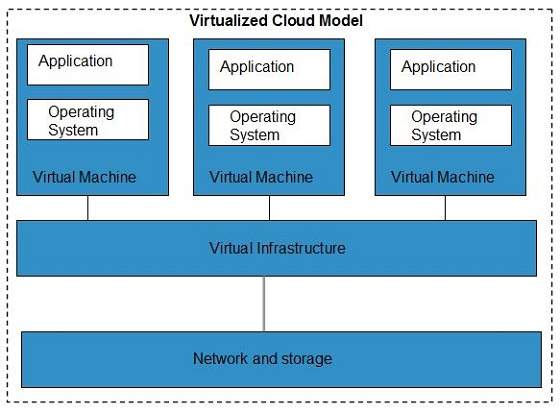
# **Cloud Computing Technologies**

There are certain technologies working behind the cloud computing platforms making cloud computing flexible, reliable, and usable. These technologies are listed below:

* Virtualization
* Service-Oriented Architecture (SOA)
* Grid Computing
* Utility Computing

## Virtualization

Virtualization is a technique, which allows to share single physical instance of an application or resource among multiple organizations or tenants (customers). It does this by assigning a logical name to a physical resource and providing a pointer to that physical resource when demanded.

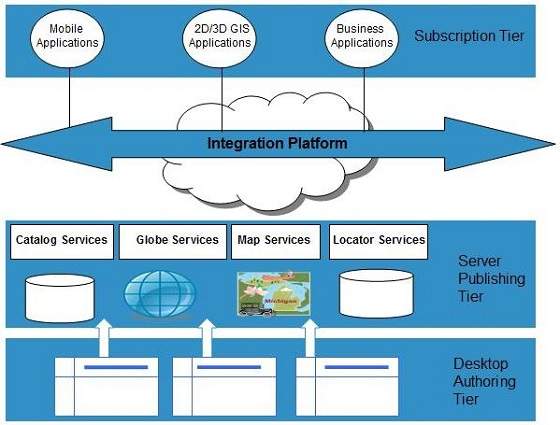


The Multitenant architecture offers virtual isolation among the multiple tenants. Hence, the organizations can use and customize their application as though they each have their instances running.

## Service-Oriented Architecture (SOA)

Service-Oriented Architecture helps to use applications as a service for other applications regardless the type of vendor, product or technology. Therefore, it is possible to exchange the data between applications of different vendors without additional programming or making changes to services.

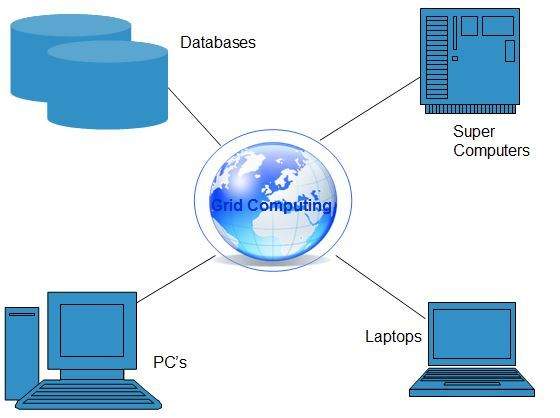
The cloud computing service oriented architecture is shown in the diagram below.



## Grid Computing

Grid Computing refers to distributed computing, in which a group of computers from multiple locations are connected with each other to achieve a common objective. These computer resources are heterogeneous and geographically dispersed.

Grid Computing breaks complex task into smaller pieces, which are distributed to CPUs that reside within the grid.



## Utility Computing

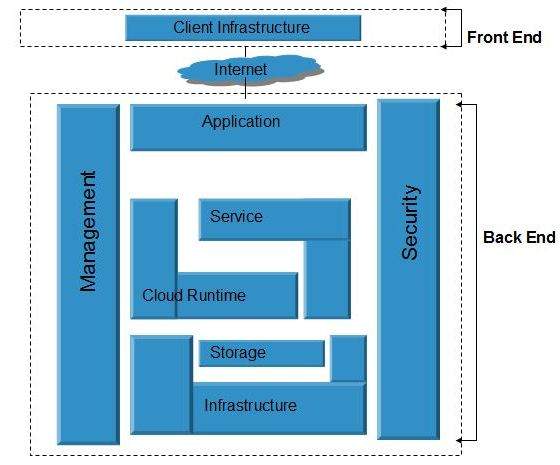
Utility computing is based on **Pay-per-Use model.** It offers computational resources on demand as a metered service. Cloud computing, grid computing, and managed IT services are based on the concept of utility computing.

# **Cloud Computing Architecture**

Cloud Computing architecture comprises of many cloud components, which are loosely coupled. We can broadly divide the cloud architecture into two parts:

* Front End
* Back End

Each of the ends is connected through a network, usually Internet. The following diagram shows the graphical view of cloud computing architecture:



## Front End

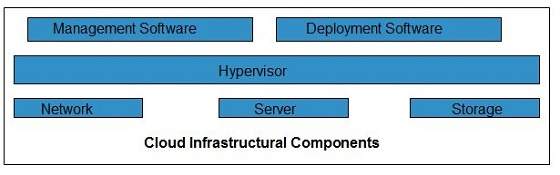
The front end refers to the client part of cloud computing system. It consists of interfaces and applications that are required to access the cloud computing platforms, Example - Web Browser.

## Back End

The back End refers to the cloud itself. It consists of all the resources required to provide cloud computing services. It comprises of huge data storage, virtual machines, security mechanism, services, deployment models, servers, etc.

# **Cloud Computing Infrastructure**

**Cloud infrastructure** consists of servers, storage devices, network, cloud management software, deployment software, and platform virtualization.



### Hypervisor

Hypervisor is a firmware or low-level program that acts as a Virtual Machine Manager. It allows to share the single physical instance of cloud resources between several tenants.

### **Management Software**

It helps to maintain and configure the infrastructure.

### **Deployment Software**

It helps to deploy and integrate the application on the cloud.

### **Network**

It is the key component of cloud infrastructure. It allows to connect cloud services over the Internet. It is also possible to deliver network as a utility over the Internet, which means, the customer can customize the network route and protocol.

### **Server**

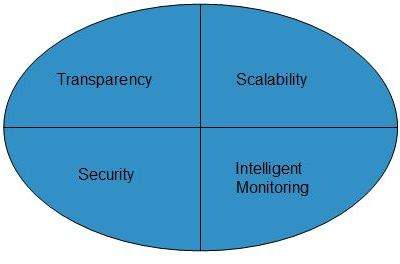
The server helps to compute the resource sharing and offers other services such as resource allocation and de-allocation, monitoring the resources, providing security etc.

### **Storage**

Cloud keeps multiple replicas of storage. If one of the storage resources fails, then it can be extracted from another one, which makes cloud computing more reliable.

## Infrastructural Constraints

Fundamental constraints that cloud infrastructure should implement are shown in the following diagram:



### Transparency

Virtualization is the key to share resources in cloud environment. But it is not possible to satisfy the demand with single resource or server. Therefore, there must be transparency in resources, load balancing and application, so that we can scale them on demand.

### Scalability

Scaling up an application delivery solution is not that easy as scaling up an application because it involves configuration overhead or even re-architecting the network. So, application delivery solution is need to be scalable which will require the virtual infrastructure such that resource can be provisioned and de-provisioned easily.

### Intelligent Monitoring

To achieve transparency and scalability, application solution delivery will need to be capable of intelligent monitoring.

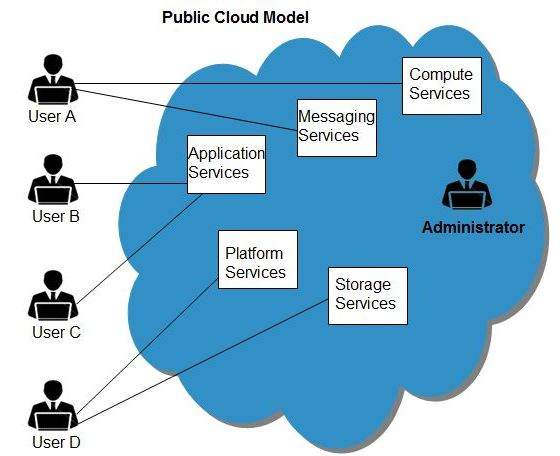
### **Security**

The mega data center in the cloud should be securely architected. Also the control node, an entry point in mega data center, also needs to be secure.

**Cloud Deployment Models**

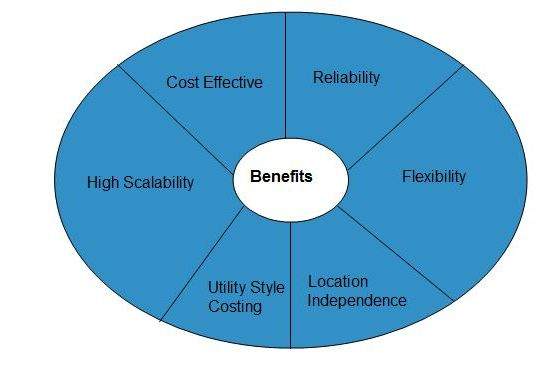
# Public Cloud Model

Public Cloud allows systems and services to be easily accessible to general public. The IT giants such as Google, Amazon and Microsoft offer cloud services via Internet. The Public Cloud Model is shown in the diagram below.



## Benefits

There are many benefits of deploying cloud as public cloud model. The following diagram shows some of those benefits:



### Cost Effective

Since **public cloud** shares same resources with large number of customers it turns out inexpensive.

