Dept: Computer Engineering		
Subject: Cloud Computing Lab	**Subject Code:** CSL605	
Year/Semester: TE-VI	**Date:**	**Page No.**
Student Name:	**Roll No.**	**Division:**

Experiment No.: 1

Aim: Introduction and overview of cloud computing

Software Used: Ms Word

Theory:
What is Cloud Computing?
Cloud Computing can be defined as delivering computing power(CPU, RAM, Network Speeds, Storage OS software) a service over a network (usually on the internet) rather than physically having the computing resources at the customer location.
Example: AWS, Azure, Google Cloud

Let's learn Cloud computing with an example -
Whenever you travel through a bus or train, you take a ticket for your destination and hold back to your seat till you reach your destination. Likewise other passengers also takes ticket and travel in the same bus with you and it hardly bothers you where they go. When your stop comes you get off the bus thanking the driver. Cloud computing is just like that bus, carrying data and information for different users and allows to use its service with minimal cost.

Why Cloud Computing?

With increase in computer and Mobile user's, data storage has become a priority in all fields. Large and small scale businesses today thrive on their data & they spent a huge amount of money to maintain this data. It requires a strong IT support and a storage hub. Not all businesses can afford high cost of in-house IT infrastructure and back up support services. For them Cloud Computing is a cheaper solution. Perhaps its efficiency in storing data, computation and less maintenance cost has succeeded to attract even bigger businesses as well.
Cloud computing decreases the hardware and software demand from the user's side. The only thing that user must be able to run is the cloud computing systems interface software, which can be as simple as Web browser, and the Cloud network takes care of the rest. We all have experienced cloud computing at some instant of time, some of the popular cloud services we have used or we are still using are mail services like gmail, hotmail or yahoo etc.

History of Cloud Computing

Before Computing was come into existence, client Server Architecture was used where all the data and control of client resides in Server side. If a single user want to access some data, firstly user need to connect to the server and after that user will get appropriate access. But it has many disadvantages. So, After Client Server computing, Distributed Computing was come into existence, in this type of computing all computers are networked together with the help of this, user can share their resources when needed. It also has certain limitations. So in order to remove limitations faced in distributed system, cloud computing was emerged.

During 1961, John MacCharty delivered his speech at MIT that "Computing Can be sold as a Utility, like Water and Electricity." According to John MacCharty it was a brilliant idea. But people at that time don't want to adopt this technology. They thought the technology they are using efficient enough for them. So, this concept of computing was not appreciated much so and very less will research on it. But as the time fleet the technology caught the idea after few years this idea is implemented. So, this is implemented by Salesforce.com in 1999.
This company started delivering an enterprise application over the internet and this way the boom of Cloud Computing was started.

In 2002, Amazon started Amazon Web Services (AWS), Amazon will provide storage, computation over the internet. In 2006 Amazon will launch Elastic Compute Cloud Commercial Service which is open for Everybody to use.

After that in 2009, Google Play also started providing Cloud Computing Enterprise Application as other companies will see the emergence of cloud Computing they also started providing their cloud services. Thus, in 2009, Microsoft launch Microsoft Azure and after that other companies like Alibaba, IBM, Oracle, HP also introduces their Cloud Services. In today the Cloud Computing become very popular and important skill.

Cloud Cube Model

The Cloud cube model helps to classified the network of cloud-based on the four-dimensional factor. The main motive of the cloud model is to secure and protect the cloud network. The cloud model supports to choose cloud creation for the security association. It also helps IT managers, organizations, and business leaders by offering a safe and protected network. Security is an essential aspect for cloud users, and most of the cloud providers understand it. The customer should also take care of that; the selected cloud formation fulfills the regulatory and location needs. They also need one thing in their mind that if cloud providers stop offering the services, where else they can move.

How data is secured in the cloud cube model?

There are various steps and point you should keep on your mind while securing your data in a cloud cube model.
1. The categorization of the data, the user must know what rules must be applied to secure and protect it.
2. It should make sure; that the data exist only in particular trust levels.
3. It should examine that what regulatory compliance and constraints are applicable.

For example: – The data must keep in a specific limit and whether it has to stay in the secure harbor or not.

When the data is categorized and can put in the needed zone, the assigned person is in a position to decide the following aspects-

- The processes and data, which are to be shift in the cloud.
- At what level the customer wants to operate in the cloud. Maybe it is infrastructure, platform, and software.
- The cloud formations, fulfill the requirements.
- In a cloud, the level of operation can be different as per the requirement.

Dimensions of Cloud Cube Model

Jericho Forum has pointed out multiple cloud service providers, and they claim to provide solutions. Thus, the customers using Cloud Computing require help in choosing the correct data within CCM which is suitable for their requirements. In CCM, there are four various dimensions.

The dimensions are as follows:

1. External and internal
2. Proprietary and Open
3. Perimeterized or known as (Per) and De-Parameterized or (D-p)
4. In Sourced and Outsourced

1. Internal and External

The internal term is allocated for the Private Cloud, and External is earmarked for the Public Cloud.

2. Proprietary and Open

Proprietary stands for paid services, or it means for contractors. The term Open is allocated for Open-Source solutions or services. In some cases, if we talk about Cloud Computing, the open term is used for a system or a platform that permits sharing and freely accessing the APIs. Google App Engine is a famous example to be considered as Open.

3. Parameterized (Per) and De-Perimeterized (D-p)

The real-world definition of Per and D-p in the architectural point of view means whether traditional IT resources like networks and firewalls are operating inside the Per or outside the D-p of the organization.

In Cloud, computing parameterized is referred to Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) and any other services or contracts which helps in supporting the infrastructure and platform.

De-parametrized is the term that refers to Software-as-a-Service (SaaS) or to any other set of services followed by the contracts and supports. It is required for software or application as they are not stuck or restricted with any hardware limitations.

4. Insourced and Outsourced

Insources term refers to the in-house development of the cloud, and the Outsource term refers to allowing contractors and service providers to handle all queries. Mostly all the Cloud Business Models come under this section.

NIST Model

Visual Model of NIST's Working Definition of Cloud Computing
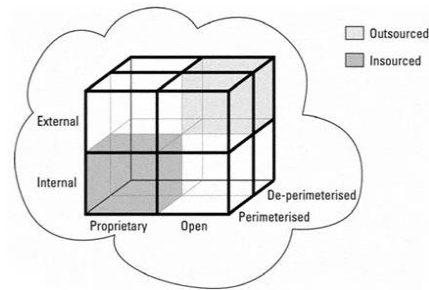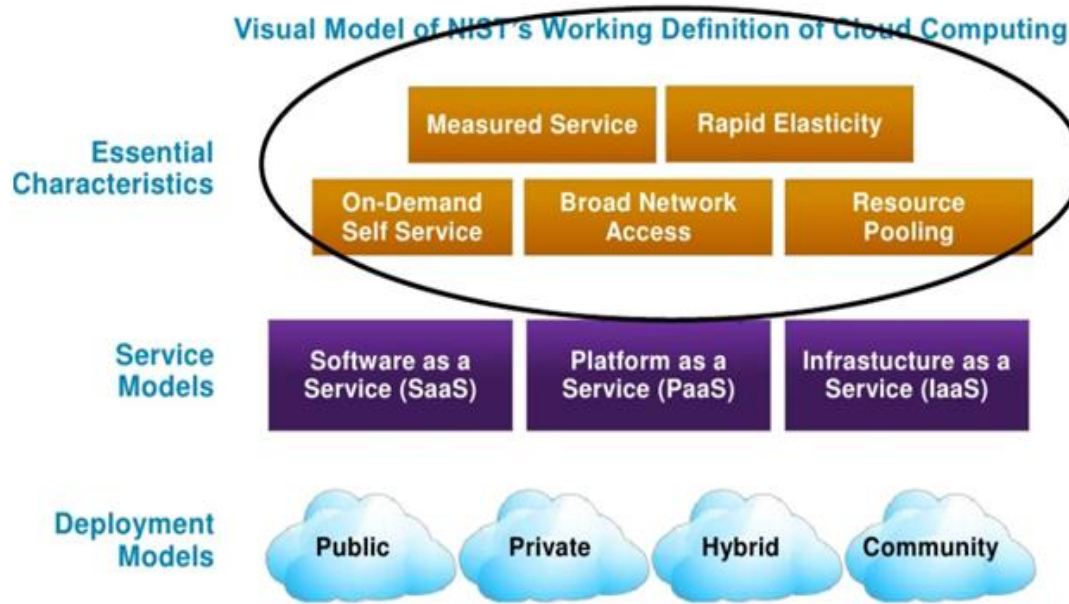
Benefits of Cloud Computing

The potential for cost saving is the major reason of cloud services adoption by many organizations. Cloud computing gives the freedom to use services as per the requirement and pay only for what you use. Due to cloud computing it has become possible to run IT operations as a outsourced unit without much in-house resources.

Following are the benefits of cloud computing:

1. Lower IT infrastructure and computer costs for users
2. Improved performance
3. Fewer Maintenance issues
4. Instant software updates
5. Improved compatibility between Operating systems
6. Backup and recovery
7. Performance and Scalability
8. Increased storage capacity
9. Increase data safety

Characteristics of cloud:

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

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

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

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

Measured service. Cloud systems automatically control and optimize resource use by leveraging a metering capability10 at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, active user accounts). Resource usage can be monitored, controlled, audited, and reported, providing transparency for both the provider and consumer of the utilized service.

Deployment Models

The cloud deployment model identifies the specific type of cloud environment based on ownership, scale, and access, as well as the cloud's nature and purpose. The location of the servers you're utilizing and who controls them are defined by a cloud deployment model. It specifies how your cloud infrastructure will look, what you can change, and whether you will be given services or will have to create everything yourself. Relationships between the infrastructure and your users are also defined by cloud deployment types.

Different types of cloud computing deployment models are:

1. Public cloud
2. Private cloud
3. Hybrid cloud
4. Community cloud
5. Multi-cloud

1. Public Cloud

The public cloud makes it possible for anybody to access systems and services. The public cloud may be less secure as it is open for everyone. The public cloud is one in which cloud infrastructure services are provided over the internet to the general people or major industry groups. The infrastructure in this cloud model is owned by the entity that delivers the cloud services, not by the consumer. It is a type of cloud hosting that allows customers and users to easily access systems and services. This form of cloud computing is an excellent example of cloud hosting, in which service providers supply services to a variety of customers. In this arrangement, storage backup and retrieval services are given for free, as a subscription, or on a per-use basis. Example: Google App Engine etc.

Advantages of the public cloud model:

- **Minimal Investment:** Because it is a pay-per-use service, there is no substantial upfront fee, making it excellent for enterprises that require immediate access to resources.
- **No setup cost:** The entire infrastructure is fully subsidized by the cloud service providers, thus there is no need to set up any hardware.
- **Infrastructure Management is not required:** Using the public cloud does not necessitate infrastructure management.
- **No maintenance:** The maintenance work is done by the service provider (Not users).
- **Dynamic Scalability:** To fulfill your company's needs, on-demand resources are accessible.

