

PROFESSIONAL TRAINING REPORT
at
Sathyabama Institute of Science and Technology
(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Technology Degree in Information Technology

By
BOOMINATHAN.M
(40120018)



DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI, TAMILNADU – 600119

APRIL - 2023



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with "A" Grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600 119.

Phone: 044 - 2450 3150 / 51 / 52 / 54 / 55 Fax: 044 - 2450 2344
www.sathyabama.ac.in



DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of **Boominathan.M**
(40120018) who carried the project entitled "**Virtual machines with SSH key**" under my
supervision from January 2023 to April 2023.

V. Nirmalrani
28/04/2023
Internal Guide

Dr. V.Nirmalrani

for Dr. R. Subhashini
Head of the Department

Dr. R. Subhashini M.E., Ph.D.,

Submitted for Viva voce Examination held on ~~28-04-2023~~

[Signature]
Internal Examiner

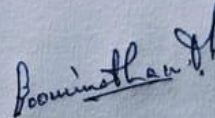
[Signature]
External Examiner

DECLARATION

I, Boominathan.M hereby declare that the Project Report entitled "**Virtual machines with ssh key**" done by me under the guidance of Dr. V.Nirmalrani is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology.

DATE: 28.04.2023.

PLACE: Chennai



SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr. R. Subhashini, M.E., Ph.D.**, Head of the Department of Information Technology for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. V.Nirmalrani** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

TRAINING CERTIFICATE

Ref.No.: 12 /FSPCC/BC001/04/2023



इलेक्ट्रॉनिक्स एवं
सूचना प्रौद्योगिकी मंत्रालय
MINISTRY OF
ELECTRONICS AND
INFORMATION TECHNOLOGY



COURSE COMPLETION LETTER

It is certified that

Boominathan M

Sathyabama Institute of Science and Technology

has successfully undergone the Bridge Course Training Program on

Cloud Computing Technology

Under "FutureSkills PRIME" Programme

commenced from 11 Mar, 2023


Shri. V. Sridharan

Chief Investigator

Cloud Computing Technology

Lead Resource Centre

L R PRAKASH

Digitally signed by L R PRAKASH
Date: 2023.04.26 03:52:40
+05'30'

Cdr.(Retd.) L.R. Prakash

Senior Director

Centre Head

C-DAC, Chennai

ABSTRACT

A virtual machine with an SSH key is a powerful tool for remote access and management of a computer system. An SSH key provides secure authentication and communication between the virtual machine and a remote client, allowing for seamless access and management of the virtual machine's resources. This abstract will discuss the benefits of using a virtual machine with an SSH key, including increased security and flexibility in managing the virtual machine. Additionally, the abstract will explore the different use cases for virtual machines with SSH keys, such as software development, testing, and deployment. Overall, virtual machines with SSH keys provide a robust and reliable solution for remote access and management of computer systems. By using an SSH key for authentication and communication, virtual machines can be accessed and managed from remote locations without compromising security. The benefits of using virtual machines with SSH keys extend beyond security, as they provide a flexible solution for a variety of use cases, including software development, testing, and deployment. With virtual machines, users can create multiple instances with different configurations. In addition to providing enhanced security, virtual machines with SSH keys also offer increased flexibility, as users can easily scale up or down depending on their needs. Overall, virtual machines with SSH keys are a reliable and powerful solution for remote access and management of computer systems, offering a range of benefits for users in various industries and applications.

TABLES OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO
	ABSTRACT	vi
	LIST OF FIGURES	ix
	LIST OF ABBREVIATION	x
1	INTRODUCTION	1
	1.1 Introduction	1
	1.2 SSH Authenticates User	2
	1.3 Aim and Scope	3
	1.3.1 Aim	3
	1.3.2 Scope	3
	1.4 Objective	4
	1.5 Limitation	5
	1.6 Assumption and Hypothesis	6
	1.7 Conclusion	6
2	AIM AND SYSTEM REQUIREMENTS	7
	2.1 Introduction	7
	2.2 system requirements	7
	2.3 Conclusion	8
3	METHODOLOGY AND TECHNOLOGIES	9
	3.1 Development Design	9
	3.2 Ssh key authentication	10
	3.2.1 Public and private keys	10
	3.2.2 Creation of virtual machine	11
	3.2.3 Creation of ssh key	14
	3.2.4 Installing cloud init	15
	3.2.5 Launching a instance	17
	3.2.6 Adding fingerprint authentication	10

4	RESULT AND DISCUSSION	20
	Sample Screenshots	20
	VM From External Network	21
5	SUMMARY AND CONCLUSION	24
	Conclusion	24
	Future Work	24
	REFERENCE	26