### **Reliability**

The **public cloud** employs large number of resources from different locations. If any of the resources fails, public cloud can employ another one.

### **Flexibility**

The public cloud can smoothly integrate with private cloud, which gives customers a flexible approach.

### **Location Independence**

Public cloud services are delivered through Internet, ensuring location independence.

### **Utility Style Costing**

Public cloud is also based on **pay-per-use** model and resources are accessible whenever customer needs them.

### High Scalability

Cloud resources are made available on demand from a pool of resources, i.e., they can be scaled up or down according the requirement.

## Disadvantages

Here are some disadvantages of public cloud model:

### Low Security

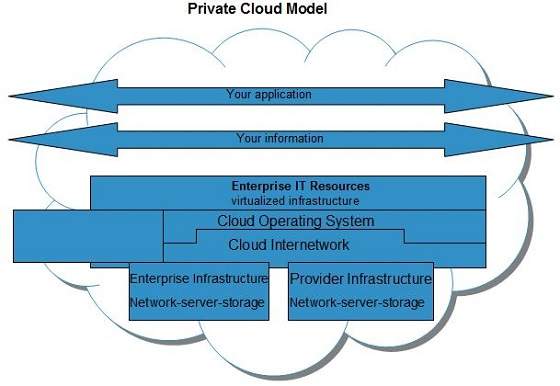
In **public cloud model,** data is hosted off-site and resources are shared publicly, therefore does not ensure higher level of security.

### Less Customizable

It is comparatively less customizable than private cloud.

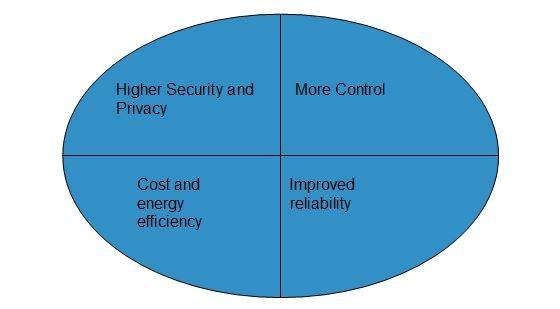
# **Private Cloud Model**

Private Cloud allows systems and services to be accessible within an organization. The Private Cloud is operated only within a single organization. However, it may be managed internally by the organization itself or by third-party. The private cloud model is shown in the diagram below.



## Benefits

There are many benefits of deploying cloud as private cloud model. The following diagram shows some of those benefits:



### **High Security and Privacy**

Private cloud operations are not available to general public and resources are shared from distinct pool of resources. Therefore, it ensures high security and privacy.

### **More Control**

The private cloud has more control on its resources and hardware than public cloud because it is accessed only within an organization.

### **Cost and Energy Efficiency**

The private cloud resources are not as cost effective as resources in public clouds but they offer more efficiency than public cloud resources.

## Disadvantages

Here are the disadvantages of using private cloud model:

### **Restricted Area of Operation**

The private cloud is only accessible locally and is very difficult to deploy globally.

### **High Priced**

Purchasing new hardware in order to fulfill the demand is a costly transaction.

### **Limited Scalability**

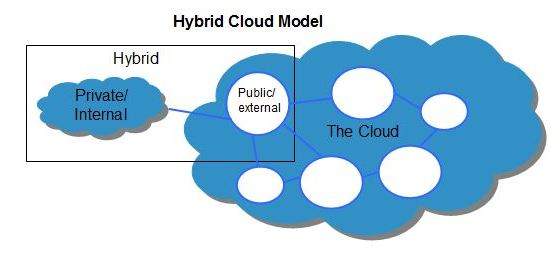
The private cloud can be scaled only within capacity of internal hosted resources.

### **Additional Skills**

To maintain cloud deployment, organization requires skilled expertise.

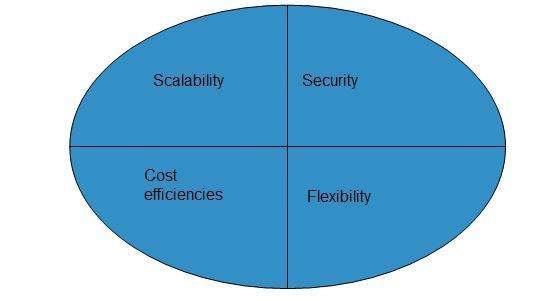
# **Hybrid Cloud Model**

Hybrid Cloud is a mixture of public and private cloud. Non-critical activities are performed using public cloud while the critical activities are performed using private cloud. The Hybrid Cloud Model is shown in the diagram below.



## Benefits

There are many benefits of deploying cloud as hybrid cloud model. The following diagram shows some of those benefits:



### **Scalability**

It offers features of both, the public cloud scalability and the private cloud scalability.

### Flexibility

It offers secure resources and scalable public resources.

### Cost Efficiency

Public clouds are more cost effective than private ones. Therefore, hybrid clouds can be cost saving.

### **Security**

The private cloud in hybrid cloud ensures higher degree of security.

## Disadvantages

### **Networking Issues**

Networking becomes complex due to presence of private and public cloud.

### **Security Compliance**

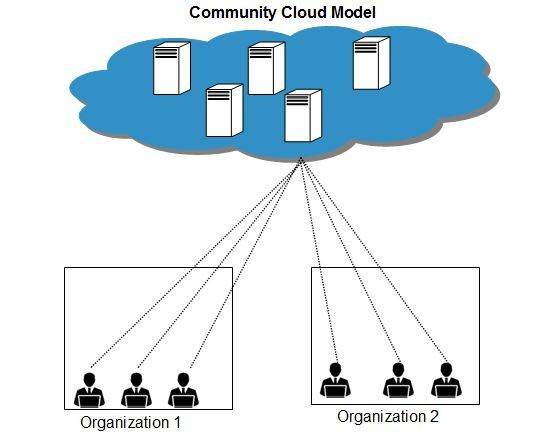
It is necessary to ensure that cloud services are compliant with security policies of the organization.

### **Infrastructure Dependency**

The hybrid cloud model is dependent on internal IT infrastructure, therefore it is necessary to ensure redundancy across data centers.

# Community Cloud Model

Community Cloud allows system and services to be accessible by group of organizations. It shares the infrastructure between several organizations from a specific community. It may be managed internally by organizations or by the third-party. The Community Cloud Model is shown in the diagram below.



## Benefits

There are many benefits of deploying cloud as community cloud model.



### **Cost Effective**

Community cloud offers same advantages as that of private cloud at low cost.

### **Sharing Among Organizations**

Community cloud provides an infrastructure to share cloud resources and capabilities among several organizations.

### **Security**

The community cloud is comparatively more secure than the public cloud but less secured than the private cloud.

## Issues

* Since all data is located at one place, one must be careful in storing data in community cloud because it might be accessible to others.
* It is also challenging to allocate responsibilities of governance, security and cost among organizations.

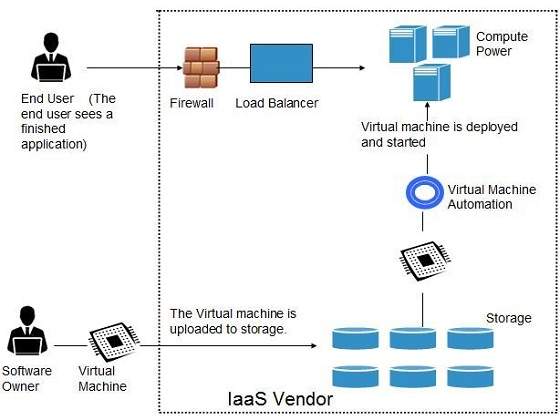
# Cloud Service Models

# Cloud Computing Infrastructure as a Service (IaaS)

Infrastructure-as-a-Service provides access to fundamental resources such as physical machines, virtual machines, virtual storage, etc. Apart from these resources, the IaaS also offers:

* Virtual machine disk storage
* Virtual local area network (VLANs)
* Load balancers
* IP addresses
* Software bundles

All of the above resources are made available to end user via server virtualization. Moreover, these resources are accessed by the customers as if they own them.



## Benefits

**IaaS** allows the cloud provider to freely locate the infrastructure over the Internet in a cost-effective manner. Some of the key benefits of IaaS are listed below:

* Full control of the computing resources through administrative access to VMs.
* Flexible and efficient renting of computer hardware.
* Portability, interoperability with legacy applications.

### **Full control over computing resources through administrative access to VMs**

IaaS allows the customer to access computing resources through administrative access to virtual machines in the following manner:

* Customer issues administrative command to cloud provider to run the virtual machine or to save data on cloud server.
* Customer issues administrative command to virtual machines they owned to start web server or to install new applications.

### **Flexible and efficient renting of computer hardware**

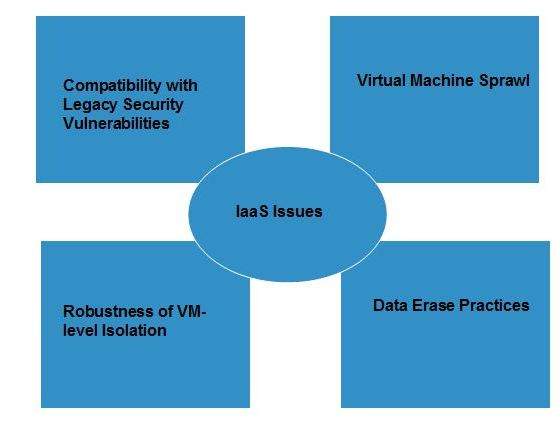
IaaS resources such as virtual machines, storage devices, bandwidth, IP addresses, monitoring services, firewalls, etc. are made available to the customers on rent. The payment is based upon the amount of time the customer retains a resource. Also with administrative access to virtual machines, the customer can run any software, even a custom operating system.