2. Private Cloud

The private cloud deployment model is the exact opposite of the public cloud deployment model. It's a one-on-one environment for a single user (customer). There is no need to share your hardware with anyone else. The distinction between private and public cloud is in how you handle all of the hardware. It is also called the "internal cloud" & it refers to the ability to access systems and services within a given border or organization. The cloud platform is implemented in a cloud-based secure environment that is protected by powerful firewalls and underthe supervision   of   an   organization's   IT   department. The private cloud gives the greater flexibility of control over cloud resources.

Advantages of the private cloud model:

- **Better Control:** You are the sole owner of the property. You gain complete command over service integration, IT operations, policies, and user behavior.
- **Data Security and Privacy:** It's suitable for storing corporate information to which only authorized staff have access. By segmenting resources within the same infrastructure, improved access and security can be achieved.
- **Supports Legacy Systems:** This approach is designed to work with legacy systems that are unable to access the public cloud.
- **Customization:** Unlike a public cloud deployment, a private cloud allows a company to tailor its solution to meet its specific needs.

3. Hybrid cloud

By bridging the public and private worlds with a layer of proprietary software, hybrid cloud computing gives the best of both worlds. With a hybrid solution, you may host the app in a safe environment while taking advantage of the public cloud's cost savings. Organizations can move data and applications between different clouds using a combination of two or more cloud deployment methods, depending on their needs.

Advantages of the hybrid cloud model:

- **Flexibility and control:** Businesses with more flexibility can design personalized solutions that meet their particular needs.
- **Cost:** Because public clouds provide for scalability, you'll only be responsible for paying for the extra capacity if you require it.
- **Security:** Because data is properly separated, the chances of data theft by attackers are considerably reduced.

4. Community cloud

It allows systems and services to be accessible by a group of organizations. It is a distributed system that is created by integrating the services of different clouds to address the specific needs of a community, industry, or business. The infrastructure of the community could be shared between the organization which has shared concerns or tasks. It is generally managed by a third party or by the combination of one or more organizations in the community.

Advantages of the community cloud model:

- **Cost Effective:** It is cost-effective because the cloud is shared by multiple organizations or communities.
- **Security:** Community cloud provides better security.
- **Shared resources:** It allows you to share resources, infrastructure, etc. with multiple organizations.
- **Collaboration and data sharing:** It is suitable for both collaboration and data sharing.

5. Multi-cloud

We're talking about employing multiple cloud providers at the same time under this paradigm, as the name implies. It's similar to the hybrid cloud deployment approach, whichcombines public and private cloud resources. Instead of merging private and public clouds, multi-cloud uses many public clouds. Although public cloud providers provide numerous tools to improve the reliability of their services, mishaps still occur. It's quite rare that two distinct clouds would have an incident at the same moment. As a result, multi-cloud deployment improves the high availability of your services even more.

Advantages of a multi-cloud model:

- You can mix and match the best features of each cloud provider's services to suit the demands of your apps, workloads, and business by choosing different cloud providers.
- **Reduced Latency:** To reduce latency and improve user experience, you can choose cloud regions and zones that are close to your clients.
- **High availability of service:** It's quite rare that two distinct clouds would have an incident at the same moment. So, the multi-cloud deployment improves the high availability of your services.

Cloud Service Models

There are the following three types of cloud service models -

1. Software as a Service (SaaS)
2. Platform as a Service (PaaS)
3. Infrastructure as a Service (IaaS)

1) Software-as-a-Service (SaaS)

- SaaS is known as **'On-Demand Software'.**
- It is a software distribution model. In this model, the applications are hosted by a cloud service provider and publicized to the customers over internet.
- In SaaS, associated data and software are hosted centrally on the cloud server.
- User can access SaaS by using a thin client through a web browser.
- CRM, Office Suite, Email, games, etc. are the software applications which are provided as a service through Internet.
- The companies like Google, Microsoft provide their applications as a service to the end users.

Advantages of SaaS

- SaaS is easy to buy because the pricing of SaaS is based on monthly or annual fee and it allows the organizations to access business functionalities at a small cost, which is less than licensed applications.
- SaaS needed less hardware, because the software is hosted remotely, hence organizations do not need to invest in additional hardware.
- Less maintenance cost is required for SaaS and do not require special software or hardware versions.

Disadvantages of SaaS

- SaaS applications are totally dependent on Internet connection. They are not usable without Internet connection.
- It is difficult to switch amongst the SaaS vendors.

2) Platform-as-a-Service (PaaS)

- PaaS is a programming platform for developers. This platform is generated for the programmers to create, test, run and manage the applications.
- A developer can easily write the application and deploy it directly into PaaS layer.
- PaaS gives the runtime environment for application development and deployment tools.
- Google Apps Engine(GAE), Windows Azure, SalesForce.com are the examples of PaaS.

Advantages of PaaS

- PaaS is easier to develop. Developer can concentrate on the development and innovation without worrying about the infrastructure.
- In PaaS, developer only requires a PC and an Internet connection to start building applications.

Disadvantages of PaaS

- One developer can write the applications as per the platform provided by PaaS vendor hence the moving the application to another PaaS vendor is a problem.

3) Infrastructure-as-a-Service (IaaS)

IaaS is a way to deliver a cloud computing infrastructure like server, storage, network and operating system.

- The customers can access these resources over cloud computing platform i.e Internet as an on-demand service.
- In IaaS, you buy complete resources rather than purchasing server, software, datacenter space or network equipment.
- IaaS was earlier called as Hardware as a Service(HaaS). It is a Cloud computing platform based model.
- HaaS differs from IaaS in the way that users have the bare hardware on which they can deploy their own infrastructure using most appropriate software.

Advantages of IaaS

- In IaaS, user can dynamically choose a CPU, memory storage configuration according to need.
- Users can easily access the vast computing power available on IaaS Cloud platform.

Disadvantages of IaaS
- IaaS cloud computing platform model is dependent on availability of Internet and virtualization services.

Advantages of Cloud Computing
Cost Savings
Cost saving is the biggest benefit of cloud computing. It helps you to save substantial capital cost as it does not need any physical hardware investments. Also, you do not need trained personnel to maintain the hardware. The buying and managing of equipment is done by the cloud service provider.

Strategic edge
Cloud computing offers a competitive edge over your competitors. It helps you to access the latest and applications any time without spending your time and money on installations.
High Speed
Cloud computing allows you to deploy your service quickly in fewer clicks. This faster deployment allows you to get the resources required for your system within fewer minutes.
Back-up and restore data
Once the data is stored in a Cloud, it is easier to get the back-up and recovery of that, which is otherwise very time taking process on-premise.

Automatic Software Integration
In the cloud, software integration is something that occurs automatically. Therefore, you don't need to take additional efforts to customize and integrate your applications as per your preferences.

Reliability
Reliability is one of the biggest pluses of cloud computing. You can always get instantly updated about the changes.
Mobility
Employees who are working on the premises or at the remote locations can easily access all the could services. All they need is an Internet connectivity.
Unlimited storage capacity
The cloud offers almost limitless storage capacity. At any time you can quickly expand your storage capacity with very nominal monthly fees.
Collaboration
The cloud computing platform helps employees who are located in different geographies to collaborate in a highly convenient and secure manner.
Quick Deployment
Last but not least, cloud computing gives you the advantage of rapid deployment. So, when you decide to use the cloud, your entire system can be fully functional in very few minutes. Although, the amount of time taken depends on what kind of technologies are used in your business.

Other Important Benefits

Apart from the above, some other advantages of cloud computing are:

- On-Demand Self-service
- Multi-tenancy
- Offers Resilient Computing
- Fast and effective virtualization
- Provide you low-cost software
- Offers advanced online security
- Location and Device Independence
- Always available, and scales automatically to adjust to the increase in demand
- Allows pay-per-use
- Web-based control & interfaces
- API Access available.

Disadvantages of Cloud Computing

Performance Can Vary

When you are working in a cloud environment, your application is running on the server which simultaneously provides resources to other businesses. Any greedy behavior or DDOS attack on your tenant could affect the performance of your shared resource.

Technical Issues

Cloud technology is always prone to an outage and other technical issues. Even, the best cloud service provider companies may face this type of trouble despite maintaining high standards of maintenance.

Security Threat in the Cloud

Another drawback while working with cloud computing services is security risk. Before adopting cloud technology, you should be well aware of the fact that you will be sharing all your company's sensitive information to a third-party cloud computing service provider. Hackers might access this information.

Downtime

Downtime should also be considered while working with cloud computing. That's because your cloud provider may face power loss, low internet connectivity, service maintenance, etc.

Internet Connectivity

Good Internet connectivity is a must in cloud computing. You can't access cloud without an internet connection. Moreover, you don't have any other way to gather data from the cloud.

Lower Bandwidth

Many cloud storage service providers limit bandwidth usage of their users. So, in case if your organization surpasses the given allowance, the additional charges could be significantly costly

Lacks of Support

Cloud Computing companies fail to provide proper support to the customers. Moreover, they want their user to depend on FAQs or online help, which can be a tedious job for non-technical persons.

Here is a list of Top Cloud Service Providers for Quick Reference

- Amazon Web ServicesAlibaba Cloud.
- Microsoft AzureGoogle Cloud Platform.
- VMwareRackspace.
- SalesforceOracle Cloud.
- Verizon CloudNavisite.
- IBM CloudOpenNebula.

Conclusion:

Despite all the pro and cons, we can't deny the fact that Cloud Computing is the fastest growing part of network-based computing. It offers a great advantage to customers of all sizes: simple
users, developers, enterprises and all types of organizations. So, this technology here to stay for a long time.

Sign and Remark:

R1 (3 Marks)	R2 (2 Marks)	R3 (5 Marks)	R4 (5 Mark)	Total (15 Marks)	Signature

Dept: Computer Engineering		
Subject: Cloud Computing Lab	Subject Code: CSL605	
Year/Semester: TE-VI	Date:	Page No.
Student Name:	Roll No.	Division:

Experiment No.: 2

Aim: Perform case study on Hosted Virtualization using Virtual Box & KVM.

Software Required: Virtual Box

Theory :-
Virtualization is a technique of how to separate a service from the underlying physical delivery of that service. It is the process of creating a virtual version of something like computer hardware. It was initially developed during the mainframe era. It involves using specialized software to create a virtual or software-created version of a computing resource rather than the actual version of the same resource. With the help of Virtualization, multiple operating systems and applications can run on same machine and its same hardware at the same time, increasing the utilization and flexibility of hardware.

