

## Experiment No. 1

Page No.	1
Date	

Title:- Installation and Configuration of virtualization.

Aim:- To understand the concepts of virtualization and implementation of it.

Theory:-

Virtualization:-

Virtualization is the "creation of a virtual version of something, such as a server, a desk top, a storage device, an OS or network resources".

Creation of virtual machine over existing OS and hardware is known as hardware virtualization. A virtual machine provides an environment that is logically separated from the underlying hardware.

The machine on which the virtual machine is going to create is known as Host Machine and the virtual machine is referred as a Guest Machine.

Types of Virtualizations:-

i) Hardware Virtualization:-

When the virtual machine software or virtual machine manager (VMM) is directly installed on the hardware system is known as hardware virtualization.

Usage:- Controlling virtual machine is much easier.

than controlling a physical server.

### 2) Operating System Virtualization:-

When the virtual machine software or VMM is installed on HOST OS. Instead of directly on the hardware system is known as OS virtualization.

Usage:- testing the applications on different platforms of OS.

### 3) Server Virtualization:-

When the virtual machine software or VMM is directly installed on the server system is known as server virtualization.

Usage:- Single server can be divided into multiple servers on the demand basis and for balancing the load.

### 4) Storage Virtualization:-

Storage virtualization is process of grouping the physical storage from multiple network storage devices so that it looks like a single storage device.

Usage:- done for back-up and recovery purposes.

### Conclusion:-

Thus, we have performed installation and configuration of virtualization.



## Experiment No-2

Page No.	3
Date	

Title:- Implementation of Xen Server and Docker

Aim:- To study and implement Xen Server and Docker.

Theory:-

Xen Server:-

Citrix Xen Server is a hypervisor platform that enables the creation and management of virtualized server infrastructure. It is developed by Citrix Systems and is built over the Xen Virtual Machine Hypervisor. Xen Server provides server virtualization and monitoring services. It is available in a 64-bit hypervisor platform and can be executed on entire x86 series of processors.

It consolidates a physical server's computing power into multiple virtual machines, all emulating as a standard server. To provide to operational requirements of a standard server and supports most server O.S., such as Linux and Windows Server on guest server machines.

Docker:-

It is an open platform for developing, shipping and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your

infrastructure so you can do same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing and deploying code quickly, you can significantly reduce that delay between writing code and running it in production.

Docker provides the ability to package and run an application in loosely isolated environment called a container. The isolation and security allows you to run many containers simultaneously on a given host. Containers are lightweight and contain everything needed to run the application, so you don't need to rely on what is currently installed on the host. You can easily share containers while you work, and be sure that everyone you share with gets the same containers that works in the same way.

### Conclusion:-

Thus, I have successfully installed and configured Docker on Linux operating system.



### Experiment No.3

Page No.	5
Date	

Title:- Installation and Configuration of ~~Microsoft~~ Microsoft Azure App Services

Aim:- To understand the concepts of Microsoft Azure App Services and creating custom by using it.

Theory:-

Azure App Services is a fully managed PaaS provided by Microsoft Azure. It allows developers to build, deploy and manage web applications, RESTful APIs, and mobile app backends. It supports a wide range of programming languages, frameworks, and services such as .NET, Node.js, Python, PHP, Ruby, and Java. With built-in scaling, high availability, security, and integration with Azure DevOps, Azure App Services simplifies app deployment and management.

Idea:-

Azure App Services abstracts away the underlying infrastructure needed to run web apps, providing a platform where you can focus on your code. You don't need to worry about server management, networking, or storage - Azure handles those aspects, allowing developers to deploy their apps quickly and easily. It's ideal for hosting web apps, APIs, and microservices in the cloud.