### **Portability, interoperability with legacy applications**

It is possible to maintain legacy between applications and workloads between IaaS clouds. For example, network applications such as web server or e-mail server that normally runs on customer-owned server hardware can also run from VMs in IaaS cloud.

## Issues

IaaS shares issues with PaaS and SaaS, such as Network dependence and browser based risks. It also has some specific issues, which are mentioned in the following diagram:



### **Compatibility with legacy security vulnerabilities**

Because IaaS offers the customer to run legacy software in provider's infrastructure, it exposes customers to all of the security vulnerabilities of such legacy software.

### Virtual Machine sprawl

The VM can become out-of-date with respect to security updates because IaaS allows the customer to operate the virtual machines in running, suspended and off state. However, the provider can automatically update such VMs, but this mechanism is hard and complex.

### **Robustness of VM-level isolation**

IaaS offers an isolated environment to individual customers through hypervisor. Hypervisor is a software layer that includes hardware support for virtualization to split a physical computer into multiple virtual machines.

### Data erase practices

The customer uses virtual machines that in turn use the common disk resources provided by the cloud provider. When the customer releases the resource, the cloud provider must ensure that next customer to rent the resource does not observe data residue from previous customer.

## Characteristics

Here are the characteristics of IaaS service model:

* Virtual machines with pre-installed software.
* Virtual machines with pre-installed operating systems such as Windows, Linux, and Solaris.
* On-demand availability of resources.
* Allows to store copies of data at different locations.
* The computing resources can be easily scaled up and down.

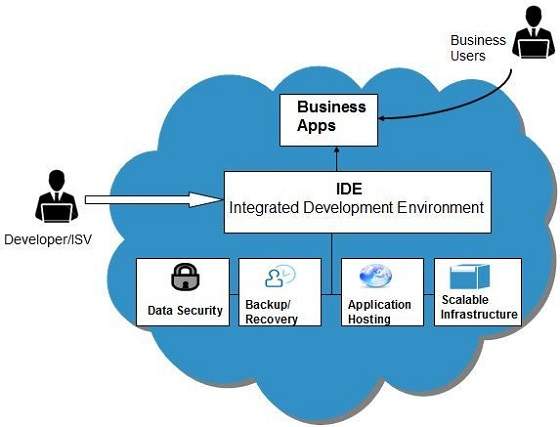
# **Cloud Computing Platform as a Service (PaaS)**

Platform-as-a-Service offers the runtime environment for applications. It also offers development and deployment tools required to develop applications. PaaS has a feature of point-and-click tools that enables non-developers to create web applications.

App Engine of Google and Force.com are examples of PaaS offering vendors. Developer may log on to these websites and use the built-in API to create web-based applications.

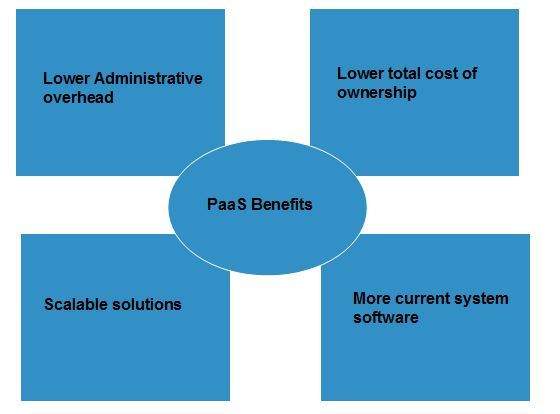
But the disadvantage of using PaaS is that, the developer locks-in with a particular vendor. For example, an application written in Python against API of Google, and using App Engine of Google is likely to work only in that environment.

The following diagram shows how PaaS offers an API and development tools to the developers and how it helps the end user to access business applications.



## Benefits

Following are the benefits of PaaS model:



### **Lower administrative overhead**

Customer need not bother about the administration because it is the responsibility of cloud provider.

### **Lower total cost of ownership**

Customer need not purchase expensive hardware, servers, power, and data storage.

### **Scalable solutions**

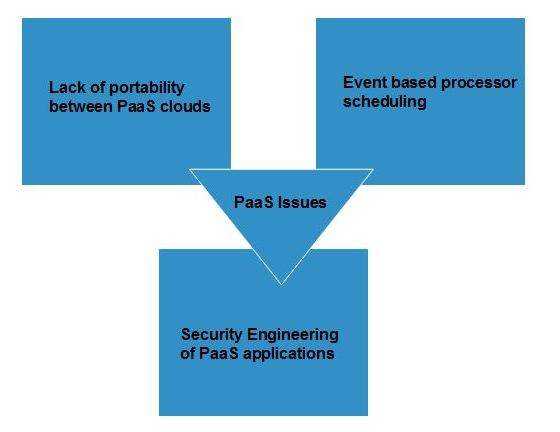
It is very easy to scale the resources up or down automatically, based on their demand.

### **More current system software**

It is the responsibility of the cloud provider to maintain software versions and patch installations.

## Issues

Like **SaaS, PaaS** also places significant burdens on customer's browsers to maintain reliable and secure connections to the provider’s systems. Therefore, PaaS shares many of the issues of SaaS. However, there are some specific issues associated with PaaS as shown in the following diagram:



### **Lack of portability between PaaS clouds**

Although standard languages are used, yet the implementations of platform services may vary. For example, file, queue, or hash table interfaces of one platform may differ from another, making it difficult to transfer the workloads from one platform to another.

### **Event based processor scheduling**

The PaaS applications are event-oriented which poses resource constraints on applications, i.e., they have to answer a request in a given interval of time.

### **Security engineering of PaaS applications**

Since PaaS applications are dependent on network, they must explicitly use cryptography and manage security exposures.

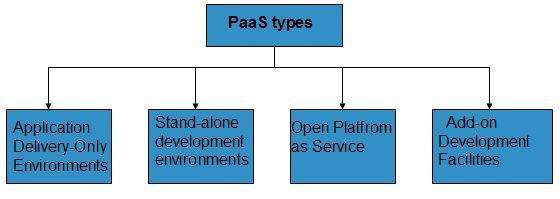
## Characteristics

Here are the characteristics of PaaS service model:

* PaaS offers browser-based development environment. It allows the developer to create database and edit the application code either via Application Programming Interface or point-and-click tools.
* PaaS provides built-in security, scalability, and web service interfaces.
* PaaS provides built-in tools for defining workflow, approval processes, and business rules.
* It is easy to integrate PaaS with other applications on the same platform.
* PaaS also provides web services interfaces that allow us to connect the applications outside the platform.

## PaaS Types

Based on the functions, PaaS can be classified into four types as shown in the following diagram:



### **Stand-alone development environments**

The stand-alone PaaS works as an independent entity for a specific function. It does not include licensing or technical dependencies on specific SaaS applications.

### **Application delivery-only environments**

The application delivery PaaS includes on-demand scaling and application security.

### **Open platform as a service**

Open PaaS offers an open source software that helps a PaaS provider to run applications.

### **Add-on development facilities**

The **add-on PaaS** allows to customize the existing SaaS platform.

# **Cloud Computing Software as a Service (SaaS)**

Software-as–a-Service (SaaS) model allows to provide software application as a service to the end users. It refers to a software that is deployed on a host service and is accessible via Internet. There are several SaaS applications listed below:

* Billing and invoicing system
* Customer Relationship Management (CRM) applications
* Help desk applications
* Human Resource (HR) solutions

Some of the SaaS applications are not customizable such as Microsoft Office Suite. But SaaS provides us Application Programming Interface (API), which allows the developer to develop a customized application.

## Characteristics

Here are the characteristics of SaaS service model:

* SaaS makes the software available over the Internet.
* The software applications are maintained by the vendor.
* The license to the software may be subscription based or usage based. And it is billed on recurring basis.
* SaaS applications are cost-effective since they do not require any maintenance at end user side.
* They are available on demand.
* They can be scaled up or down on demand.
* They are automatically upgraded and updated.
* SaaS offers shared data model. Therefore, multiple users can share single instance of infrastructure. It is not required to hard code the functionality for individual users.
* All users run the same version of the software.

## Benefits

Using SaaS has proved to be beneficial in terms of scalability, efficiency and performance. Some of the benefits are listed below:

* Modest software tools
* Efficient use of software licenses
* Centralized management and data
* Platform responsibilities managed by provider
* Multitenant solutions

### Modest software tools

The SaaS application deployment requires a little or no client side software installation, which results in the following benefits:

* No requirement for complex software packages at client side
* Little or no risk of configuration at client side
* Low distribution cost

### **Efficient use of software licenses**

The customer can have single license for multiple computers running at different locations which reduces the licensing cost. Also, there is no requirement for license servers because the software runs in the provider's infrastructure.

### Centralized management and data

The cloud provider stores data centrally. However, the cloud providers may store data in a decentralized manner for the sake of redundancy and reliability.

### Platform responsibilities managed by providers

All platform responsibilities such as backups, system maintenance, security, hardware refresh, power management, etc. are performed by the cloud provider. The customer does not need to bother about them.

### Multitenant solutions

Multitenant solutions allow multiple users to share single instance of different resources in virtual isolation. Customers can customize their application without affecting the core functionality.