In other words, one of the main cost effective, hardware reducing, and energy saving techniques used by cloud providers is virtualization. Virtualization allows to share a single physical instance of a resource or an application among multiple customers and organizations at one time. It does this by assigning a logical name to a physical storage and providing a pointer to that physical resource on demand. The term virtualization is often synonymous with hardware virtualization, which plays a fundamental role in efficiently delivering Infrastructure-as-a-Service (IaaS) solutions for cloud computing. Moreover, virtualization technologies provide a virtual environment for not only executing applications but also for storage, memory, and networking.
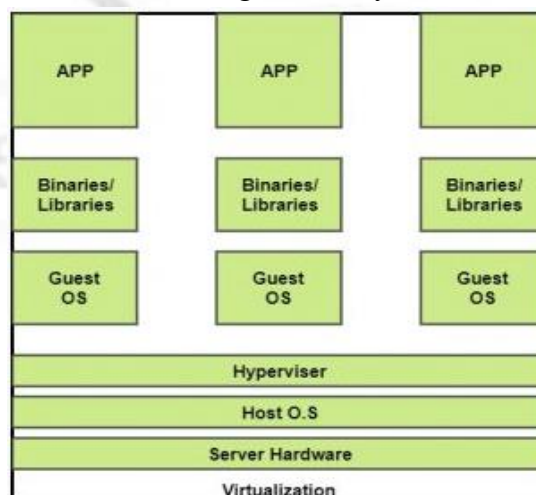
The machine on which the virtual machine is going to be built is known as Host Machine and that virtual machine is referred as a Guest Machine.

BENEFITS OF VIRTUALIZATION

1. More flexible and efficient allocation of resources.
2. Enhance development productivity.
3. It lowers the cost of IT infrastructure.
4. Remote access and rapid scalability.
5. High availability and disaster recovery.
6. Pay peruse of the IT infrastructure on demand.
7. Enables running multiple operating systems.

Types of Virtualization:

1. Application Virtualization.
2. Network Virtualization.
3. Desktop Virtualization.
4. Storage Virtualization.
5. Server Virtualization.
6. Data virtualization.

1. Application Virtualization:

Application virtualization helps a user to have remote access of an application from a server. The server stores all personal information and other characteristics of the application but can still run on a local workstation through the internet. Example of this would be a user who needs to run two different versions of the same software. Technologies that use application virtualization are hosted applications and packaged applications.

2. Network Virtualization:

The ability to run multiple virtual networks with each has a separate control and data plan. It co-exists together on top of one physical network. It can be managed by individual parties that potentially confidential to each other. Network virtualization provides a facility to create and provision virtual networks—logical switches, routers, firewalls, load balancer, Virtual Private Network (VPN), and workload security within days or even in weeks.

3. Desktop Virtualization:

Desktop virtualization allows the users' OS to be remotely stored on a server in the data centre. It allows the user to access their desktop virtually, from any location by a different machine. Users who want specific operating systems other than Windows Server will need to have a virtual desktop. Main benefits of desktop virtualization are user mobility, portability, easy management of software installation, updates, and patches.

4. Storage Virtualization:

Storage virtualization is an array of servers that are managed by a virtual storage system. The servers aren't aware of exactly where their data is stored, and instead function more like worker bees in a hive. It makes managing storage from multiple sources to be managed and utilized as a single repository. storage virtualization software maintains smooth operations, consistent performance and a continuous suite of advanced functions despite changes, break down and differences in the underlying equipment.

5. Server Virtualization:

This is a kind of virtualization in which masking of server resources takes place. Here, the central-server (physical server) is divided into multiple different virtual servers by changing the identity number, processors. So, each

system can operate its own operating systems in isolate manner. Where each sub-server knows the identity of the central server. It causes an increase in the performance and reduces the operating cost by the deployment of main server resources into a sub-server resource. It's beneficial in virtual migration, reduce energy consumption, reduce infrastructural cost, etc.

6. Data virtualization:

This is the kind of virtualization in which the data is collected from various sources and managed that at a single place without knowing more about the technical information like how data is collected, stored & formatted then arranged that data logically so that its virtual view can be accessed by its interested people and stakeholders, and users through the various cloud services remotely. Many big giant companies are providing their services like Oracle, IBM, At scale, Cdata, etc.

It can be used to performing various kind of tasks such as:

- **Data-integration**
- **Business-integration**
- **Service-oriented architecture data-services**
- **Searching organizational data**

VirtualBox is designed to run virtual machines on your physical machine without reinstalling your OS that is running on a physical machine.

Creating a Virtual Machine with VirtualBox

To create a virtual machine with Oracle VirtualBox, we should follow the steps given below.

Step 1 − To begin with, click on the "Oracle VM VirtualBox" icon on the desktop as shown in the screenshot below.

Step 2 − The next step is to click on "New" button, which is in the top left hand side of the screen.
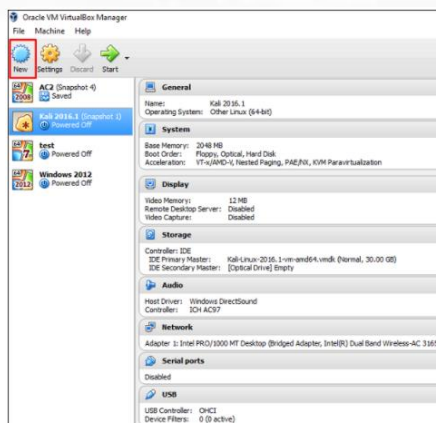
Step 3 − A table will pop-up requesting you the parameters for the virtual machine. These will be −
- **Name** − We have to put a friendly name for this Virtual Machine.
- **Type** − Enter the OS that is going to be installed on it.
- **Version** − Enter the specific version for that OS, which we have selected earlier.

Once all the above parameters are filled, click on "Next".
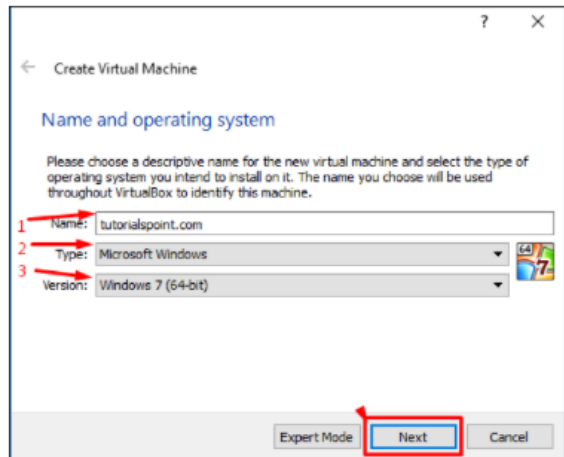
Step 4 − Select the amount of memory that you need to allocate in this VM → Click on "Next".
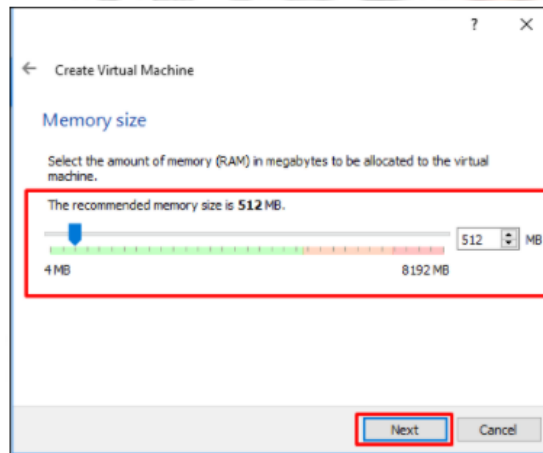
Step 5 − Check one of the three options for the HDD and click on "Create".
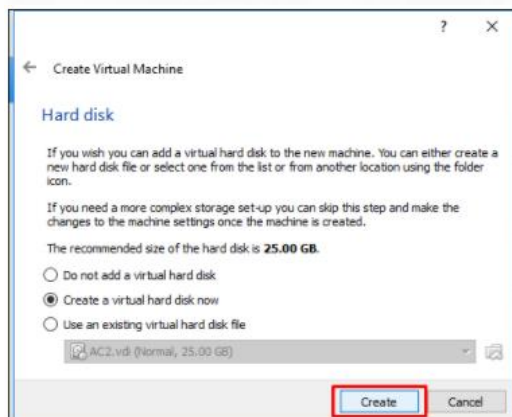
Step 6 − Select a file extension for your virtual HDD (It is recommended to use a common file extension that most of the hypervisors use like VHD) → click on "Next".
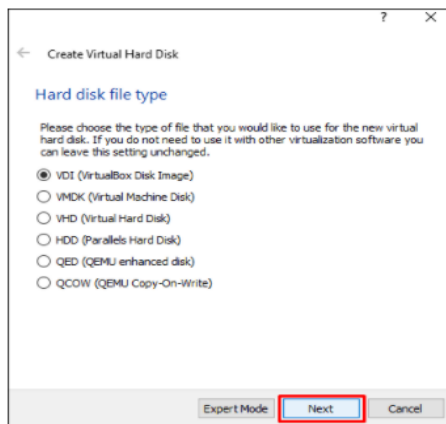
Step 7 − Choose whether you want the Virtual HDD as dynamic or fixed. This is based on your needs → Click on "Next".
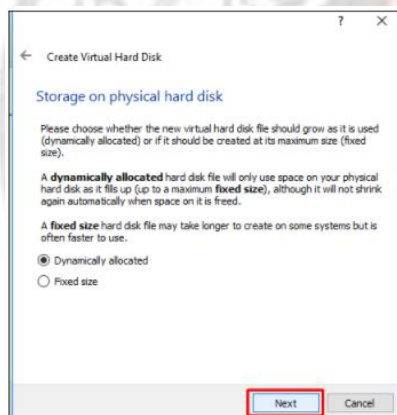
Step 8 − Put a name for your virtual HDD file and select the disk size for your VM → Click on "Create".
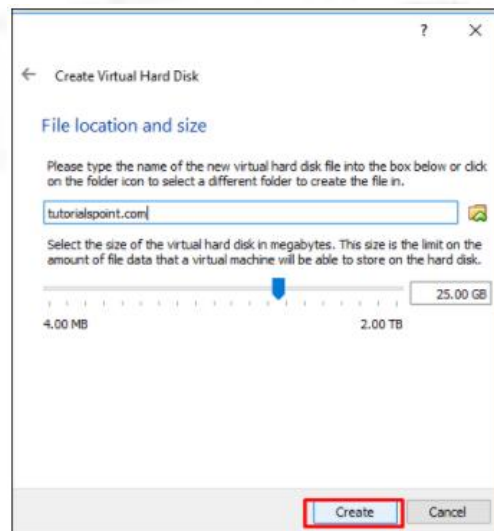
All the above steps can be done in one shot by selecting the "Expert mode".
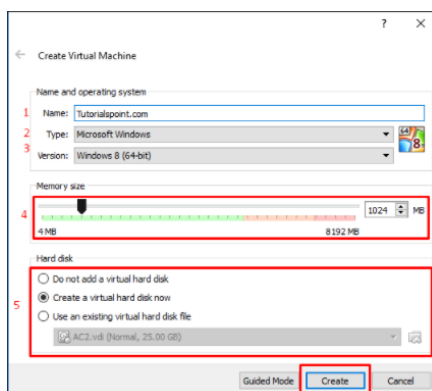
The virtual machine created will be as shown in the screenshot below.
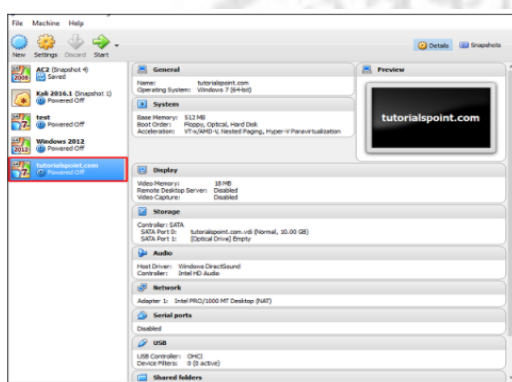
Setting up Networking with VirtualBox

There are two types of networking modes in VirtualBox, which are −

- Nat Networks and
- Host-only Networks.