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
3.1	Development design	9
3.2.1	Instance details	13
3.2.2	Administrator details	13
3.2.3	Creation of ssh key	14
3.2.4	Adding cloud init	16
3.2.5	Cloud init package	17
3.2.6	Launching a instance	18
3.2.7	Adding fingerprint	19
4.1	Home Page	20
4.2	Accessing virtual machine	21
4.3	Virtual machine from external network	22
4.4	Update commands in virtual machine	22
4.5	Accessing cloud init	23

LIST OF ABBREVIATION

VM	Virtual Machine
SSH	Secure Shell
INIT	Initialization

CHAPTER-1

INTRODUCTION

INTRODUCTION:

SSH is a secure protocol used as the primary means of connecting to Linux servers remotely. It provides a text-based interface by spawning a remote shell. After connecting, all commands you type in your local terminal are sent to the remote server and executed there. In this cheat sheet-style guide, we will cover some common ways of connecting with SSH to achieve your objectives. This can be used as a quick reference when you need to know how to connect to or configure your server in different ways. Read the first if you are unfamiliar with SSH in general or are just getting started. Use whichever subsequent sections are applicable to what you are trying to achieve. Most sections are not predicated on any other, so you can use the following examples independently. Use the Contents menu on the left side of this page (at wide page widths) or your browser's find function to locate the sections you need. Copy and paste the command-line examples given, substituting the highlighted values with your own values. The most common way of connecting to a remote Linux server is through SSH. SSH stands for Secure Shell and provides a safe and secure way of executing commands, making changes, and configuring services remotely. When you connect through SSH, you log in using an account that exists on the remote server. When you connect through SSH, you will be dropped into a shell session, which is a text-based interface where you can interact with your server. For the duration of your SSH session, any commands that you type into your local terminal are sent through an encrypted SSH tunnel and executed on your server. The SSH connection is implemented using a client-server model. This means for an SSH connection to be established, the remote machine must be running a piece of software called an SSH daemon

SSH AUTHENTICATES USER:

Clients generally authenticate either using passwords (less secure and not recommended) or SSH keys, which are very secure. Password logins are encrypted and are easy to understand for new users. However, automated bots and malicious users repeatedly try to authenticate to accounts that allow password-based logins, which can lead to security compromises. For this reason, we recommend always setting up SSH key-based authentication for most configurations.

SSH keys are a matching set of cryptographic keys which can be used for authentication. Each set contains a public and a private key. The public key can be shared freely without concern, while the private key must be vigilantly guarded and never exposed to anyone. To authenticate using SSH keys, a user must have an SSH key pair on their local computer.

On the remote server, the public key must be copied to a file within the user's home directory at `~/.ssh/authorized_keys`. This file contains a list of public keys, one-per-line, that are authorized to log into this account. When a client connects to the host, wishing to use SSH key authentication, it will inform the server of this intent and will tell the server which public key to use. The server then checks its `authorized_keys` file for the public key, generates a random string, and encrypts it using the public key. This encrypted message can only be decrypted with the associated private key. The server will send this encrypted message to the client to test whether they actually have the associated private key.

Upon receipt of this message, the client will decrypt it using the private key and combine the random string that is revealed with a previously negotiated session ID. It then generates an MD5 hash of this value and transmits it back to the server. The server already had the original message and the session ID, so it can compare an MD5 hash generated by those values and determine that the client must have the private key. Now that you know how SSH works, we can begin to discuss some examples to demonstrate different ways of working with SSH

AIM AND SCOPE:

SSH keys are used in virtual machines to enhance security and provide authenticated access without the need for passwords. This is accomplished by generating an SSH key pair consisting of a public key and a private key. The public key is copied to the virtual machine, and the private key is kept in a secure location. Once the public key is added to the `authorized_keys` file in the virtual machine, users can log in without entering a password. This approach offers an additional layer of security and can help protect against brute-force attacks. Furthermore, it can make it easier to manage multiple virtual machines by eliminating the need to remember and enter passwords for each one.

SCOPE:

The scope for using SSH key pairs in virtual machines is quite broad and can be applied to a variety of use cases. Here are a few examples.

- **Server administration:** In a server environment, administrators can use SSH key pairs to securely log in to virtual machines to manage server configurations, perform updates, and troubleshoot issues.
- **Development and testing:** Developers and testers often use virtual machines to develop and test software applications. Using SSH key pairs, developers can access virtual machines securely without the need for passwords, which can help reduce the risk of unauthorized access or data breaches.
- **Cloud computing:** SSH key pairs are commonly used in cloud computing environments where virtual machines are frequently provisioned and decommissioned. This approach helps manage access to virtual machines securely, making it easier to manage multiple virtual machines in a cloudbased infrastructure.