## Issues

There are several issues associated with SaaS, some of them are listed below:

* Browser based risks
* Network dependence
* Lack of portability between SaaS clouds

### **Browser based risks**

If the customer visits malicious website and browser becomes infected, the subsequent access to SaaS application might compromise the customer's data.

To avoid such risks, the customer can use multiple browsers and dedicate a specific browser to access SaaS applications or can use virtual desktop while accessing the SaaS applications.

### **Network dependence**

The SaaS application can be delivered only when network is continuously available. Also network should be reliable but the network reliability cannot be guaranteed either by cloud provider or by the customer.

### **Lack of portability between SaaS clouds**

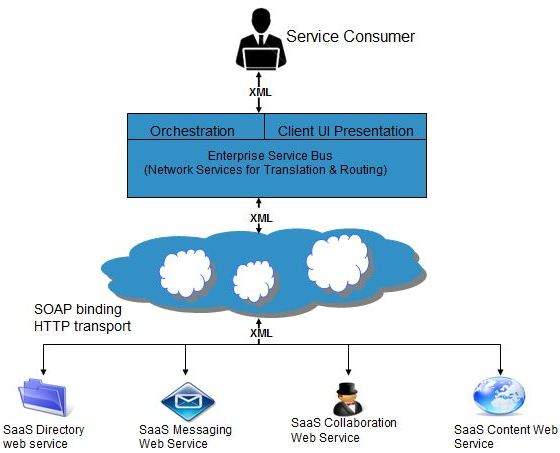
Transferring workloads from one SaaS cloud to another is not so easy because workflow, business logics, user interfaces, support scripts can be provider specific.

## Open SaaS and SOA

Open SaaS uses those SaaS applications, which are developed using open source programming language. These SaaS applications can run on any open source operating system and database. Open SaaS has several benefits listed below:

* No License Required
* Low Deployment Cost
* Less Vendor Lock-in
* More portable applications
* More Robust Solution

The following diagram shows the SaaS implementation based on SOA:



# **Cloud Computing Identity as a Service (IDaaS)**

Employees in a company require to login to system to perform various tasks. These systems may be based on local server or cloud based. Following are the problems that an employee might face:

* Remembering different username and password combinations for accessing multiple servers.
* If an employee leaves the company, it is required to ensure that each account of that user is disabled. This increases workload on IT staff.

To solve above problems, a new technique emerged which is known as Identity-as–a-Service (IDaaS).

IDaaS offers management of identity information as a digital entity. This identity can be used during electronic transactions.

## Identity

Identity refers to set of attributes associated with something to make it recognizable. All objects may have same attributes, but their identities cannot be the same. A unique identity is assigned through unique identification attribute.

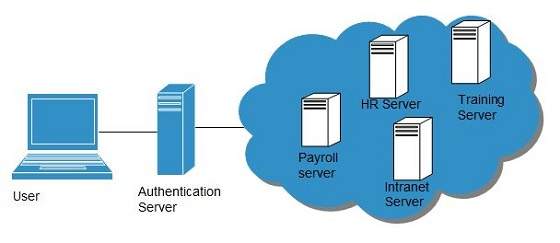
There are several identity services that are deployed to validate services such as validating web sites, transactions, transaction participants, client, etc. Identity-as-a-Service may include the following:

* Directory services
* Federated services
* Registration
* Authentication services
* Risk and event monitoring
* Single sign-on services
* Identity and profile management

## Single Sign-On (SSO)

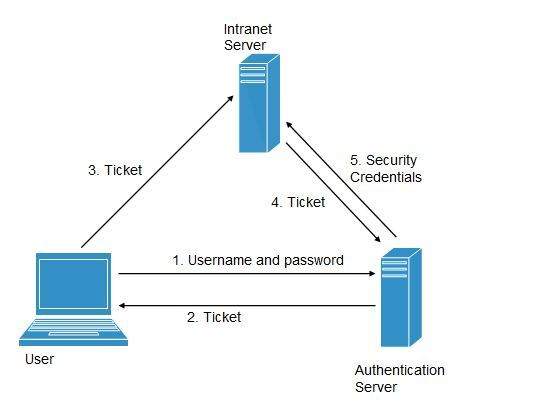
To solve the problem of using different username and password combinations for different servers, companies now employ Single Sign-On software, which allows the user to login only one time and manage the access to other systems.

SSO has single authentication server, managing multiple accesses to other systems, as shown in the following diagram:



### **SSO Working**

There are several implementations of SSO. Here, we discuss the common ones:



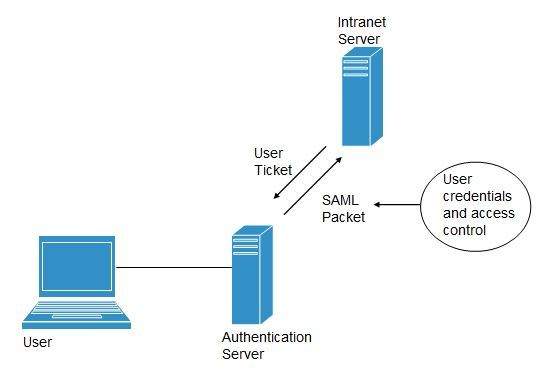
Following steps explain the working of Single Sign-On software:

* User logs into the authentication server using a username and password.
* The authentication server returns the user's ticket.
* User sends the ticket to intranet server.
* Intranet server sends the ticket to the authentication server.
* Authentication server sends the user's security credentials for that server back to the intranet server.

If an employee leaves the company, then disabling the user account at the authentication server prohibits the user's access to all the systems.

## Federated Identity Management (FIDM)

FIDM describes the technologies and protocols that enable a user to package security credentials across security domains. It uses Security Markup Language (SAML) to package a user's security credentials as shown in the following diagram:



## OpenID

It offers users to login into multiple websites with single account. Google, Yahoo!, Flickr, MySpace, WordPress.com are some of the companies that support OpenID.

## Benefits

* Increased site conversation rates
* Access to greater user profile content
* Fewer problems with lost passwords
* Ease of content integration into social networking sites

# **Cloud Computing Network as a Service (NaaS)**

Network-as-a-Service allows us to access to network infrastructure directly and securely. NaaS makes it possible to deploy custom routing protocols.

NaaS uses virtualized network infrastructure to provide network services to the customer. It is the responsibility of NaaS provider to maintain and manage the network resources. Having a provider working for a customer decreases the workload of the customer. Moreover, NaaS offers network as a utility. NaaS is also based on pay-per-use model.

## How NaaS is delivered

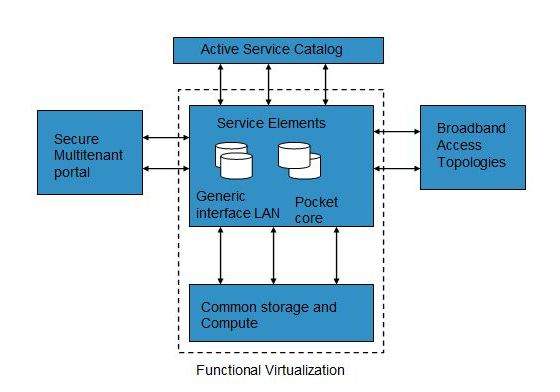
To use NaaS model, the customer is required to logon to the web portal, where he can get online API. Here, the customer can customize the route.

In turn, customer has to pay for the capacity used. It is also possible to turn off the capacity at any time.

## Mobile NaaS

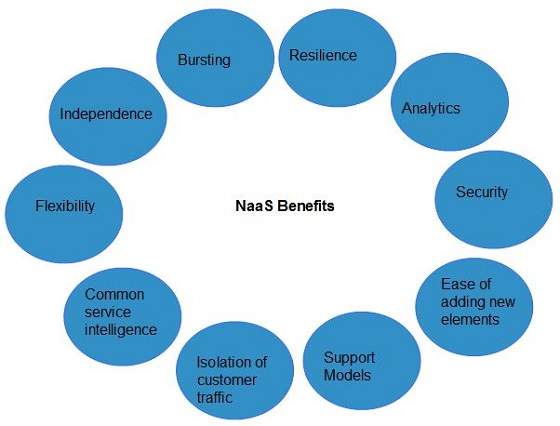
Mobile NaaS offers more efficient and flexible control over mobile devices. It uses virtualization to simplify the architecture thereby creating more efficient processes.

Following diagram shows the Mobile NaaS service elements:



## NaaS Benefits

NaaS offers a number of benefits as discussed below:



### **Independence**

Each customer is independent and can segregate the network.

### **Bursting**

The customer pays for high-capacity network only on requirement.

### **Resilience**

The reliability treatments are available, which can be applied for critical applications.

### **Analytics**

The data protection solutions are available, which can be applied for highly sensitive applications.

### Ease of Adding New Service Elements

It is very easy to integrate new service elements to the network.

### **Support Models**

A number of support models are available to reduce operation cost.

### **Isolation of Customer Traffic**

The customer traffic is logically isolated.

# **Cloud Computing Data Storage**

Cloud Storage is a service that allows to save data on offsite storage system managed by third-party and is made accessible by a **web services API.**

## Storage Devices

Storage devices can be broadly classified into two categories:

* Block Storage Devices
* File Storage Devices

### **Block Storage Devices**

The block storage devices offer raw storage to the clients. These raw storage are partitioned to create volumes.

### **File Storage Devices**

The file Storage Devices offer storage to clients in the form of files, maintaining its own file system. This storage is in the form of Network Attached Storage (NAS).