Both of these are explained in detail below.

Nat Networks

For setting up Nat Networks, we should follow the steps given below.

Step 1 − Go to Oracle VM VirtualBox Manager → Click on "Preferences…"
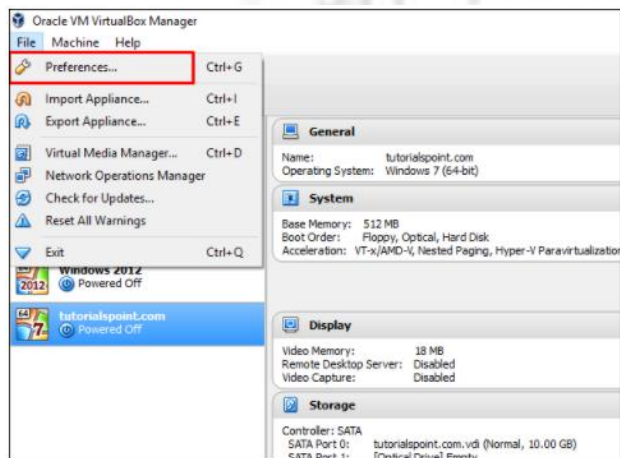
Step 2 − Click on "Network" and then on the left panel click on the "NAT Networks" tab.
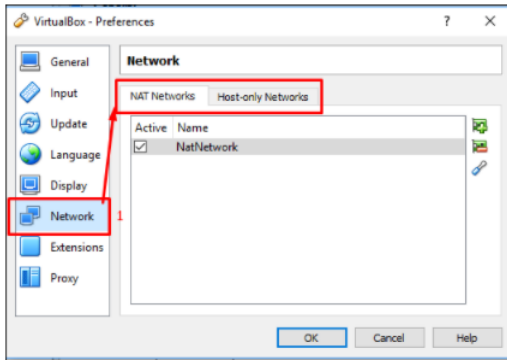
Step 3 − Click on the "+" button, which is highlighted in the screenshot below.
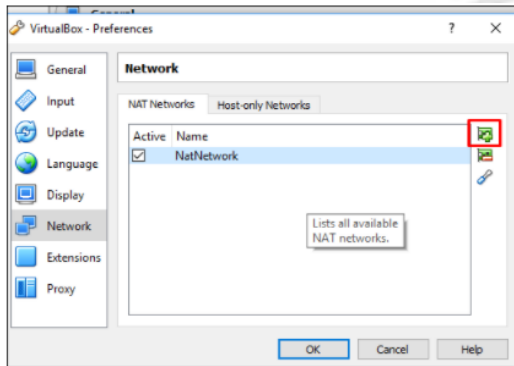
Step 4 − Here, we have to put the "Network Name" and the IP range for this network that will be NAT-ed, in order to have access to internet and to other networks. Host-only Networks For setting up Host-only Networks, we should follow the steps given below.
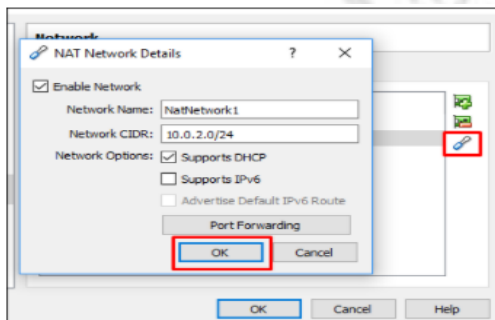
Step 1 − If you click on the "Host-only Networks" tab, you can create networks that are isolated from the other networks. However, VM hosts communicate with each other and the Hypervisor machine. Click on the "+" sign.
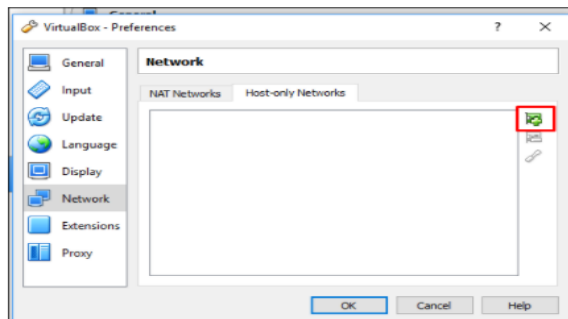
Step 2 − The host interface will continue to be created as shown in the screenshot below.
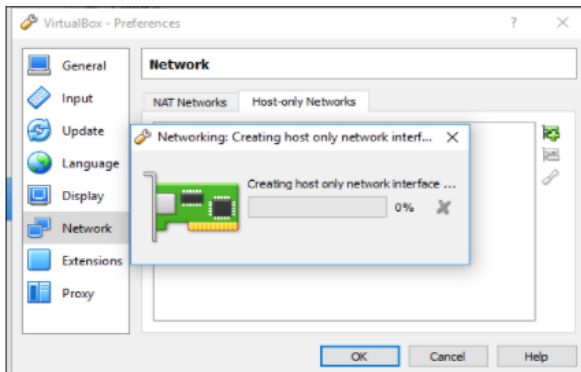
Step 3 − If you click [🖉] on button, you can edit the settings.
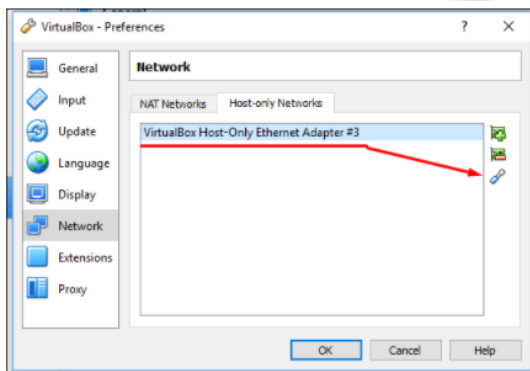
Step 4 − If you want your host machines to take "DHCP IP", click on the "DHCP Server" tab and check the box "Enable Server" → Click "OK".
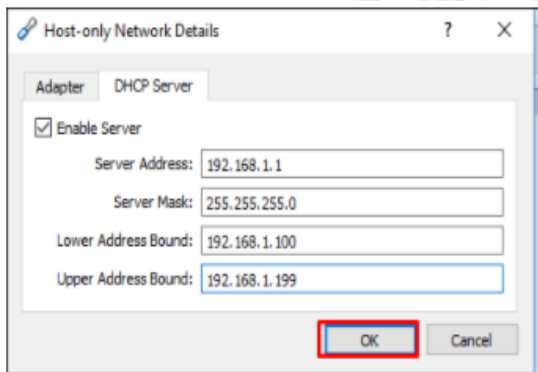
Step 5 − In the "Adapter" tab, put the IP of the hypervisor.
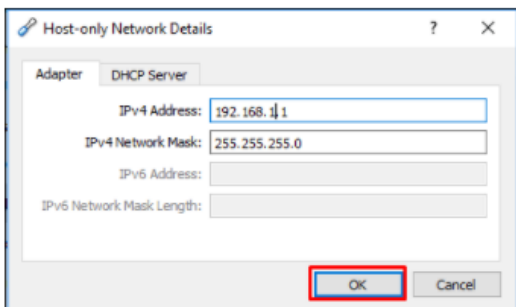
After all these preparations for setting up the network modes is complete. It is now time to assign a network to our VMs.

To do this, Click on the VMs on the left side of the panel, then right click on the "Network" option and a table will be open.
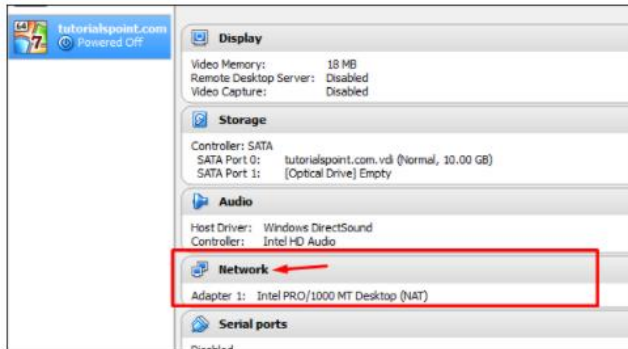
You can have up to four Ethernet adaptors per machine. The following image has four sections highlighted, which are explained below.

- Check the box "Enable Network Adapter" to enable the vNIC on the VM and attach it to one network.
- You can have many networks created, so we have to select one of them in the "Name" dropdown box.
- In the adapter type dropdown-box, we have to select a physical NIC that the hypervisor has.
- Promiscuous Mode: Here, we can select "Deny", if we do not want the VMs to communicate with each other.

Once all the above parameters are completed. Click on "OK".
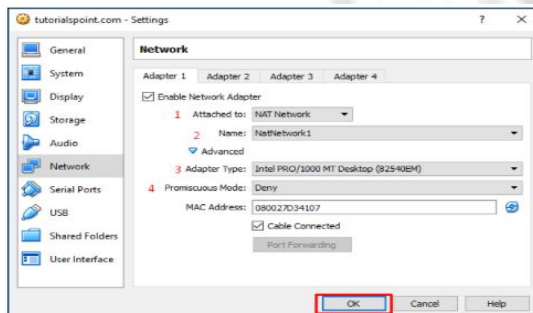

Allocating Processors & Memory to a VM

To allocate processors and memory to a virtual machine using VirtualBox, we should follow the steps given below.

Step 1 − To allocate a processor and memory, you have to click on "Settings" after you have selected the VM.
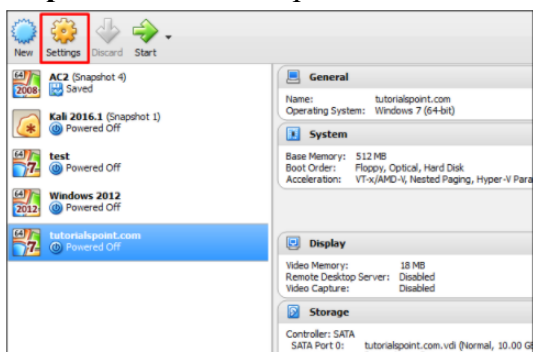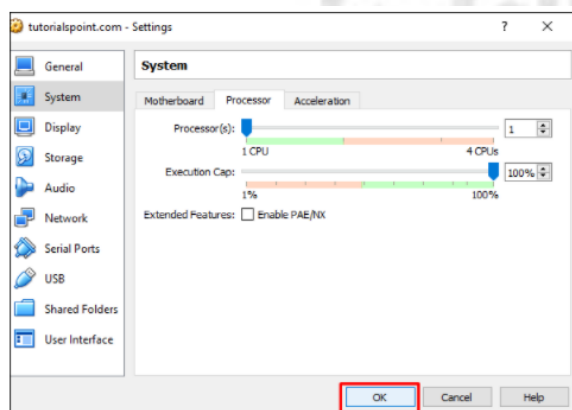
Step 2 − Click on "System" on the left side tab, then click on the "Motherboard" tab. Move the arrow left or right to allocate the memory as shown in the screenshot below.