- DevOps automation: In DevOps environments, SSH key pairs can be used to automate tasks such as code deployment and configuration management. Using SSH key pairs allows for secure, automated access to virtual machines, reducing the need for manual intervention and streamlining the development process.
- Overall, the scope for using SSH key pairs in virtual machines is quite extensive and can be applied to a wide range of use cases. By providing secure, authenticated access, SSH key pairs help ensure the integrity and security of virtual machines, making them an essential tool for many organizations.

OBJECTIVE:

The objective of using virtual machines with SSH key and access in an external network is to provide secure and authenticated access to virtual machines from outside the network.

Here are a few steps to achieve this objective:

- Configure the virtual machine: First, you need to configure the virtual machine to allow SSH access from external networks. This involves configuring the virtual machine's network settings to allow incoming traffic on the SSH port (usually port 22).
- Generate an SSH key pair: Next, you need to generate an SSH key pair on the client machine from which you will be accessing the virtual machine. This involves using the `ssh-keygen` command to generate the key pair, and then copying the public key to the virtual machine.
- Configure firewall rules: To ensure the security of the network, you need to configure firewall rules to allow incoming traffic only on the SSH port and block all other incoming traffic.
- Test SSH access: Once the above steps are completed, you can test SSH access to the virtual machine from outside the network using the private key generated in step 2.

By following these steps, you can achieve the objective of providing secure and authenticated access to virtual machines in an external network. This approach provides an added layer of security and can help protect against unauthorized access and data breaches. It also enables remote access to virtual machines, making it easier to manage and maintain virtualized infrastructure.

LIMITATION:

Virtual machines (VMs) with SSH key authentication have several limitations, some of which include:

Increased complexity: SSH key authentication requires generating and managing a public/private key pair, which can be more complex than traditional password-based authentication. Dependency on key management: The security of SSH key authentication relies heavily on the management of the private key. If the key is compromised or lost, access to the VM can be compromised.

Limited access control: SSH key authentication only provides authentication, it does not provide authorization. Therefore, access control policies such as granting or revoking access to a user or group of users is not possible through SSH key authentication alone.

Lack of visibility: SSH key authentication does not provide a way to track user activity or monitor user behavior on the VM.

Risk of unauthorized access: If an unauthorized user gains access to a user's private key, they can use it to gain access to the VM.

ASSUMPTION AND HYPOTHESIS:

Assumption:

The Virtual Machine (VM) is hosted on a cloud service provider, such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform (GCP).

- The VM is running a Linux operating system.
- The SSH key is the preferred method of authentication to access the VM.
- The SSH key has been securely generated and properly stored on the client machine.
- The client machine has a secure network connection to the VM.

Hypothesis:

- Using SSH key authentication to access the VM is more secure than using a username and password combination.
- The use of SSH key authentication reduces the risk of brute force attacks on the VM.
- The use of SSH key authentication reduces the risk of password guessing attacks on the VM.
- SSH key authentication allows for a more streamlined and efficient access to the VM, as users do not need to remember or enter a password each time they connect.
- SSH key authentication allows for granular control over access to the VM, as individual keys can be added or removed as necessary.

CHAPTER-2

AIM AND SYSTEM REQUIREMENTS

AIM:

To access a virtual machine (VM) from an external network using an SSH key

.

SYSTEM REQUIREMENTS:

general guidelines for system requirements for a VM to access from an external network with SSH key:

- **Processor:** The VM should have a processor with at least two cores to handle the demands of the operating system and any applications running on the VM.
- **Memory (RAM):** The VM should have at least 4 GB of RAM to ensure that it can handle the demands of the operating system and any applications running on the VM.
- **Storage:** The VM should have adequate storage space to store the operating system and any applications that will be running on the VM. A minimum of 50 GB of storage is recommended.
- **Operating System:** The operating system should be a supported version of Linux or Windows Server that is up-to-date with security patches.
- **Networking:** The VM should have a network interface card (NIC) that is configured to allow external access. The NIC should be configured with a static IP address that can be accessed from external networks.

- SSH Server: The VM should have an SSH server installed and configured to allow external access using SSH keys.
- Firewall: The VM's firewall should be configured to allow incoming SSH traffic on the appropriate port (usually port 22).

CHAPTER-3

METHODOLOGY

The project methodology is important because it helps to organize investigation in a scientific way to overcome problems, structure, plan, and control the process of developing an information system. The system will use Iterative Model development as a framework methodology. After system has been completely developed, it should be tested to make sure it achieves the objectives of the project.