## Cloud Storage Classes

Cloud storage can be broadly classified into two categories:

* Unmanaged Cloud Storage
* Managed Cloud Storage

### Unmanaged Cloud Storage

Unmanaged cloud storage means the storage is preconfigured for the customer. The customer can neither format, nor install his own file system or change drive properties.

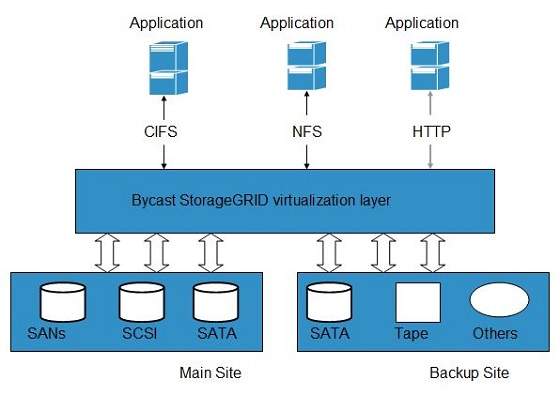
### Managed Cloud Storage

Managed cloud storage offers online storage space on-demand. The managed cloud storage system appears to the user to be a raw disk that the user can partition and format.

## Creating Cloud Storage System

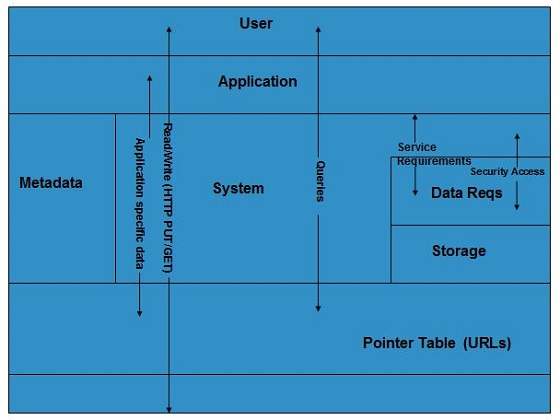
The cloud storage system stores multiple copies of data on multiple servers, at multiple locations. If one system fails, then it is required only to change the pointer to the location, where the object is stored.

To aggregate the storage assets into cloud storage systems, the cloud provider can use storage virtualization software known as **StorageGRID.** It creates a virtualization layer that fetches storage from different storage devices into a single management system. It can also manage data from **CIFS** and **NFS** file systems over the Internet. The following diagram shows how StorageGRID virtualizes the storage into storage clouds:



## Virtual Storage Containers

The virtual storage containers offer high performance cloud storage systems. Logical Unit Number (LUN) of device, files and other objects are created in virtual storage containers. Following diagram shows a virtual storage container, defining a cloud storage domain:



## Challenges

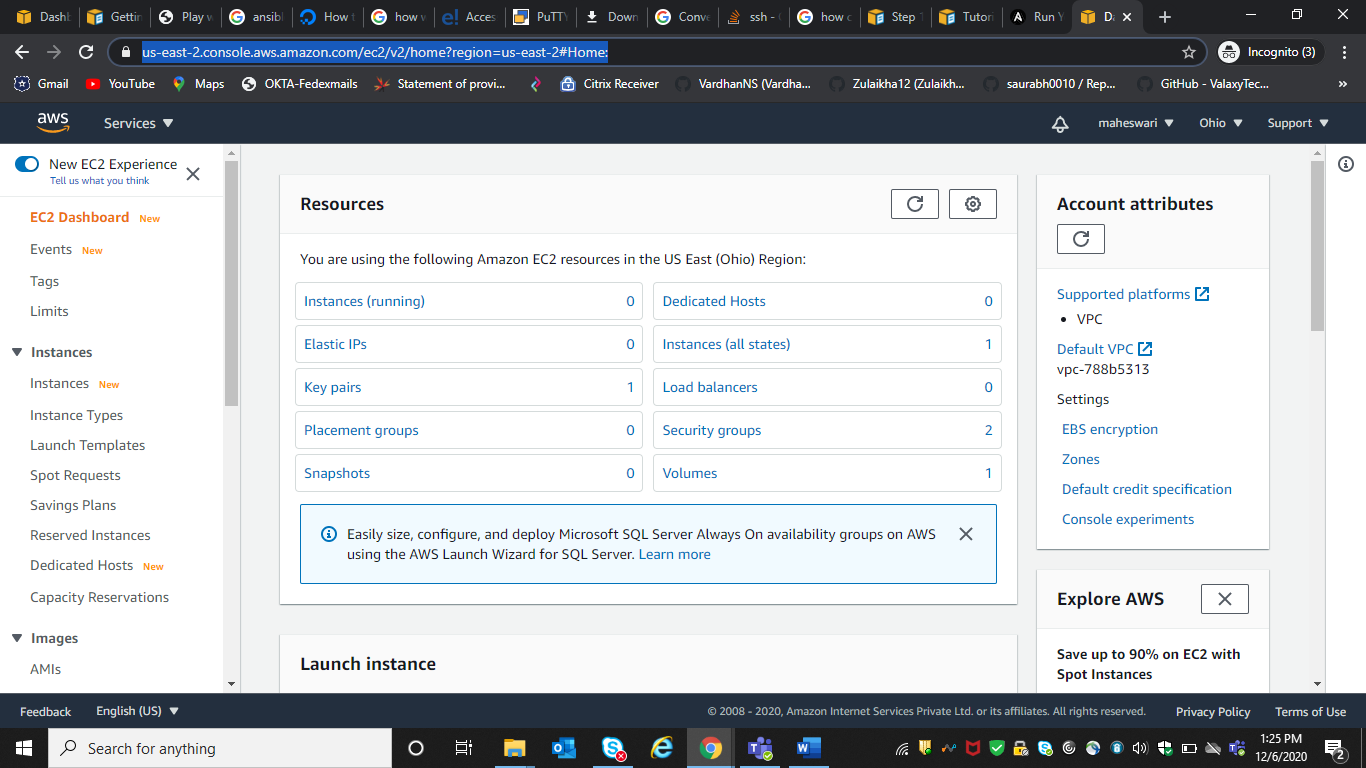
Storing the data in cloud is not that simple task. Apart from its flexibility and convenience, it also has several challenges faced by the customers. The customers must be able to:

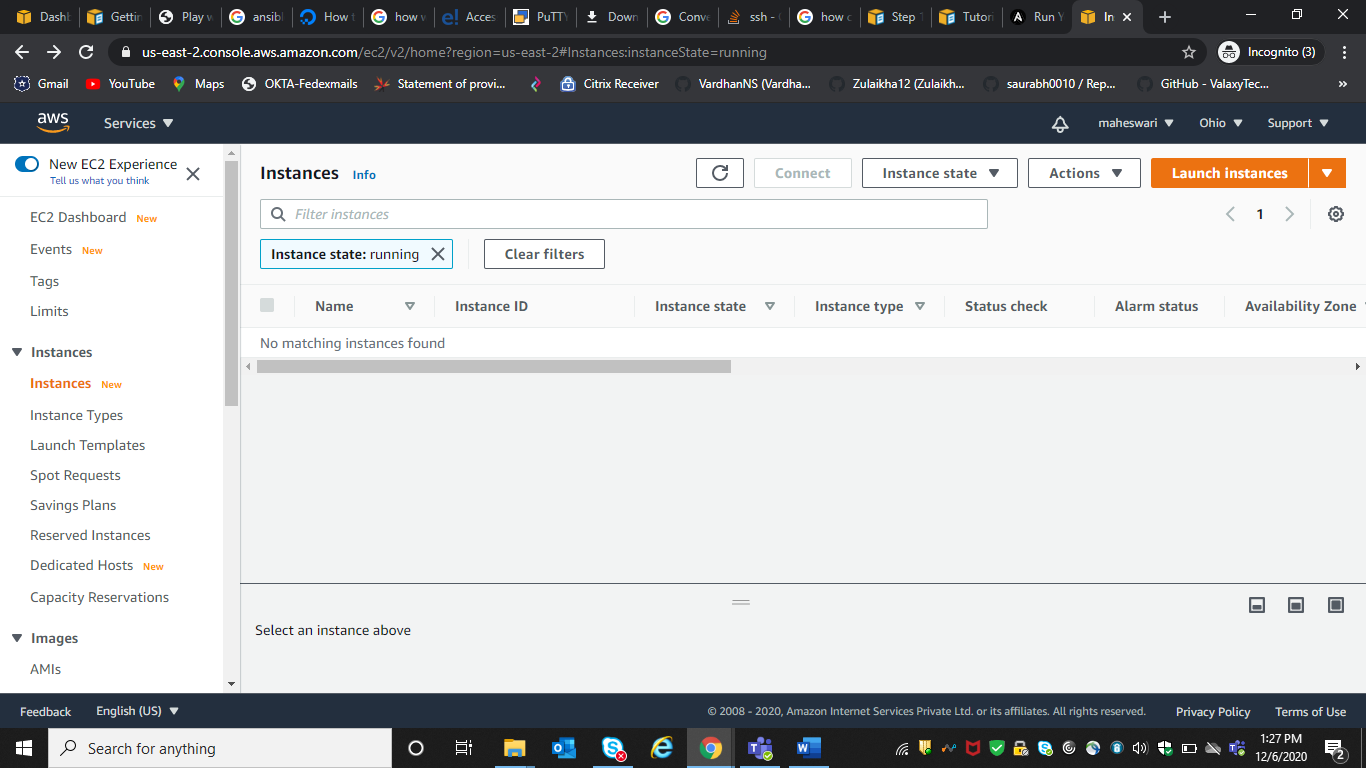
* Get provision for additional storage on-demand.
* Know and restrict the physical location of the stored data.
* Verify how data was erased.
* Have access to a documented process for disposing of data storage hardware.
* Have administrator access control over data.