Step 3 − To allocate processors, click on the "Processor" tab. Move the arrow left or right to allocate the number of processors as shown in the screenshot below.
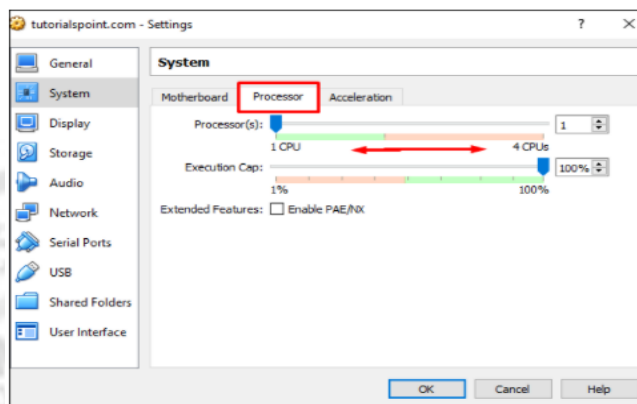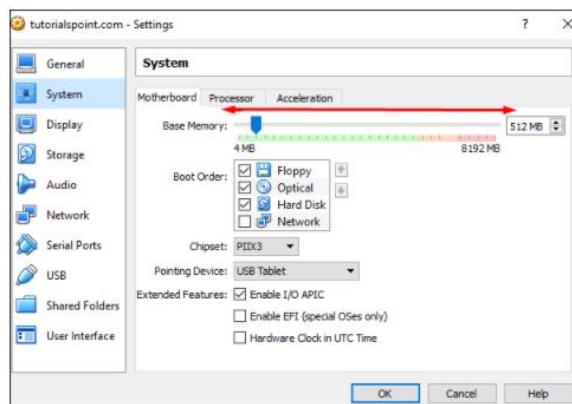
Step 4 − After all those changes are done → click on "OK".

Conclusion:

VirtualBox is a powerful virtualization solution that can be used for running distinct operating systems. VirtualBox's user interface is convenient and standardized for all supported host operating systems. VirtualBox for running software developed for different operating systems on your single physical machine simultaneously without the need to install multiple operating systems on a physical machine, and reboot to change the OS. VirtualBox provides a high range of virtual networking modes.

Sign and Remark:

R1 (3 Marks)	R2 (2 Marks)	R3 (5 Marks)	R4 (5 Mark)	Total (15 Marks)	Signature

Dept: Computer Engineering		
Subject: Cloud Computing Lab	**Subject Code:** CSL605	
Year/Semester: TE-VI	**Date:**	**Page No.**
Student Name:	**Roll No.**	**Division:**

Experiment No.:3

Aim: To study and Implement Bare-metal Virtualization using Xen.

Software Required:

Theory:
A **hypervisor**, also known as a virtual machine monitor or VMM, is a type of virtualization software that supports the creation and management of virtual machines (VMs) by separating a computer's software from its hardware. Hypervisors translate requests between the physical and virtual resources, making virtualization possible. When a hypervisor is installed directly on the hardware of a physical machine, between the hardware and the operating system (OS), it is called a **bare metal hypervisor**. Some bare metal hypervisors are embedded into the firmware at the same level as the motherboard basic input/output system (BIOS). This is necessary for some systems to enable the operating system on a computer to access and use virtualization software.

Because the bare metal hypervisor separates the OS from the underlying hardware, the software no longer relies on or is limited to specific hardware devices or drivers. This means bare metal hypervisors allow operating systems and their associated applications to run on a variety of types of hardware. They also allow multiple operating systems and virtual machines (guest machines) to reside on the same physical server (host machine). Because the virtual machines are independent of the physical machine, they can move from machine to machine or platform to platform, shifting workloads and allocating networking, memory, storage, and processing resources across multiple servers according to needs. For example, when an application needs more processing power, it can seamlessly access additional machines through the virtualization software. This results in greater cost and energy efficiency and better performance, using fewer physical machines.

XenServer 7.1 Cumulative Updates are provided as both an update to the previous Cumulative Update of XenServer 7.1 and as a base installation that can be used to upgrade other versions of XenServer or to create a fresh installation.
Fresh installation of XenServer 7.1 Cumulative Update X:
- Use the XenServer 7.1 Base Installation ISO including Cumulative Update X file
You can download this file from the download site
- Review the information in System Requirements, To license Citrix XenServer:, and Install XenServer and XenCenter before installing XenServer.
XenServer installs directly on bare-metal hardware avoiding the complexity, overhead, and performance bottlenecks of an underlying operating system. It uses the device drivers available

from the Linux kernel. As a result, XenServer can run on a wide variety of hardware and storage devices.

To install or upgrade the XenServer host:

1. Boot the computer from the installation CD or, if applicable, network-boot from your TFTP server.
2. Following the initial boot messages and the Welcome to XenServer screen, select your keymap (keyboard layout) for the installation.
3. The Welcome to XenServer Setup screen is displayed.

XenServer ships with a broad driver set that supports most modern server hardware configurations. However, if you have been provided with any additional essential device drivers, press F9. The installer steps you through installing the necessary drivers.

4. The XenServer EULA is displayed. Use the Page Up and Page Down keys to scroll through and read the agreement. Select **Accept EULA** to proceed.
5. Select the appropriate action. You might see any of the following options:
   - *Perform clean installation*
   - *Upgrade*: If the installer detects a previously installed version of XenServer, it offers the option to upgrade. For information about upgrading your XenServer host, see **Upgrade from an existing version**.
   - *Restore*: If the installer detects a previously created backup installation, it offers the option to restore XenServer from the backup.

Make your selection, and choose **OK** to proceed.

6. If you have multiple local hard disks, choose a Primary Disk for the installation. Select Ok.
7. Choose which disk(s) you would like to use for virtual machine storage. Information about a specific disk can be viewed by pressing F5.

If you want to use Thin Provisioning to optimize the utilization of available storage, select **Enable thin provisioning**. Citrix Virtual Desktop users are *strongly* recommended to select this option in order for local caching to work properly. For details, see **IntelliCache**.

Choose **OK**.

8. Select your installation media source.

To install from a CD, choose **Local media**. To install by using network, select **HTTP** or **FTP** or **NFS**.

Choose **OK** to proceed.

If you select **HTTP** or **FTP** or **NFS**:

- Set up networking so that the installer can connect to the XenServer installation media files.

If the computer has multiple NICs, select one of them to be used to access the XenServer installation media files. Choose **OK** to proceed.

- Choose **Automatic configuration (DHCP)** to configure the NIC using DHCP, or Static configuration to configure the NIC manually. If you choose **Static configuration**, enter details as appropriate.
- If you choose **HTTP** or **FTP**, provide the URL for your HTTP or FTP repository, and a user name and password, if appropriate.
- If you choose **NFS**, provide the server and path of your NFS share.

Select **OK** to proceed.

9. Indicate if you want to verify the integrity of the installation media. If you select **Verify installation source**, the SHA256 checksum of the packages is calculated and checked against the known value. Verification can take some time. Make your selection and choose **OK** to proceed.
10. Set and confirm a root password, which XenCenter uses to connect to the XenServer host. You also use this password (with user name "root") to log into **xsconsole**, the system configuration console.

11. Set up the primary management interface that is used to connect to XenCenter.

If your computer has multiple NICs, select the NIC which you want to use for management. Choose **OK** to proceed.

12. Configure the Management NIC IP address by choosing **Automatic configuration (DHCP)** to configure the NIC using DHCP, or **Static configuration** to configure the NIC manually. To have the management interface on a VLAN network, provide the VLAN ID.

13. Specify the hostname and the DNS configuration, manually or automatically via DHCP.

In the **Hostname Configuration** section, select **Automatically set via DHCP** to have the DHCP server provide the hostname along with the IP address. If you select **Manually specify**, enter the hostname for the server in the field provided.

In the **DNS Configuration** section, choose **Automatically set via DHCP** to get name service configuration using DHCP. If you select **Manually specify**, enter the IP addresses of your primary (required), secondary (optional), and tertiary (optional) DNS servers in the fields provided.

Select **OK** to proceed.

14. Select your time zone by geographical area and city. You can type the first letter of the desired locale to jump to the first entry that begins with this letter. Choose **OK** to proceed.

15. Specify how you want the server to determine local time: using NTP or manual time entry. Make your selection, and choose **OK** to proceed.

16. If using NTP, select **NTP is configured by my DHCP server** or enter at least one NTP server name or IP address in the fields below. Choose **OK**.

17. Select **Install XenServer**.

If you elected to set the date and time manually, you are prompted to do so during the installation. Once set, choose **OK** to proceed.

18. If you are installing from CD, the next screen asks if you want to install any supplemental packs from a CD. If you plan to install any supplemental packs provided by your hardware supplier, choose **Yes**.

If you choose to install supplemental packs, you are prompted to insert them. Eject the XenServer installation CD, and insert the supplemental pack CD. Choose **OK**.

Select **Use media** to proceed with the installation.

Repeat for each pack to be installed.

19. From the **Installation Complete** screen, eject the installation CD (if installing from CD) and select **OK** to reboot the server.

After the server reboots, XenServer displays **xsconsole**, a system configuration console. To access a local shell from **xsconsole**, press **Alt+F3**; to return to **xsconsole**, press **Alt+F1**.

Install XenCenter

XenCenter must be installed on a Windows machine that can connect to the XenServer host through your network. Ensure that .NET framework version 4.6 or above is installed on this system.

Download the latest version of XenCenter from the **XenServer 7.1 Download page**.

To install XenCenter:

1. Before installing XenCenter, be sure to uninstall any previous version.
2. Launch the installer. Double-click XenCenter.msi to begin the installation.
3. Follow the Setup wizard, which allows you to modify the default destination folder and then to install XenCenter.

Connect XenCenter to the XenServer host

To connect XenCenter to the XenServer host:
1. Launch XenCenter. The program opens to the **Home** tab.
2. Click the **Add New Server** icon.
3. Enter the IP address of the XenServer host in the **Server** field. Type the root user name and password that you set during XenServer installation. Click **Add**.
4. The first time you add a host, the **Save and Restore Connection State** dialog box appears. This dialog enables you to set your preferences for storing your host connection information and automatically restoring host connections.

If you later want to change your preferences, you can do so using XenCenter or the Windows Registry Editor.
To do so in XenCenter: from the main menu, select **Tools** and then **Options**. The **Options** dialog box opens. Select the **Save and Restore** tab and set your preferences. Click **OK** to save your changes.
The wizard's initial screen asks what type of computer you want to add. Choose the Physical Computers to be Provisioned as Virtual Machine Hosts option, shown in the screenshot below.