DEVELOPMENT DESIGN:

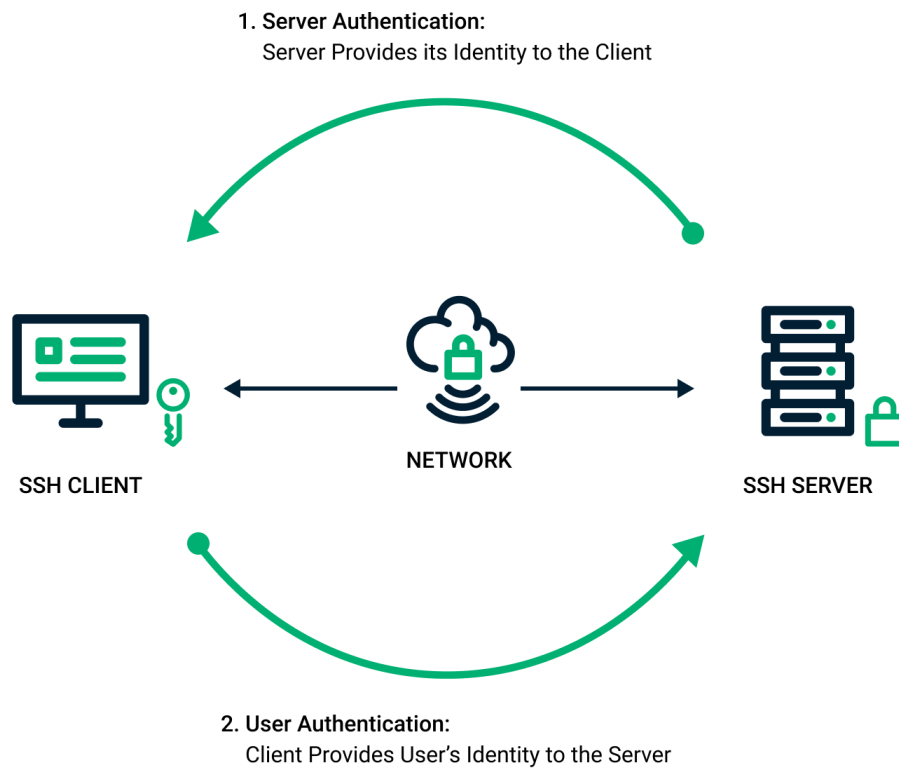


Fig 3.1 Development desing

SSH KEY AUTHENTICATION:

At a high level, SSH keys function like passwords by controlling access. But that's where the similarity ends. Digital identities need to be strong so that they cannot be stolen, convenient so that access is fast and never interrupted, and future-proof so that enterprises can stay ahead of threats.

Passwords used to offer a measure of security, but they are not as effective as they once were. That's because bad actors have become increasingly adept at stealing passwords in transit through the internet, lifting them from repositories, and obtaining them through brute force attacks.

These issues are compounded by the human factor: that people reuse, share, and continually forget passwords. The cost to businesses is high, not only because of data loss, outages, and the compromise of information, but also because passwords take time to enter, require significant support resources, and must be frequently reset.

Using PKI's cryptographic strength and authentication methods, SSH keys are not prone to malicious attacks and valid credentials are not exposed if a system or server has been compromised. PKI cryptography is being improved continuously, future-proofing identities against new and evolving threats. And SSH key authentication is more convenient than password authentication. The keys connect users and processes to a server by initiating authentication and granting access automatically, so users don't have to remember or enter their password for each and every system.

PUBLIC AND PRIVATE KEYS :

An SSH key relies upon the use of two related keys, a public key and a private key, that together create a key pair that is used as the secure access credential. The private key is secret, known only to the user, and should be encrypted and stored safely. The public key can be shared freely with any SSH server to which the user wishes to connect.

These keys are normally managed by an organization's IT team, or better yet, with the help of a trusted Certificate Authority (CA) to ensure they are stored safely.

To create the digital identity, the public and private key are both generated, and the pair is associated with each other using a strong public key cryptography algorithm.

The most common mathematical algorithms used to generate the keys are Rivest–Shamir–Adleman (RSA) and Elliptic Curve Digital Signature Algorithm (ECDSA).

These algorithms use various computation methods to generate random numeric combinations of varying length so that they cannot be exploited with a brute force attack. The key size or bit length of SSH keys helps determine the strength of protection. 2048-bit RSA keys or 521-bit ECDSA keys offer sufficient cryptographic strength to keep hackers from cracking the algorithm.

CREATION OF VIRTUAL MACHINE:

To create a Virtual Machine (VM) in Microsoft Azure, follow these steps:

- Sign in to the Azure Portal (portal.azure.com) using your Microsoft account.
- Click on the “Create a resource” button located on the left-hand side of the screen.
- Search for “Windows Server” or “Linux” in the Azure Marketplace and select the operating system you want to use.
- Choose the version, size, and other details of the VM you want to create.
- Configure the virtual network and subnet settings for the VM.
- Set up the authentication and administrative credentials for the VM.
- Review the settings and click on the “Create” button to start the VM deployment process.