**Creating EC2 Instance:**

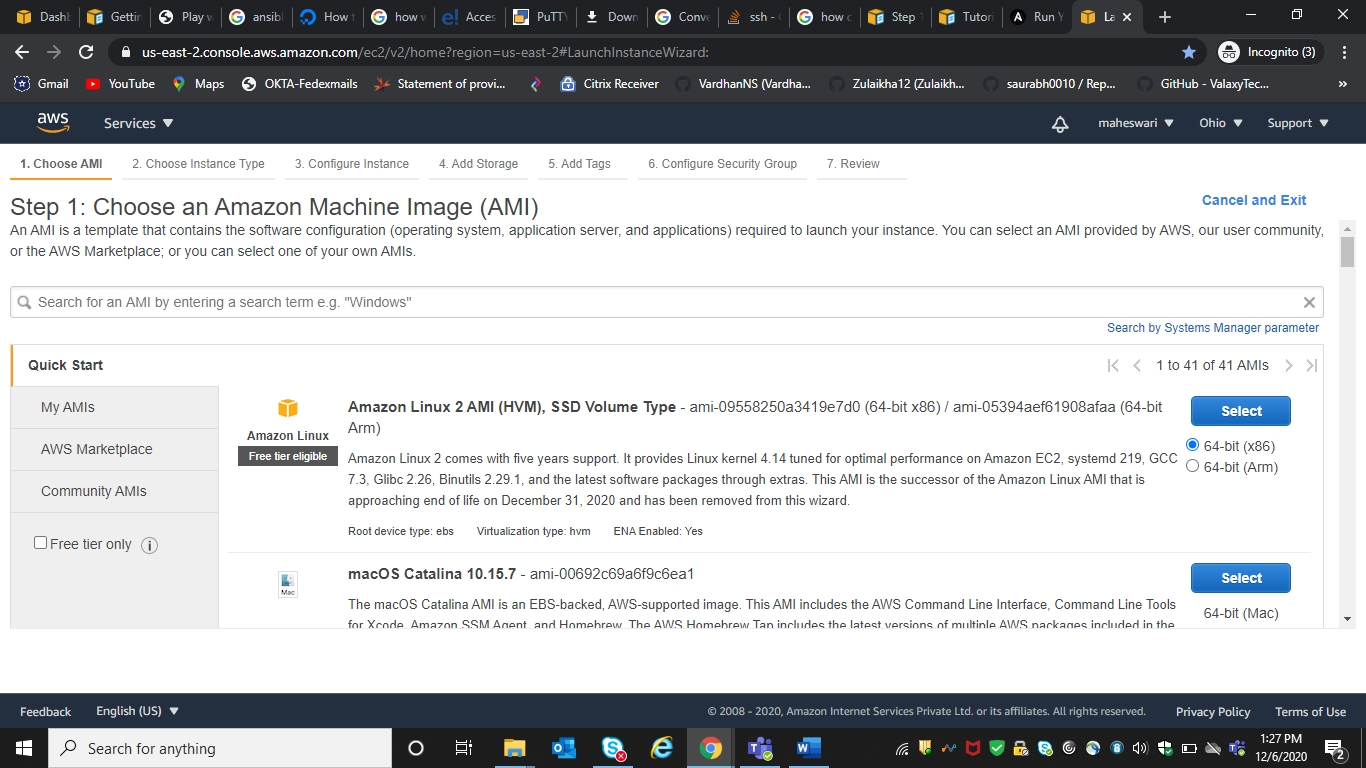
1.Go to the amazon console,

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

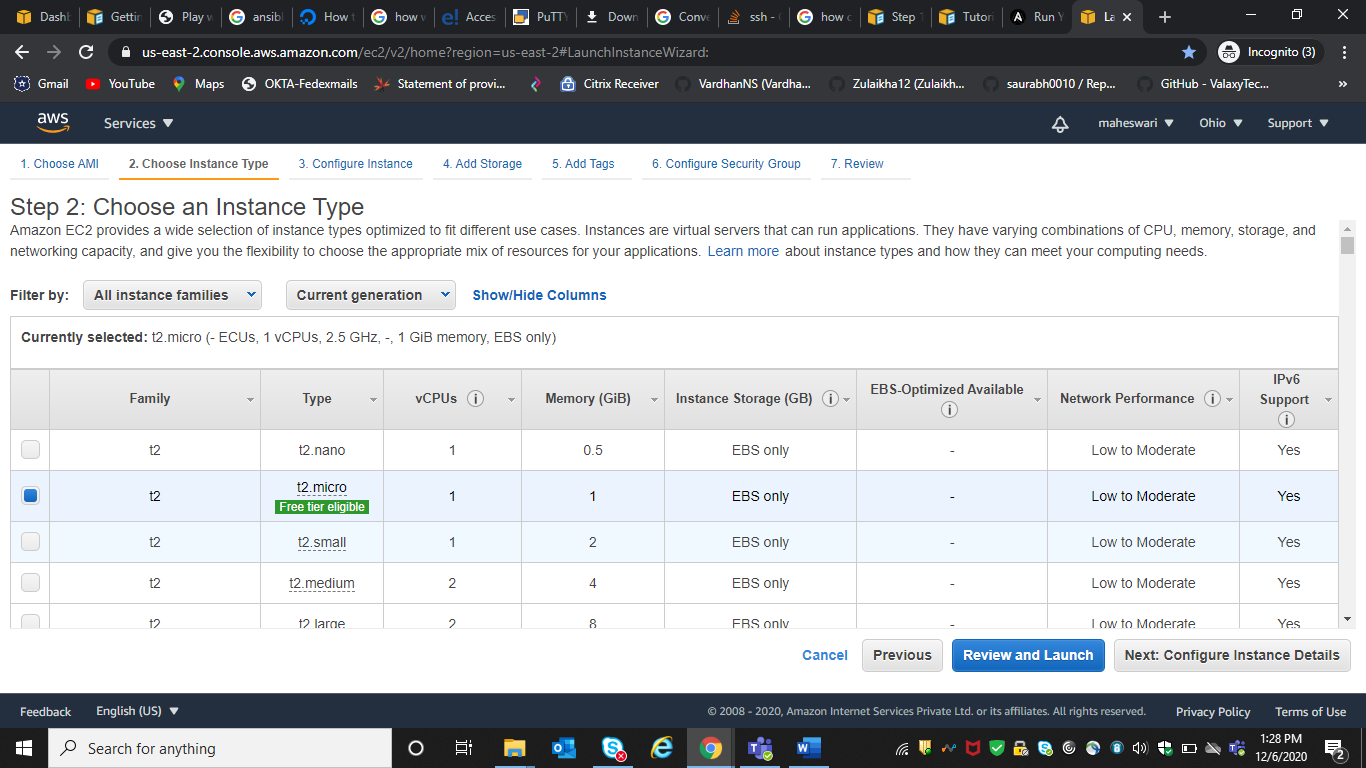
2. 3.Launch instances



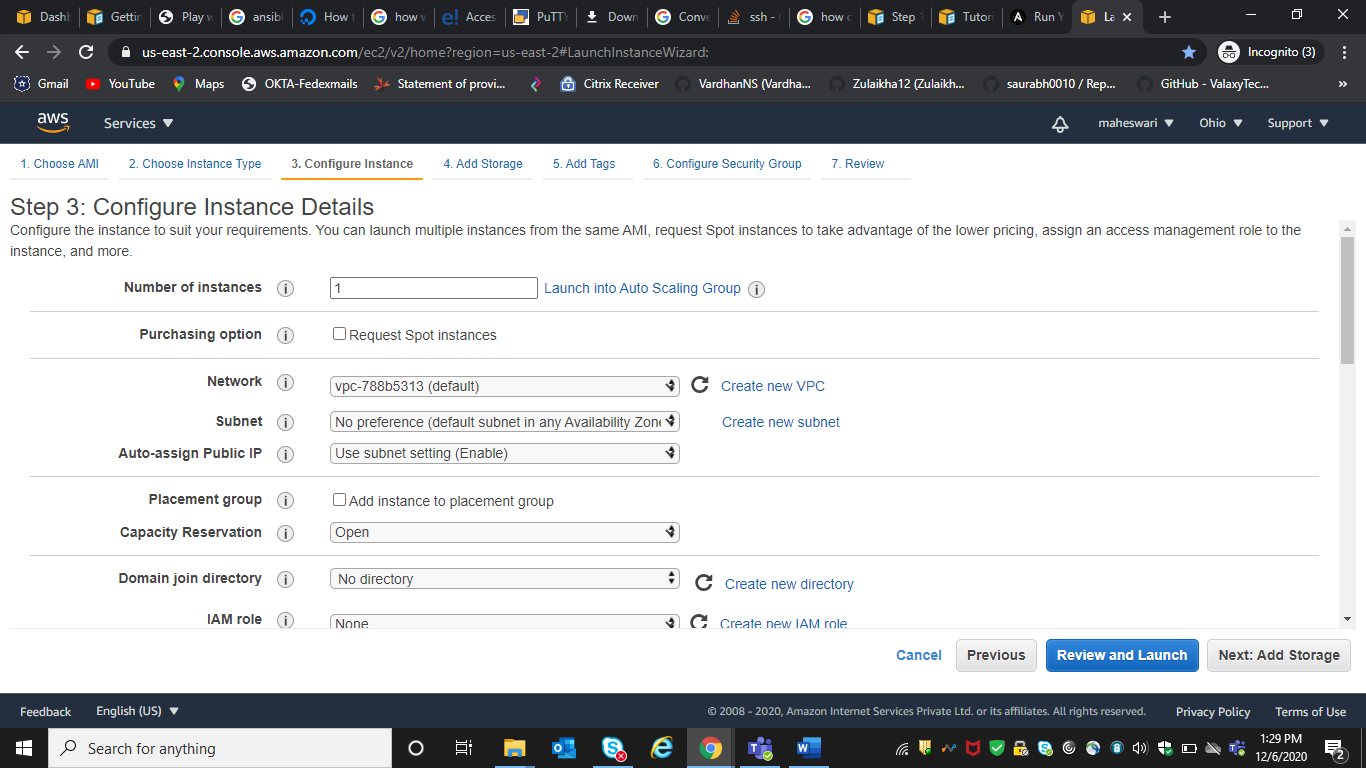
4.select AMI



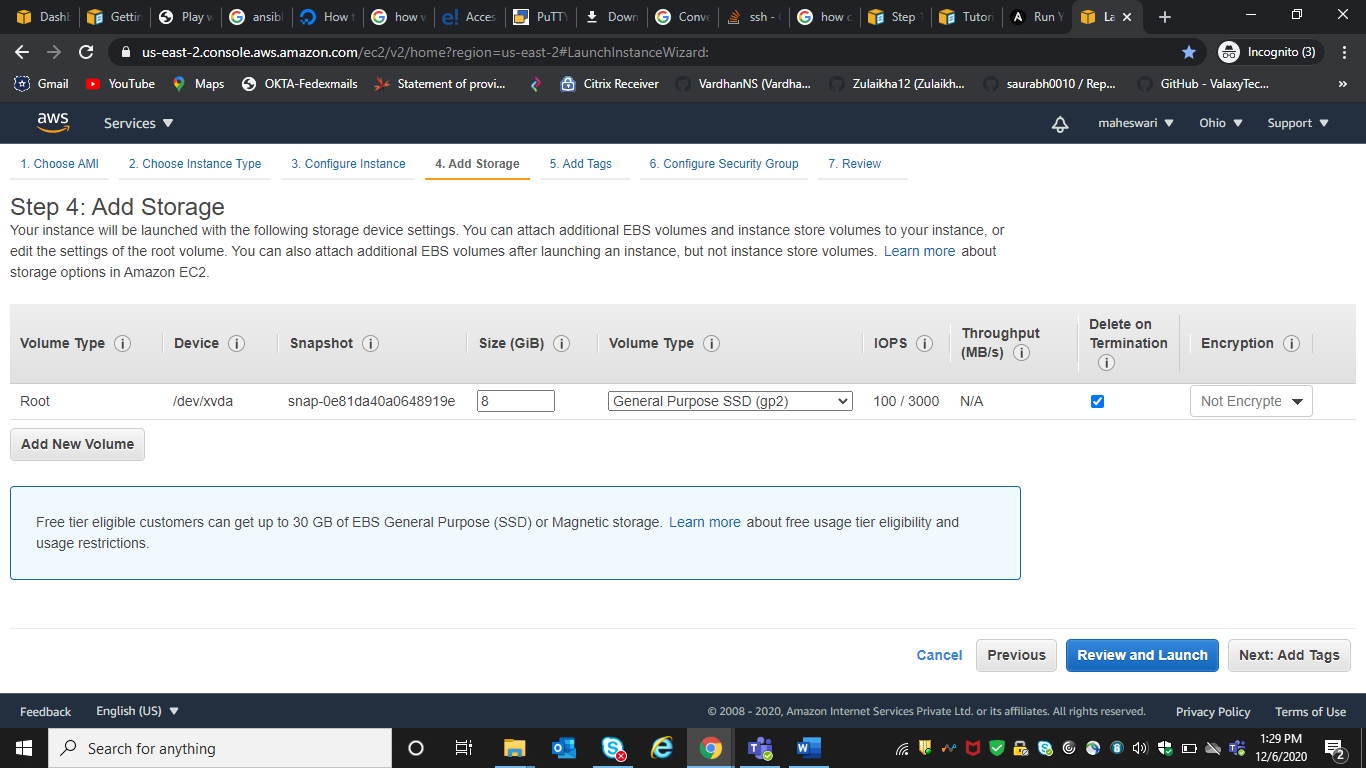
5.Instance type



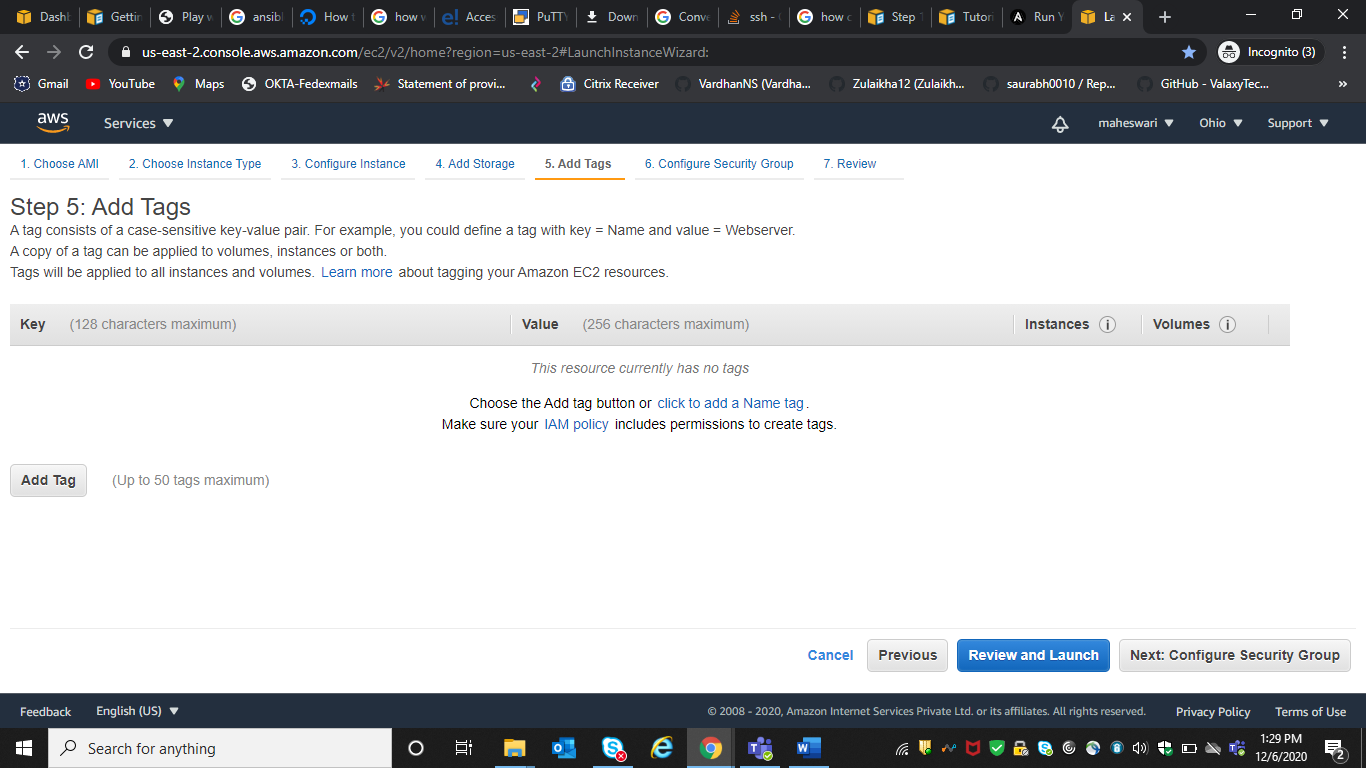
6. configurations of the instance



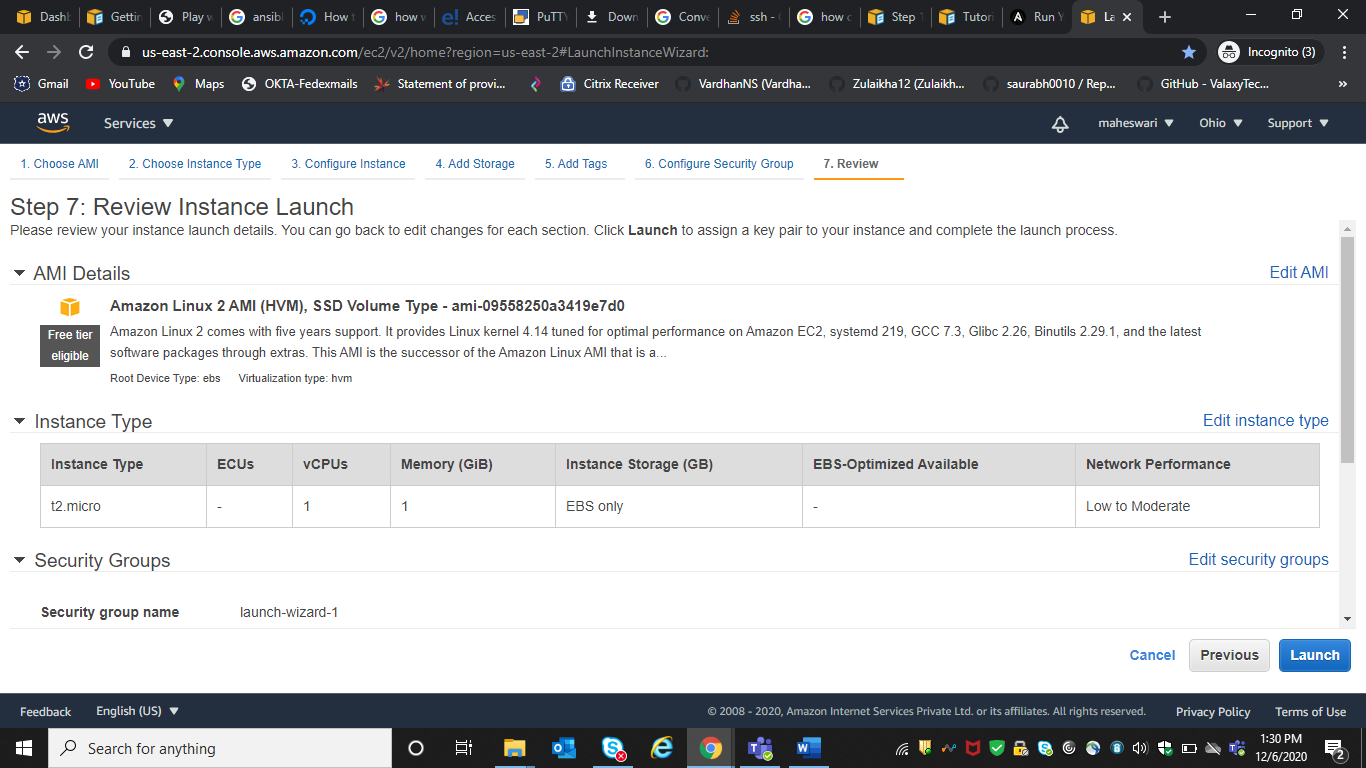
7.Storage



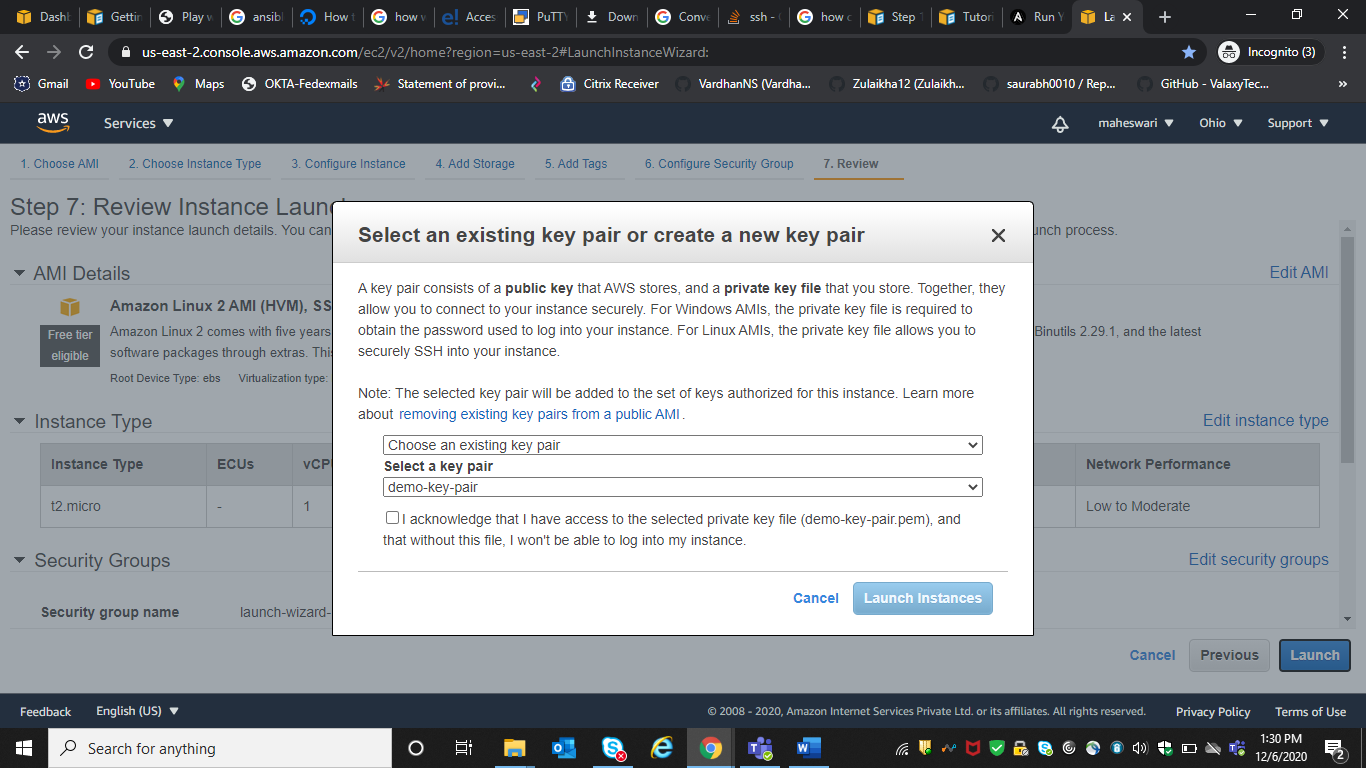
8. add tags if required



9.review and launch



10. use can use existing key value pair for connection



## Connect to your Amazon EC2 instance

After your Amazon EC2 instance is launched, follow these instructions to practice connecting to it.

Note

In these instructions, we assume you are running Windows and the Windows Desktop Connection client application. For information, see [Connecting to your Windows instance using RDP](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/connecting_to_windows_instance.html). You might need to adapt these instructions for other operating systems or other RDP connection client applications.

1. Sign in to the AWS Management Console and open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. In the navigation pane, under Instances, choose Instances.
3. Browse to and choose your Windows Server instance in the list.
4. Choose Connect.
5. Choose Get Password, and then choose Choose File.
6. Browse to and choose the Amazon EC2 instance key pair file associated with the Windows Server Amazon EC2 instance, and then choose Open.
7. Choose Decrypt Password. Make a note of the password that is displayed. You need it in step 10.
8. Choose Download Remote Desktop File, and then open the file.