Click Next, and you will be taken to the Credentials and Protocol screen. Because the new server doesn't contain an operating system, it will have to be discovered using out of band management. That being the case, you will need to select a Run As account and choose the out of band management protocol that you want to use.
Click Next to go to the Discovery Scope screen. Here you simply need to enter the IP address or IP address range of the servers that you want to discover. Click Next. If you entered a single IP address then you can skip the Target Resources screen. Otherwise, review the list of systems that have been discovered and select the ones that you want to provision.
Click Next, and you will be taken to the Provisioning Options screen. Here you need to select the host group within which you wish to place the newly provisioned hosts. You will also need to select the physical computer profile that you created earlier.

Click Next and you will be taken to the Deployment Customization screen. Here you can provide computer names, and select network adapters. The options that you configure on this screen are going to vary based on your own needs.

Click Next one more time, and you will be taken to the Summary screen. Take a moment and review the summary items. If everything appears to be correct, click Finish to begin the deployment process.

Conclusion:

Sign and Remark:

R1 (3 Marks)	R2 (2 Marks)	R3 (5 Marks)	R4 (5 Mark)	Total (15 Marks)	Signature

Dept: Computer Engineering		
Subject: Cloud Computing Lab	**Subject Code:** CSL605	
Year/Semester: TE-VI	**Date:**	**Page No.**
Student Name:	**Roll No.**	**Division:**

Experiment No.:4

Aim: To study and Implement Infrastructure as a service using AWS

Software Required:

Theory:

With the help of Virtual Machine in Cloud Computing, virtual hosts can share resources between multiple guests or Virtual Machines. They will each have their operating system instance. The two basic types of Virtual Machines are Process and System VMs. A Process Virtual Machine runs a single process as an application on a host machine. Java Virtual Machine (JVM) is considered as an example of a Process Virtual Machine. These types of Cloud-based Virtual Machines allow any system to run Java applications as if they were native to that particular system. A System Virtual Machine is a completely virtualized Virtual Machine Technology in Cloud Computing designed to replace a physical machine. It runs on a different host machine by properly using a hypervisor such as VMware ESXi to access the underlying machine's resources. A Virtualbox in Cloud Computing consists of the remaining system. It means the particular software present inside a specific Virtual Machine in Cloud Computing has no access to getting escaped or tampered by the computer.

There are a few constraints in x86 servers. To overcome this, many Information Technology companies started deployment on multiple servers. By adopting this, it resulted in a larger rate of inefficiencies and high operating costs.

It is highly recommended to enter Virtual Machine Technology in Cloud Computing, which only reckons on software that directly stimulates the hardware functionality and which, in turn, results in the creation of a virtual cloud computer system. It also allows many Information Technology companies to run and use one or more virtual networks as well as more operating systems and applications functioning on a single server. The benefit of using this resulted in greater efficiency and meeting the scalability of economies.

Microsoft developed its very own cloud service provider, called 'Azure.' Azure gives many services for Software Developers, and one of these services is 'Virtual Machines.' Virtual Machines are created virtually in Azure Cloud Computing. They look like a standard computer. The Virtual Machine in the Cloud Computing of Azure allocates many images in the cloud platform and has made the deployment quite quick and easy.

Requirements for creating cloud Virtual Machine windows 10-

● Good and Stable internet connection
● RDP software
● Edge or other browsers
● A registered account in Azure cloud

To Install RDP Software:
1. After the installer downloads, open the Microsoft Remote Desktop installer file.
2. The setup wizard screen appears. Click or tap **Next**.
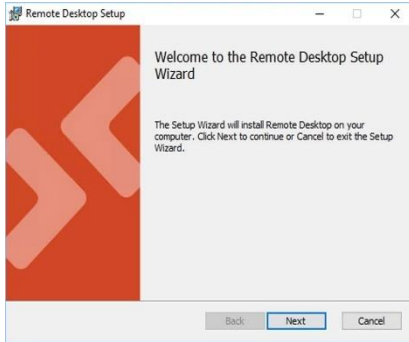
3. The license agreement screen appears. Review the license and, if you agree, check the box for **"I accept the terms in the License Agreement."** Then click or tap **Next**.


4. The software installs on your computer. If successful, the screen announcing "Completed the Remote Desktop Setup Wizard" appears. Keep the checkbox for "Launch Remote Desktop when setup exits" checked. Click or tap **Finish**.
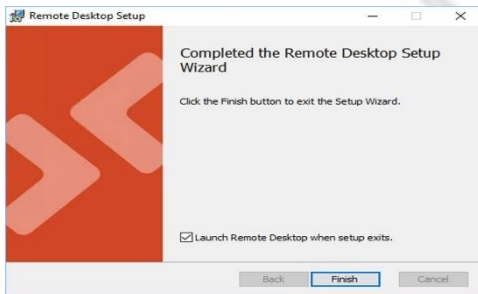
Set up Remote Desktop for Windows

- Microsoft Remote Desktop launches on your computer. On the "Let's get started" screen, click or tap **Subscribe with URL**.
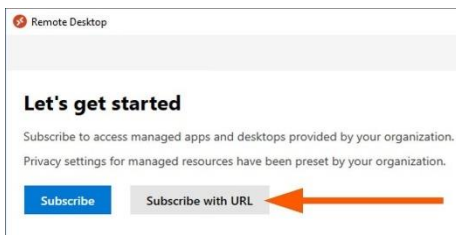
- The "Subscribe to a Workspace" section appears on the left of the screen. From your D2L course site, identify the web address for connecting to remote computers via desktop application. Enter that web address into the text box. Click or tap **Next**.
- You are prompted to sign in. Enter your marquette.edu email address. Click or tap **Next**.

- Enter your Marquette password. Click or tap **Next**.

- The "Stay signed in to all you apps" screen appears. Unselect the checkbox for **"Allow my organization to manage my device."** Click or tap the link **"No, sign in to this app only"** to continue.

- You have signed in. Your college's remote computer lab screen appears.

Note: The single computer icon pictured below represents *multiple* computers. It does not mean there is only one computer left in this remote lab.

Click or tap the **Windows 10 Desktop icon**.
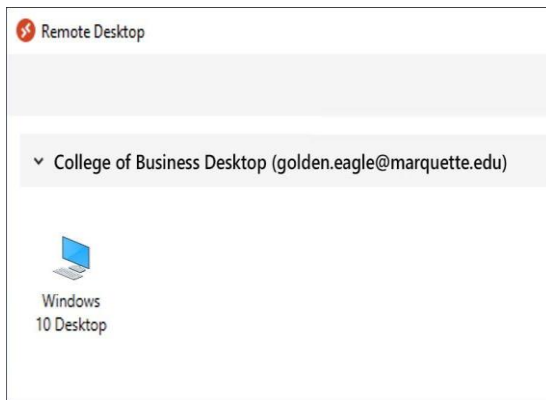
- Enter your Marquette password. Click or tap **OK**.
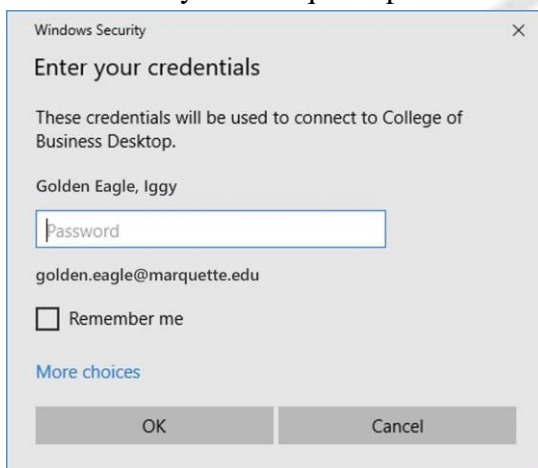
- The Windows 10 desktop appears.

You have successfully connected to the remote computer lab.

To Create an account on AWS:

Step 1: Launch an instance

You can launch a Windows instance using the AWS Management Console as described in the following procedure. This tutorial is intended to help you launch your first instance quickly, so it doesn't cover all possible options. For more information about the advanced options, see Launch an instance using the Launch Instance Wizard. For information about other ways to launch your instance, see Launch your instance.

To launch an instance

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. From the console dashboard, choose **Launch Instance**.
3. The **Choose an Amazon Machine Image (AMI)** page displays a list of basic configurations, called *Amazon Machine Images (AMIs)*, that serve as templates for your instance. Select the AMI for Windows Server 2016 Base or later. Notice that these AMIs are marked "Free tier eligible."
4. On the **Choose an Instance Type** page, you can select the hardware configuration of your instance. Select the t2.micro instance type, which is selected by default. The t2.micro instance type is eligible for the free tier. In Regions where t2.micro is unavailable, you can use a t3.micro instance under the free tier.
5. On the **Choose an Instance Type** page, choose **Review and Launch** to let the wizard complete the other configuration settings for you.
6. On the **Review Instance Launch** page, under **Security Groups**, you'll see that the wizard created and selected a security group for you. You can use this security group, or alternatively you can select the security group that you created when getting set up using the following steps:
a. Choose **Edit security groups**.
b. On the **Configure Security Group** page, ensure that **Select an existing security group** is selected.
c. Select your security group from the list of existing security groups, and then choose **Review and Launch**.
7. On the **Review Instance Launch** page, choose **Launch**.
8. When prompted for a key pair, select **Choose an existing key pair**, then select the key pair that you created when getting set up. Note that you must select an **RSA** key. **ED25519** keys are not supported for Windows instances.

Warning

Don't select **Proceed without a key pair**. If you launch your instance without a key pair, then you can't connect to it.

When you are ready, select the acknowledgement check box, and then choose **Launch Instances**.

9. A confirmation page lets you know that your instance is launching. Choose **View Instances** to close the confirmation page and return to the console.
10. On the **Instances** screen, you can view the status of the launch. It takes a short time for an instance to launch. When you launch an instance, its initial state is pending. After the instance starts, its state changes to running and it receives a public DNS name. (If the **Public IPv4 DNS** column is hidden, choose the settings icon (⚙) in the top-right corner, toggle on **Public IPv4 DNS**, and choose **Confirm**.
11. It can take a few minutes for the instance to be ready so that you can connect to it. Check that your instance has passed its status checks; you can view this information in the **Status check** column.

Step 2: Connect to your instance

To connect to a Windows instance, you must retrieve the initial administrator password and then enter this password when you connect to your instance using Remote Desktop. It takes a few minutes after instance launch before this password is available.