Advantages:-

1. Managed Platform:- Azure handles patching, updates, and maintenance, allowing developers to focus on application development rather than infrastructure management.
2. Scalability:- Supports automatic scaling to meet traffic demands, making it easy to adjust resources as needed.
3. Language and Framework Support:- Supports multiple programming languages, frameworks, and containers, providing flexibility for developers.
4. Continuous Deployment:- Integrates with various CI/CD tools to streamline the deployment process and ensure smooth, continuous integration.
5. High Availability:- Provides built-in load balancing, traffic management, and regional failovers, ensuring high uptime and availability.
6. Cost-Effective:- Pay-as-you-go pricing allows organizations to only pay for the resources they use.

Conclusion:- Thus, I have successfully installed Microsoft Azure App Service and configured it.



## Experiment No. 4

Title:- Design an Assignment to Retrieve, verify and store credentials of users.

Aim:- To use Azure cloud services to store and retrieve user data.

Theory:-

### 1. Authentication with Azure Active Directory B2C on Azure AD:-

- Azure Active Directory B2C (Business to Customer) allows you to authenticate and manage users for web and mobile apps, similar to Firebase Authentication.
- It supports local accounts (email / password) and third party authentication providers.

Steps:- i) Set up Azure AD B2C in the Azure portal.

ii) Configure user flows for sign-up, sign-in and password reset.

iii) Integrate the AD B2C SDK into your application for user authentication.

### 2. App Hosting with Azure App Service:-

- Azure App Service as a PaaS that hosts web applications, RESTful APIs, and mobile backends, much like Google App Engine.
- You can use App Service to host the backend logic for handling user requests, authentication, and interactions with your database.

### 3. Data Storage with Azure Cosmos DB or Azure SQL Database:-

- Azure Cosmos DB or Azure SQL Database can be used to store user credentials and related information.
- Cosmos DB is a fully managed NoSQL database similar to Google cloud Firestore, while Azure SQL databases is a relational database service.

### Key Advantages of using Azure Services:-

- i) End to End Integration:- Azure provides tightly integrated services for authentication, app, hosting, and data storage.
- ii) Global Reach:- Azure has data centers worldwide enabling low-latency access for global users.
- iii) Security:- Azure services, especially ADB2C, provide robust security, including multi-factor authentication (MFA) and token based authentication.

Conclusion:- Thus, I have successfully implemented project using Azure App Services.



## Experiment No. 5

Title:- Installation and Configuration of Hadoop: Develop Map Reduce application using Hadoop cluster setup.

Aim:- To install Hadoop and implement program using Map Reduce.

Theory:-

Hadoop is an open-source framework that allows to store and process big data in a distributed environment across clusters of computers ~~and~~ using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage.

At the moment, Apache Hadoop 3.x fully supports Java 8. The Open JDK 8 package in Redhat 8 contains both the runtime environment and development kit.

- Set Up a Non-Root User for Hadoop Environment:- It is advisable to create a non-root user, specifically for the Hadoop environment. A distinct user improves security and helps you manage your clusters more efficiently. To ensure the smooth functioning of Hadoop services, the user should have the ability to establish passwordless SSH connection with the local host.
- Map-Reduce is a programming model that is

mainly divided into two phases :-

- i. Map Phase
- ii. Reduce Phase

It is designed for processing the data in parallel which is divided on various nodes (machines). The Hadoop Java programs consist of Mapper class and Reducer class along with the driver class. Hadoop Mapper is a function or task which is used to process all input records from a file and generate the output which works as input for Reducer. It produces the output by defining new key value pairs.

The input data has to be converted to key-value pairs as mapper cannot process the raw input records or tuples. The mapper also generates some small blocks of data while processing the input records as a key-value pair, we will discuss the various process that occurs in Mapper, These key feature and how the key-value pairs are generated in the Mapper. In Map Reduce word count example, we find out the frequency of each word. Here, the role of Mapper is to map the keys to the existing values and the role of Reducer is to aggregate the keys of common values.

Conclusion:- Thus, we have installed Hadoop and implemented program using MapReduce.