9. If you are prompted to connect even though the publisher of the remote connection can't be identified, proceed.
10. Type the password you noted in step 7, and then proceed. (If your RDP connection client application prompts you for a user name, type Administrator.)
11. If you are prompted to connect even though the identity of the remote computer cannot be verified, proceed.
12. After you are connected, the desktop of the Amazon EC2 instance running Windows Server is displayed.
13. You can now disconnect from the Amazon EC2 instance.

Warning

Do not stop or terminate the instance. Otherwise, CodeDeploy can't deploy to it.

## Add an inbound rule that allows HTTP traffic to your Windows Server Amazon EC2 instance

The next step confirms your Amazon EC2 instance has an open HTTP port so you can see the deployed webpage on your Windows Server Amazon EC2 instance in a browser.

1. Sign in to the AWS Management Console and open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. Choose **Instances**, and then choose your instance.
3. On the **Description** tab, under **Security groups**, choose **view inbound rules**.

You should see a list of rules in your security group like the following:

Security Groups associated with i-1234567890abcdef0

Ports Protocol Source launch-wizard-*N*

22 tcp 0.0.0.0/0 ✔

1. Under **Security groups**, choose the security group for your Amazon EC2 instance. It might be named **launch-wizard-***N*. The *N* in the name is a number assigned to your security group when your instance was created.

Choose the **Inbound** tab. If the security group for your instance is configured correctly, you should see a rule with the following values:

* + **Type**: HTTP
  + **Protocol**: TCP
  + **Port Range**: 80