Microsoft Azure (Preview)

Search resources, services, and docs (G+)

Home > Virtual machines >

Create a virtual machine

Basics

Disks

Networking

Management

Advanced

Tags

Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

COSINE-ES-FailureAnalysis

Resource group *

(New) Resource group

Create new

Instance details

Virtual machine name *

Cosine2

Region *

(US) East US

Availability options

No infrastructure redundancy required

Image *

Ubuntu Server 18.04 LTS - Gen1

Browse all public and private images

Azure Spot instance

☐ Yes
☒ No

Size *

Standard_D2s_v3 - 2 vcpus, 8 GiB memory (\$xx.xx/month)

Select size

Review + create

< Previous

Next : Disks >

Fig 3.2 Creating a virtual machine

- If you have not chosen an image before, you can choose the right image from the dropdown list. You can also browse all available images for creating virtual machines, including both public and private images. It's better to go for Data Centre for the image.

Instance details

Virtual machine name * ⓘ myVM ✓

Region * ⓘ (US) East US ✓

Availability options ⓘ No infrastructure redundancy required ✓

Image * ⓘ Windows Server 2019 Datacenter - Gen1 ✓
[See all images](#)

Azure Spot instance ⓘ ☐

Size * ⓘ Standard_DS1_v2 - 1 vcpu, 3.5 GiB memory ✓
[See all sizes](#)

Fig 3.2.1 Instance details

- Carefully select the size of the virtual machine to be deployed. Here you will find different virtual machine size options in accordance with your basic configuration. Note that you cannot find all virtual machine sizes in all the Azure regions. But a safe choice can be Standard_DS1_v2.
- Fill in your username and password under the Administrator account. Make sure that your password is at least 12 characters long and also meets the specified complexity requirements.

Administrator account

Username * ⓘ azureuser ✓

Password * ⓘ ✓

Confirm password * ⓘ ✓

Fig 3.2.2 Administrator Details

CREATION OF SSH KEY:

- Open the Azure portal.
- At the top of the page, type SSH to search. Under Marketplace, select SSH keys.
- On the SSH key page, select Create.

Home > SSH Key >

Create an SSH key

Basics Tags Review + create

Creating an SSH key resource allows you to manage and use public keys stored in Azure with Linux virtual machines.
[Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Pay-As-You-Go

Resource group * ⓘ (New) mySSHKeyGroup
[Create new](#)

Instance details

Region * ⓘ (US) East US

Key pair name * mySSHKey1 ✓

SSH public key source Generate new key pair

[Review + create](#) < Previous Next : Tags >

Fig 3.2.3 Creation of ssh key

- In Resource group select Create new to create a new resource group to store your keys. Type a name for your resource group and select OK.
- In Region select a region to store your keys. You can use the keys in any region, this option is just the region where you store them.
- Type a name for your key in Key pair name.
- In SSH public key source, select Generate public key source.
- When you're done, select Review + create.
- After it passes validation, select Create.

- You'll get a pop-up window to, select Download private key and create resource that downloads the SSH key as a .pem file.



Fig 3.2.3 Generate new key pair

- Once you've downloaded the .pem file, you might want to move it somewhere on your computer where it's easy to point to from your SSH client.

INSTALLING CLOUD-INIT PACKAGES:

When you create a VM, certificates and keys are stored in the protected `/var/lib/waagent/` directory. To automate adding the certificate to the VM and configuring the web server, use cloud-init. In this example, you install and configure the NGINX web server. You can use the same process to install and configure Apache.

Create a file named `cloud-init-web-server.txt` and paste the following configuration:

[Home](#) > [Virtual machines](#) >

Add a virtual machine

myasegpudevice

Basics

Disks

Networking

Advanced

Review + create

Add additional configuration, agents, scripts or applications via virtual machine extensions or cloud-init.

Extensions

Extensions provide post-deployment configuration and automation.

For a Red Hat image, follow the [steps to add the GPU extension](#) to the VM. Add the extension after the VM is created.