The name of the administrator account depends on the language of the operating system. For example, for English, it's Administrator, for French it's Administrateur, and for Portuguese it's Administrador. For more information, see Localized Names for Administrator Account in Windows in the Microsoft TechNet Wiki.
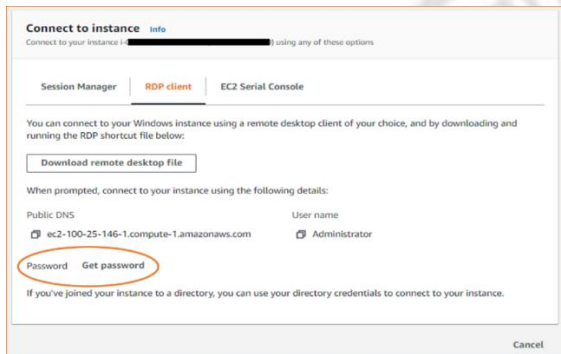
If you've joined your instance to a domain, you can connect to your instance using domain credentials you've defined in AWS Directory Service. On the Remote Desktop login screen, instead of using the local computer name and the generated password, use the fully-qualified user name for the administrator (for example, **corp.example.com\Admin**), and the password for this account.

If you receive an error while attempting to connect to your instance, see Remote Desktop can't connect to the remote computer.

- New console
- Old console

To connect to your Windows instance using an RDP client

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation pane, select **Instances**. Select the instance and then choose **Connect**.
3. On the **Connect to instance** page, choose the **RDP client** tab, and then choose **Get password**.

4. Choose **Browse** and navigate to the private key (.pem) file you created when you launched the instance. Select the file and choose **Open** to copy the entire contents of the file to this window.
5. Choose **Decrypt Password**. The console displays the default administrator password for the instance under **Password**, replacing the **Get password** link shown previously. Save the password in a safe place. This password is required to connect to the instance.

6. Choose **Download remote desktop file**. Your browser prompts you to either open or save the RDP shortcut file. When you have finished downloading the file, choose **Cancel** to return to the **Instances** page.
   - If you opened the RDP file, you'll see the **Remote Desktop Connection** dialog box.
- If you saved the RDP file, navigate to your downloads directory, and open the RDP file to display the dialog box.

7. You may get a warning that the publisher of the remote connection is unknown. Choose **Connect** to continue to connect to your instance.
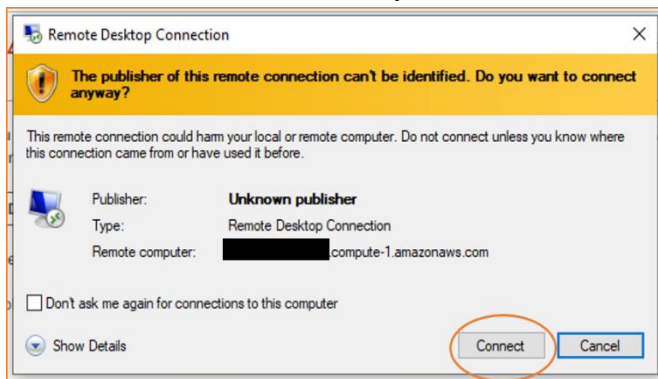
6. The administrator account is chosen by default. Copy and paste the password that you saved previously.

Tip: If you receive a "Password Failed" error, try entering the password manually. Copying and pasting content can corrupt it.

7. Due to the nature of self-signed certificates, you may get a warning that the security certificate could not be authenticated. Use the following steps to verify the identity of the remote computer, or simply choose **Yes** (Windows) or **Continue** (Mac OS X) if you trust the certificate.
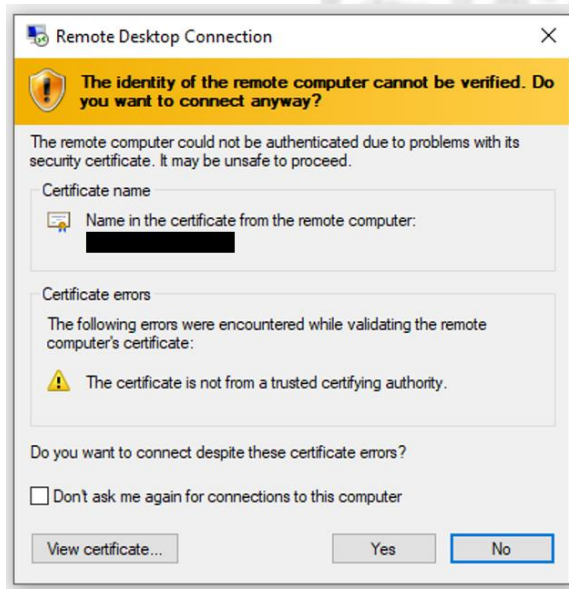
- If you are using **Remote Desktop Connection** on a Windows computer, choose **View certificate**. If you are using **Microsoft Remote Desktop** on a Mac, choose **Show Certificate**.
- Choose the **Details** tab, and scroll down to **Thumbprint** (Windows) or **SHA1 Fingerprints** (Mac OS X). This is the unique identifier for the remote computer's security certificate.
- In the Amazon EC2 console, select the instance, choose **Actions**, **Monitor and troubleshoot**, **Get system log**.
- In the system log output, look for RDPCERTIFICATE-THUMBPRINT. If this value matches the thumbprint or fingerprint of the certificate, you have verified the identity of the remote computer.
- If you are using **Remote Desktop Connection** on a Windows computer, return to the **Certificate** dialog box and choose **OK**. If you are using **Microsoft Remote Desktop** on a Mac, return to the **Verify Certificate** and

choose **Continue**.

- [Windows] Choose **Yes** in the **Remote Desktop Connection** window to connect to your instance.

[Mac OS X] Log in as prompted, using the default administrator account and the default administrator password that you recorded or copied previously. Note that you might need to switch spaces to see the login screen. For more information, see Add spaces and switch between them.

Step 3: Clean up your instance

After you've finished with the instance that you created for this tutorial, you should clean up by terminating the instance. If you want to do more with this instance before you clean up, see Next steps.

Important

Terminating an instance effectively deletes it; you can't reconnect to an instance after you've terminated it.

If you launched an instance that is not within the AWS Free Tier, you'll stop incurring charges for that instance as soon as the instance status changes to shuttingdown or terminated. To keep your instance for later, but not incur charges, you can stop the instance now and then start it again later. For more information, see Stop and start your instance.

To terminate your instance

1. In the navigation pane, choose **Instances**. In the list of instances, select the instance.
2. Choose **Instance state**, **Terminate instance**.
3. Choose **Terminate** when prompted for confirmation.

Amazon EC2 shuts down and terminates your instance. After your instance is terminated, it remains visible on the console for a short while, and then the entry is automatically deleted. You cannot remove the terminated instance from the console display yourself.

Next steps

After you start your instance, you might want to try some of the following exercises:

- Learn how to remotely manage your EC2 instance using Run Command. For more information, see AWS Systems Manager Run Command in the *AWS Systems Manager User Guide*.
- Configure a CloudWatch alarm to notify you if your usage exceeds the Free Tier. For more information, see Tracking your AWS Free Tier usage in the *AWS Billing and Cost Management User Guide*.
- Add an EBS volume. For more information, see Create an Amazon EBS volume and Attach an Amazon EBS volume to an instance.

Conclusion:

Sign and Remark:

R1 (3 Marks)	R2 (2 Marks)	R3 (5 Marks)	R4 (5 Mark)	Total (15 Marks)	Signature

DATTA MEGHE COLLEGE OF ENGINEERING

Dept: Computer Engineering		
Subject: Cloud Computing Lab	Subject Code: CSL605	
Year/Semester: TE-VI	Date:	Page No.
Student Name:	Roll No.	Division:

Experiment No.:5

Aim: To study and Implement Platform As A Service using AWS Elastic Benstalk

Software Required:
Theory:
AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

You can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring. At the same time, you retain full control over the AWS resources powering your application and can access the underlying resources at any time. There is no additional charge for Elastic Beanstalk - you pay only for the AWS resources needed to store and run your applications.

Steps To use AWS Elastic Benstalk :
Setting up: Create an AWS account
If you're not already an AWS customer, you need to create an AWS account. Signing up enables you to access Elastic Beanstalk and other AWS services that you need.
To sign up for an AWS account
1. Open the Elastic Beanstalk console, and in the **Regions** list, select your AWS Region.
Step 1: Create an example application
In this step, you create a new application starting from a pre-existing example application. Elastic Beanstalk supports platforms for different programming languages, application servers, and Docker containers. You choose a platform when you create the application.
Create an application and an environment
To create your example application, you'll use the **Create a web app** console wizard. It creates an Elastic Beanstalk application and launches an environment within it. An environment is the collection of AWS resources required to run your application code.
To create an example application
1. Open the Elastic Beanstalk console using this link: https://console.aws.amazon.com/elasticbeanstalk/home#/gettingStarted?applicationName=getting-started-app

2. Optionally add application tags.
3. For **Platform**, choose a platform, and then choose **Create application**.

To run the example application on AWS resources, Elastic Beanstalk takes the following actions. They take about five minutes to complete.

1. Creates an Elastic Beanstalk application named **getting-started-app**.
2. Launches an environment named **Getting Started App-env** with these AWS resources:
   - An Amazon Elastic Compute Cloud (Amazon EC2) instance (virtual machine)
   - An Amazon EC2 security group
   - An Amazon Simple Storage Service (Amazon S3) bucket
   - Amazon Cloud Watch alarms
   - An AWS Cloud Formation stack
   - A domain name

For details about these AWS resources, see AWS resources created for the example application.

3. Creates a new application version named **Sample Application**. This is the default Elastic Beanstalk example application file.
4. Deploys the code for the example application to the **Getting Started App-env** environment.

During the environment creation process, the console tracks progress and displays events.
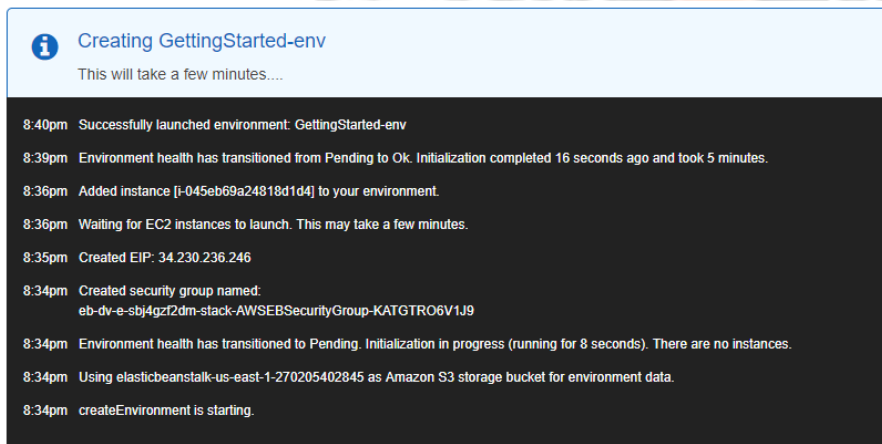
When all of the resources are launched and the EC2 instances running the application pass health checks, the environment's health changes to Ok. You can now use your web application's website.

AWS resources created for the example application

When you create the example application, Elastic Beanstalk creates the following AWS resources:

- **EC2 instance** – An Amazon EC2 virtual machine configured to run web apps on the platform you choose. Each platform runs a different set of software, configuration files, and scripts to support a specific language version, framework, web container, or combination thereof. Most platforms use either Apache or nginx as a reverse proxy that processes web traffic in front of your web app, forwards requests to it, serves static assets, and generates access and error logs.
- **Instance security group** – An Amazon EC2 security group configured to allow incoming traffic on port 80. This resource lets HTTP traffic from the load balancer reach the EC2 instance running your web app. By default, traffic is not allowed on other ports.