  + **Source**: 0.0.0.0/0

1. If you do not see a rule with these values, use the procedures in [Adding Rules to a Security Group](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-network-security.html#adding-security-group-rule) to add them to a new security rule.

# **Ansible Installation in Linux**

When you have compared and weighed your options and decided to go for Ansible. Then installed it on your system. Let's go step by step of the installation in different Linux distributions, such as:

## Prerequisites

1. PyYAML: a YAML parser and emitter for the python programming language.
2. Httplib2: a comprehensive HTTP client library.
3. parmiko: native python SSHv2 protocol library.
4. Distro: RHEL/ CentOS/ Debian/ Ubuntu Linux.
5. Jinja2: a modern and designer friendly templating language for python.
6. sshpass: a non-interactive ssh password authentication.

## Install Ansible on RedHat/Centos systems

**Step 1:** Install the EPEL repo

1. [root@ansible-server ~]# sudo yum install epel-release

**Step 2:** Install the Ansible package.

1. [root@ansible-server ~]# sudo yum install -y ansible

## Install Ansible on Debian/Ubuntu systems

**Step 1:** First perform an update to the packages

1. $ sudo apt update

**Step 2:** Then install the software properties common package.

1. $ sudo apt install software-properties-common

**Step 3:** And install the Ansible personal package archive.

1. $ sudo apt-add-repository ppa:ansible/ansible

**Step 4:** Install the Ansible.

1. $ sudo apt update
2. $ sudo apt install ansible

## Install Ansible using pip

The pip command is a tool for installing and managing python packages.

**Step 1:** This given method works on the Linux and UNIX like systems.

1. $ sudo pip install ansible