Extensions ⓘ

NVIDIA GPU Driver Extension

Microsoft.HpcCompute

Select an extension to install

Custom data and cloud init

Pass a cloud-init script, configuration file, or other data into the virtual machine while it is being provisioned. The data will be saved on the VM in a known location. [Learn more about custom data for VMs](#)

Custom data

```
#cloud-config
package_upgrade: true
packages:
  - nginx
  - nodejs
  - npm
write_files:
```

ⓘ

Custom data on the selected image will be processed by cloud-init. [Learn more about custom data and cloud init](#)

Review + create

Previous

Next: Review + create

Fig 3.2.4 Adding cloud init

On the **Review + Create** tab, review the specifications for the VM. Then select **Create**.

```
#cloud-config
package_upgrade: true
packages:
  - nginx
write_files:
  - owner: www-data:www-data
    path: /etc/nginx/sites-available/default
    content: |
      server {
        listen 443 ssl;
        ssl_certificate /etc/nginx/ssl/mycert.cert;
        ssl_certificate_key /etc/nginx/ssl/mycert.prv;
      }
runcmd:
  - secretsname=$(find /var/lib/waagent/ -name "*.prv" | cut -c -57)
  - mkdir /etc/nginx/ssl
  - cp $secretsname.crt /etc/nginx/ssl/mycert.cert
  - cp $secretsname.prv /etc/nginx/ssl/mycert.prv
  - service nginx restart
```

Fig 3.2.5 Cloud init package

LAUNCHING A INSTANCE:

Here are the steps to launch an existing Virtual Machine (VM) in Azure:

- Login to the Azure portal at <https://portal.azure.com/>.
- Navigate to the Virtual Machines blade by clicking on “Virtual Machines” in the left-hand navigation menu.
- Locate the existing VM you want to launch and click on its name to open its Overview page.
- Click on the “Start” button in the top toolbar to start the VM.

nginx-vm

Virtual machine

Search

Connect Start Restart Stop Capture Delete Refresh Open in mobile Feedback CLI / PS

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Networking

Connect

Disks

Size

Microsoft Defender for Cloud

Advisor recommendations

Extensions + applications

Continuous delivery

Availability + scaling

Configuration

Identity

Properties

Locks

Operations

Bastion

Auto-shutdown

Backup

Disaster recovery

Essentials

Resource group (move)

my-nginx-rg

Status

Running

Location

North Europe

Subscription (move)

Free Trial

Subscription ID

598b67e8-f10a-4405-89ae-417afd201194

Tags (edit)

Click here to add tags

Operating system

Linux (ubuntu 20.04)

Size

Standard D2s v3 (2 vcpus, 8 GiB memory)

Public IP address

104.41.222.85

Virtual network/subnet

nginx-vm-vnet/default

DNS name

Not configured

JSON View

Properties

Monitoring

Capabilities (7)

Recommendations

Tutorials

Virtual machine

Computer name

nginx-vm

Health state

-

Operating system

Linux (ubuntu 20.04)

Publisher

canonical

Offer

0001-com-ubuntu-server-focal

Plan

20_04-lts-gen2

VM generation

V2

VM architecture

x64

Agent status

Ready

Agent version

2.9.0.4

Host group

None

Host

-

Proximity placement group

-

Colocation status

N/A

Capacity reservation group

-

Availability + scaling

Availability zone

-

Networking

Public IP address

104.41.222.85 (Network interface nginx-vm963)

Public IP address (IPv6)

-

Private IP address

10.1.0.4

Private IP address (IPv6)