- **Amazon S3 bucket** – A storage location for your source code, logs, and other artifacts that are created when you use Elastic Beanstalk.
- **Amazon Cloud Watch alarms** – Two Cloud Watch alarms that monitor the load on the instances in your environment and are triggered if the load is too high or too low. When an alarm is triggered, your Auto Scaling group scales up or down in response.
- **AWS Cloud Formation stack** – Elastic Beanstalk uses AWS Cloud Formation to launch the resources in your environment and propagate configuration changes. The resources are defined in a template that you can view in the AWS Cloud Formation console.
- **Domain name** – A domain name that routes to your web app in the form *subdomain.region.elasticbeanstalk.com*.
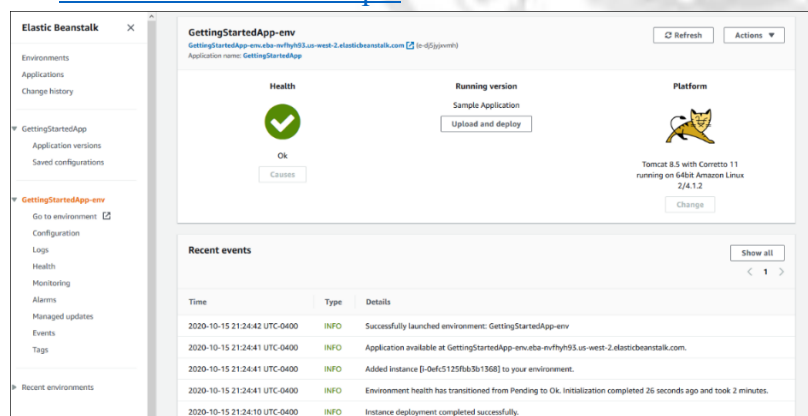
Step 2: Explore your environment

To see an overview of your Elastic Beanstalk application's environment, use the environment page in the Elastic Beanstalk console.

To view the environment overview

1. Open the Elastic Beanstalk console, and in the **Regions** list, select your AWS Region.
2. In the navigation pane, choose **Environments**, and then choose the name of your environment from the list.

Note: If you have many environments, use the search bar to filter the environment list.

The environment overview pane shows top level information about your environment. This includes its name, its URL, its current health status, the name of the currently deployed application version, and the platform version that the application is running on. Below the overview pane you can see the five most recent environment events. To learn more about environment tiers, platforms, application versions, and other Elastic Beanstalk concepts, see Elastic Beanstalk concepts.

While Elastic Beanstalk creates your AWS resources and launches your application, the environment is in a Pending state. Status messages about launch events are continuously added to the overview.

The environment's **URL** is located at the top of the overview, below the environment name. This is the URL of the web application that the environment is running. Choose this URL to get to the example application's *Congratulations* page.

The navigation page on the left side of the console links to other pages that contain more detailed information about your environment and provide access to additional features:

- **Configuration** – Shows the resources provisioned for this environment, such as the Amazon Elastic Compute Cloud (Amazon EC2) instances that host your application. You can configure some of the provisioned resources on this page.
- **Health** – Shows the status of and detailed health information about the Amazon EC2 instances running your application.
- **Monitoring** – Shows statistics for the environment, such as average latency and CPU utilization. You can use this page to create alarms for the metrics that you are monitoring.
- **Events** – Shows information or error messages from the Elastic Beanstalk service and from other services whose resources this environment uses.
- **Tags** – Shows environment tags and allows you to manage them. Tags are key-value pairs that are applied to your environment.

Step 3: Deploy a new version of your application Periodically, you might need to deploy a new version of your application. You can deploy a new version at any time, as long as no other update operations are in progress on your environment. The application version that you started this tutorial with is called **Sample Application**.
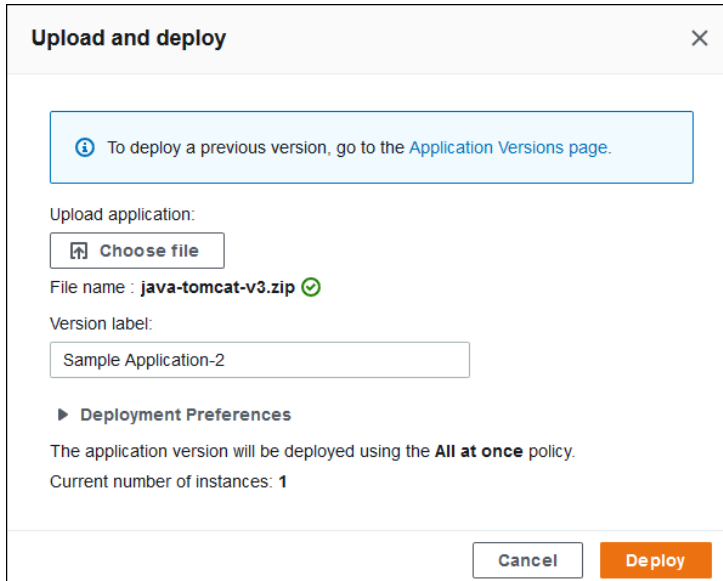
To update your application version

1. Download the sample application that matches your environment's platform. Use one of the following applications.
    - **Docker** – docker.zip
    - **Multicontainer Docker** – docker-multicontainer-v2.zip
    - **Preconfigured Docker (Glassfish)** – docker-glassfish-v1.zip
    - **Go** – go.zip
    - **Corretto** – corretto.zip
    - **Tomcat** – tomcat.zip
    - **.NET Core on Linux** – dotnet-core-linux.zip
    - **.NET** – dotnet-asp-v1.zip
    - **Node.js** – nodejs.zip
    - **PHP** – php.zip
    - **Python** – python.zip
    - **Ruby** – ruby.zip
2. Open the Elastic Beanstalk console, and in the **Regions** list, select your AWS Region.
3. In the navigation pane, choose **Environments**, and then choose the name of your environment from the list.

Note: If you have many environments, use the search bar to filter the environment list.

4. On the environment overview page, choose **Upload and deploy**.
5. Choose **Choose file**, and then upload the sample application source bundle that you downloadedThe console automatically fills in the **Version label** with a new unique label. If you type in your own version label, ensure that it's unique.
6. Choose **Deploy**.

While Elastic Beanstalk deploys your file to your Amazon EC2 instances, you can view the deployment status on the environment's overview.

While the application version is updated, the **Environment Health** status is gray. When the deployment is complete, Elastic Beanstalk performs an application health check. When the application responds to the health check, it's considered healthy and the status returns to green. The environment overview shows the new **Running**

Version—the name you provided as the **Version label**.

Elastic Beanstalk also uploads your new application version and adds it to the table of application versions. To view the table, choose **Application versions** under **getting-started-app** on the navigation pane.

Step 4: Configure your environment

You can configure your environment to better suit your application. For example, if you have a compute-intensive application, you can change the type of Amazon Elastic Compute Cloud (Amazon EC2) instance that is running your application. To apply configuration changes, Elastic Beanstalk performs an environment update.

Some configuration changes are simple and happen quickly. Some changes require deleting and recreating AWS resources, which can take several minutes. When you change configuration settings, Elastic Beanstalk warns you about potential application downtime.

Make a configuration change In this example of a configuration change, you edit your environment's capacity settings. You configure a load-balanced, scalable environment that has between two and four Amazon EC2 instances in its Auto Scaling group, and then you verify that the change occurred. Elastic Beanstalk creates an additional Amazon EC2 instance, adding to the single instance that it created initially. Then, Elastic Beanstalk associates both instances with the environment's load balancer. As a result, your application's responsiveness is improved and its availability is increased.

To change your environment's capacity

1. Open the Elastic Beanstalk console, and in the **Regions** list, select your AWS Region.
2. In the navigation pane, choose **Environments**, and then choose the name of your environment from the list.

Note:- If you have many environments, use the search bar to filter the environment list.

1. In the navigation pane, choose **Configuration**.
2. In the **Capacity** configuration category, choose **Edit**.

3. In the **Auto Scaling group** section, change **Environment type** to **Load balanced**.
4. In the **Instances** row, change **Max** to **4**, and then change **Min** to **2**.
5. Choose **Apply**.
6. A warning tells you that this update replaces all of your current instances. Choose **Confirm**.
7. In the navigation pane, choose **Events**.

The environment update can take a few minutes. To find out that it's complete, look for the event **Successfully deployed new configuration to environment** in the event list. This confirms that the Auto Scaling minimum instance count has been set to 2. Elastic Beanstalk automatically launches the second instance.

Verify the configuration change

When the environment update is complete and the environment is ready, verify your change.

To verify the increased capacity

1. In the navigation pane, choose **Health**.
2. Look at the **Enhanced health overview** page.

You can see that two Amazon EC2 instances are listed following the **Overall** line. Your environment capacity has increased to two instances.

Step 5: Clean up

Congratulations! You have successfully deployed a sample application to the AWS Cloud, uploaded a new version, and modified its configuration to add a second Auto Scaling instance. To ensure that you're not charged for any services you aren't using, delete all application versions and terminate the environment. This also deletes the AWS resources that the environment created for you.

To delete the application and all associated resources

1. Delete all application versions.
    a. Open the Elastic Beanstalk console, and in the **Regions** list, select your AWS Region.
    b. In the navigation pane, choose **Applications**, and then choose **getting-started-app**.
    c. In the navigation pane, find your application's name and choose **Application versions**.

    d. On the **Application versions** page, select all application versions that you want to delete.

    e. Choose **Actions**, and then choose **Delete**.

    f. Turn on **Delete versions from Amazon S3**.

    g. Choose **Delete**, and then choose **Done**.

2. Terminate the environment.

    a. In the navigation pane, choose **getting-started-app**, and then choose **Getting Started App-env** in the environment list.

    b. Choose **Environment actions**, and then choose **Terminate Environment**.

    c. Confirm that you want to terminate **Getting Started App-env** by typing the environment name, and then choose **Terminate**.

3. Delete the getting-started-app application.

    a. In the navigation pane, choose the **getting-started-app**.

    b. Choose **Actions**, and then choose **Delete application**.

    c. Confirm that you want to delete **getting-started-app** by typing the application name, and then choose **Delete**.

Conclusion:

Sign and Remark:

R1 (3 Marks)	R2 (2 Marks)	R3 (5 Marks)	R4 (5 Mark)	Total (15 Marks)	Signature

Dept: Computer Engineering		
Subject: Cloud Computing Lab	**Subject Code:** CSL605	
Year/Semester: TE-VI	**Date:**	**Page No.**
Student Name:	**Roll No.**	**Division:**

Sign and Remark:

R1 (3 Marks)	R2 (2 Marks)	R3 (5 Marks)	R4 (5 Mark)	Total (15 Marks)	Signature