-

Virtual network/subnet

nginx-vm-vnet/default

DNS name

Configure

Size

Size

Standard D2s v3

vCPUs

2

RAM

8 GiB

Disk

OS disk

nginx-vm_OsDisk_1_8c5ca4431a3043ac967b1341356c0f28

Encryption at host

Disabled

Azure disk encryption

Not enabled

Ephemeral OS disk

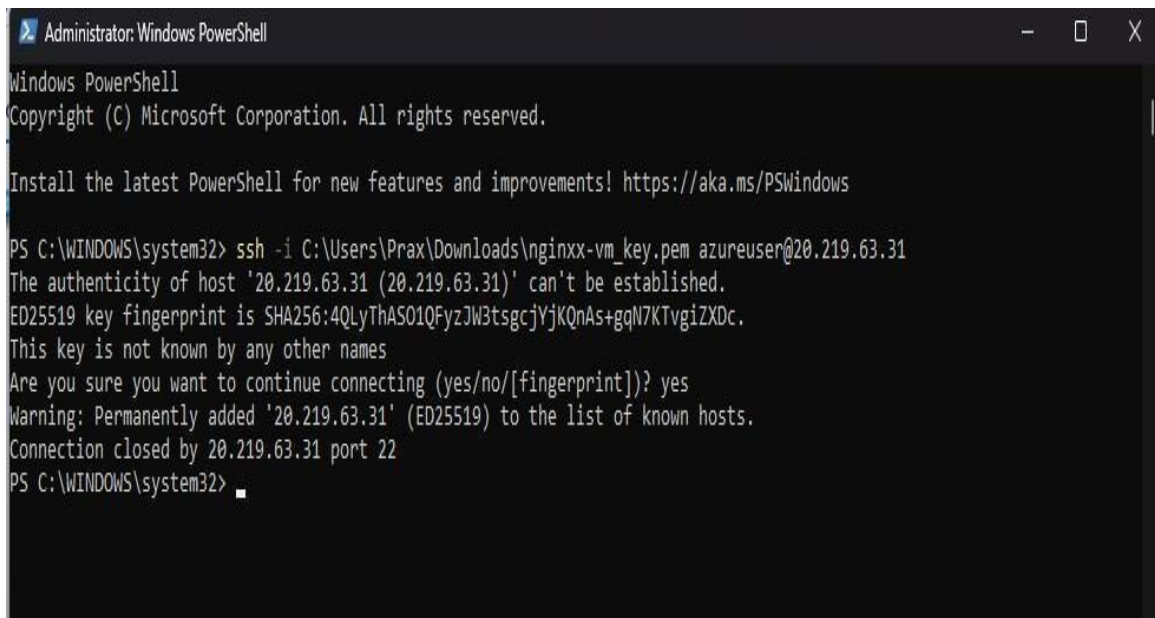
N/A

Data disks

0

Fig 3.2.6 Launching a instance

ADDING FINGERPRINT AUTHENTICATION:



```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\WINDOWS\system32> ssh -i C:\Users\Prax\Downloads\nginx-vm_key.pem azureuser@20.219.63.31
The authenticity of host '20.219.63.31 (20.219.63.31)' can't be established.
ED25519 key fingerprint is SHA256:4QLyThAS01QFyzJW3tsgcjVjKQnAs+ggN7KTvgiZXDc.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '20.219.63.31' (ED25519) to the list of known hosts.
Connection closed by 20.219.63.31 port 22
PS C:\WINDOWS\system32> .
```

Fig 3.2.7 Adding fingerprint

CHAPTER-4

RESULT AND DISCUSSION

SAMPLE SCREENSHOTS:

Implementing and testing are the important parts in order to produce the good quality of system. It refers to the process of testing implementations of technology specifications

HOME PAGE:

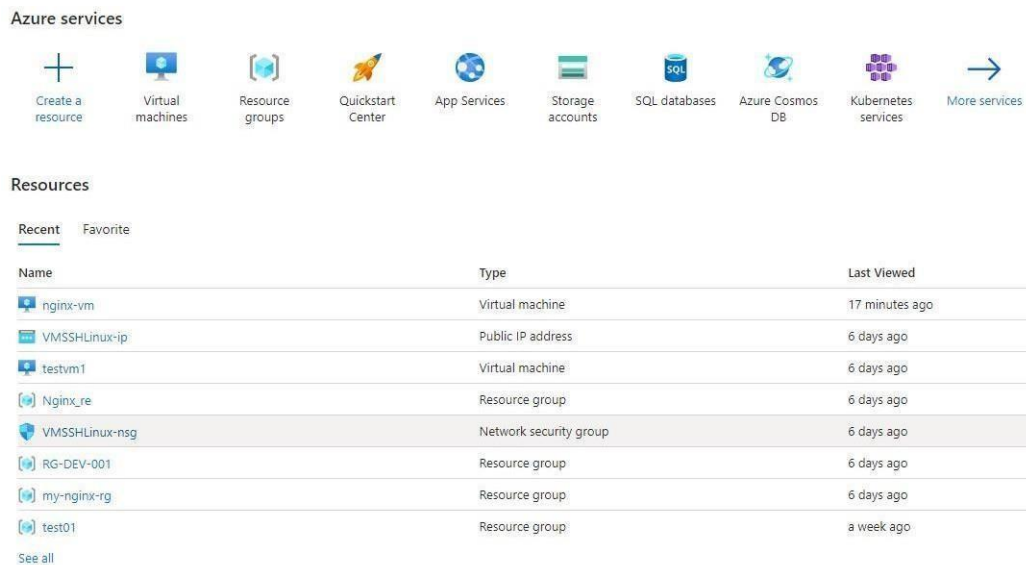


Fig 4.1 Home Page

VM FROM EXTERNAL NETWORK:

```
cs: Select azureuser@nginx-vm: ~
Microsoft Windows [Version 10.0.22621.1555]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>ssh -i C:\Users\Prax\Downloads\104.41.222.85
ssh: Could not resolve hostname -: No such host is known.

C:\Windows\System32>ssh -i C:\Users\Prax\Downloads\nginx-vm_key.pem azureuser@104.41.222.85
The authenticity of host '104.41.222.85 (104.41.222.85)' can't be established.
ED25519 key fingerprint is SHA256:vmDAdVaRebhG8nFAv6EfemUNTSObKmqrNYwTxDFdFq4.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '104.41.222.85' (ED25519) to the list of known hosts.
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-1035-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Mon Apr 17 12:42:12 UTC 2023

System load:  0.0               Processes:            127
Usage of /:   5.9% of 28.89GB   Users logged in:     0
Memory usage: 4%               IPv4 address for eth0: 10.1.0.4
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

azureuser@nginx-vm:~$
```

Fig 4.2 Accessing virtual machine

```
Select azureuser@nginx-vm: ~

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

* Introducing Expanded Security Maintenance for Applications.
Receive updates to over 25,000 software packages with your
Ubuntu Pro subscription. Free for personal use.

https://ubuntu.com/azure/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '22.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Apr 17 12:42:14 2023 from 183.82.204.253
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

azureuser@nginx-vm:~$
```

Fig 4.3 Vm from external network

```
azureuser@nginx-vm: ~

5 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

azureuser@nginx-vm:~$ apt list --upgradable
Listing... Done
ubuntu-advantage-tools/focal-updates 27.14.4~20.04 amd64 [upgradable from: 27.13.6~20.04.1]
vim-common/focal-security 2:8.1.2269-1ubuntu5.14 all [upgradable from: 2:8.1.2269-1ubuntu5.13]
vim-runtime/focal-security 2:8.1.2269-1ubuntu5.14 all [upgradable from: 2:8.1.2269-1ubuntu5.13]
vim-tiny/focal-security 2:8.1.2269-1ubuntu5.14 amd64 [upgradable from: 2:8.1.2269-1ubuntu5.13]
vim/focal-security 2:8.1.2269-1ubuntu5.14 amd64 [upgradable from: 2:8.1.2269-1ubuntu5.13]
xxd/focal-security 2:8.1.2269-1ubuntu5.14 amd64 [upgradable from: 2:8.1.2269-1ubuntu5.13]
azureuser@nginx-vm:~$
```

Fig 4.4 update commands in virtual machine



Fig 4.5 accessing cloud init

CHAPTER-5

SUMMARY AND DISCUSSION

Conclusion:

In conclusion, using virtual machines with SSH keys offers a secure and efficient way to manage and access remote servers. By leveraging SSH keys, users can authenticate their identity without the need for passwords, which can be vulnerable to attacks like brute-force attacks. Additionally, virtual machines provide a flexible and scalable infrastructure that can be easily provisioned and managed, making them an ideal choice for many organizations. Overall, incorporating virtual machines with SSH keys can enhance security and productivity in a variety of use case, making them a valuable tool for many businesses and individuals.

Future work:

In terms of future work, there are several areas where virtual machines with SSH keys could be further improved. One potential avenue for research is in the development of more advanced authentication mechanisms that can provide even greater security and protection against potential threats. Additionally, there is a growing need for virtual machines that can support a wider range of workloads and applications, which may require the development of new virtualization technologies or more efficient resource management strategies. Another area of potential future work is in the integration of virtual machines with other emerging technologies such as containerization, serverless computing, and edge computing. By combining these technologies with virtual machines and SSH keys, it may be possible to create even more flexible and scalable computing environments that can support a wider range of use cases.

Finally, there is a need for continued research into the best practices for managing and securing virtual machines with SSH keys, particularly as new threats and vulnerabilities are discovered. Ongoing efforts to develop and refine these best practices will be essential for ensuring that virtual machines remain a secure and reliable solution for remote server management and access.

There are several areas where SSH (Secure Shell) could be further improved in the future. One potential area of focus is in enhancing the performance and scalability of SSH connections, particularly for high-traffic environments. This could involve the development of new protocols or more efficient encryption algorithms that can better handle large volumes of data.

Another area of potential future work is in improving the user experience of SSH, particularly for non-technical users. This could involve the development of more intuitive interfaces or the implementation of more user-friendly authentication mechanisms that don't require a deep understanding of encryption or network security.

In terms of security, there is a need for continued research and development of new encryption algorithms and authentication mechanisms that can provide even greater protection against potential threats. This may include the development of new approaches to key management, more advanced techniques for detecting and preventing attacks, or the implementation of new security protocols that can better protect against emerging threats.

REFERENCE

1. Karen Scarfone and Paul Hoffman, Guidelines on Firewalls and Firewall Policy, National Institute of Standards and Technology, NIST Special Publication 80041 Revision 1.
2. Whyman, B.: Cloud Computing, Information Security and Privacy Advisory Board, pp.11–13 (December 5, 2008).
3. J Kich, Virtual Machine Security Guidelines Version 1.0. The Center for Internet Security, September 2007.
4. B. R. Kandukuri, R. P. V. and A. Rakshit, "Cloud Security Issues" in Proceedings of the 2009 IEEE International Conference on Services Computing, 2009, pp. 517-520.