

CREATING A VM ON AZURE
AND
HOSTING A STATIC WEBSITE

ABSTRACT

This project aims to create a Virtual Machine (VM) on Microsoft Azure and utilize HTML and CSS to develop a static website. The primary steps involve setting up the VM, configuring it for web hosting, and deploying the static website. Additionally, the project will explore the optional configurations of linking a custom domain and enabling SSL for secure communication.

The initial phase focuses on provisioning a VM on Azure, selecting appropriate resources, and ensuring secure access through SSH. Once the VM is operational, the next step involves setting up a web server environment, typically using Nginx or Apache, to host the static website.

The development of the static website entails creating HTML files for structure and CSS files for styling. This website will be transferred to the Azure VM and configured for public access.

To enhance the website's accessibility and security, the project will cover the process of purchasing and configuring a custom domain. Integrating SSL certificates will ensure encrypted connections, boosting the website's security and trustworthiness.

By the end of this project, a fully functional static website will be hosted on an Azure VM, with an optional custom domain and SSL configuration, demonstrating a complete end-to-end deployment process.

OBJECTIVE

The objective of this project is to provide a comprehensive guide to creating a Virtual Machine (VM) on Microsoft Azure, developing a static website using HTML and CSS, and hosting this website on the VM. Additionally, the project aims to cover the optional steps of configuring a custom domain and enabling SSL for enhanced security. The specific objectives are as follows:

1. Provision a Virtual Machine on Azure:

- Select and configure the appropriate VM resources.
- Ensure secure access to the VM via SSH.

2. Set Up a Web Server:

- Install and configure a web server (Nginx or Apache) on the Azure VM.
- Prepare the server environment for hosting a static website.

3. Develop a Static Website:

- Create HTML files for the website's structure.
- Design and implement CSS for styling the website.

4. Deploy the Static Website:

- Transfer the website files to the Azure VM.
- Configure the web server to serve the static website publicly.

5. Configure a Custom Domain:

- Purchase or use an existing custom domain.
- Set up domain name system (DNS) settings to point to the Azure VM.

6. Optional: Enable SSL:

- Obtain and install an SSL certificate.
- Configure the web server to use SSL for secure communication.

By achieving these objectives, the project will demonstrate a complete end-to-end process of hosting a static website on an Azure VM, with optional enhancements for domain configuration and security.

INTRODUCTION

The digital landscape of today demands that individuals and businesses maintain a strong online presence. A crucial part of this presence is a well-designed and accessible website. Hosting a website on a reliable and scalable platform like Microsoft Azure offers numerous benefits, including high availability, security, and performance. This project guide aims to assist individuals in navigating the process of creating and hosting a static website on Azure using HTML and CSS.

Azure Cloud Computing

Microsoft Azure, often referred to simply as Azure, is a cloud computing platform and service created by Microsoft. Launched in 2010, Azure has grown to become one of the leading cloud service providers, competing with other major platforms like Amazon Web Services (AWS) and Google Cloud Platform (GCP). Azure offers a wide array of services that support various programming languages, frameworks, and tools, providing developers and businesses with the flexibility to build, deploy, and manage applications and services.

Benefits of Azure:

1. Wide Range of Services:

Compute Services: Azure offers various compute options, including Virtual Machines (VMs), Azure Kubernetes Service (AKS), and Azure Functions. These services allow users to run applications, host websites, and manage containerized workloads.

Storage Services: Azure provides scalable and secure storage solutions such as Azure Blob Storage, Azure File Storage, and Azure Disk Storage, catering to different data storage needs.

Database Services: Azure supports multiple database options, including Azure SQL Database, Azure Cosmos DB, and Azure Database for PostgreSQL, MySQL, and MariaDB.

Networking Services: Azure's networking capabilities include Virtual Networks (VNets), Azure Load Balancer, Azure DNS, and Azure Content Delivery Network (CDN), ensuring reliable and high-performance connectivity.

AI and Machine Learning: Azure AI services, such as Azure Cognitive Services and Azure Machine Learning, enable users to build intelligent applications with advanced analytics and machine learning models.

DevOps and Development Tools: Azure DevOps provides a suite of tools for continuous integration and continuous delivery (CI/CD), while Azure provides integrated development environments (IDEs) and SDKs for various programming languages.

2. Global Presence:

Azure operates in multiple regions worldwide, providing data centers in over 60 regions. This extensive network ensures low-latency access to resources and data redundancy for disaster recovery.

3. Security and Compliance:

Azure emphasizes security and compliance, offering robust security features such as Azure Security Center, Azure Active Directory, and Azure Key Vault. It also complies with various international standards and regulations, making it suitable for industries with strict compliance requirements.

4. Scalability and Flexibility:

Azure allows users to scale their resources up or down based on demand, ensuring cost-efficiency and performance optimization. The platform supports a wide range of operating systems, programming languages, frameworks, and tools, giving users the flexibility to choose the best technologies for their projects.

5. Cost Management:

Azure provides tools for monitoring and managing cloud expenditure, such as Azure Cost Management and Azure Advisor. These tools help users optimize their resource usage and control costs effectively.

6. Integration with Microsoft Ecosystem:

As a Microsoft product, Azure integrates seamlessly with other Microsoft services and products, such as Office 365, Dynamics 365, and Power BI. This integration provides a cohesive experience for businesses already using Microsoft technologies.

Popular Use Cases for Azure:

1. Web and Mobile App Development:

Azure App Service allows developers to build, deploy, and scale web and mobile applications quickly. It supports multiple programming languages and frameworks, including .NET, Java, PHP, Node.js, and Python.

2. Data Analytics and Business Intelligence:

Azure provides robust data analytics and business intelligence tools, such as Azure Synapse Analytics and Power BI, enabling organizations to gain insights from their data and make informed decisions.

3. Internet of Things (IoT):

Azure IoT Hub and Azure IoT Central provide a comprehensive platform for building and managing IoT solutions, from device connectivity to data analysis and visualization.

4. Hybrid Cloud and On-Premises Integration:

Azure Arc and Azure Stack allow organizations to extend Azure services to their on-premises data centers, enabling hybrid cloud deployments and seamless integration between cloud and on-premises environments.

5. Artificial Intelligence and Machine Learning:

Azure's AI and machine learning services enable developers to build intelligent applications with capabilities such as natural language processing, image recognition, and predictive analytics.

By leveraging Azure's extensive range of services and features, organizations can accelerate their digital transformation, enhance operational efficiency, and drive innovation in a cost-effective and scalable manner.

Azure, a cloud computing service by Microsoft, provides an extensive range of services and resources to build, deploy, and manage applications. One of its fundamental offerings is the Virtual Machine (VM), which allows users to run their own server in the cloud. By leveraging Azure VMs, users can create a customized and secure environment for their websites.

A static website, built with HTML and CSS, is an excellent starting point for those looking to establish an online presence. Unlike dynamic websites, static websites are straightforward to create and host, as they do not require server-side processing. This simplicity makes them ideal for portfolios, informational sites, and other content-focused platforms.

This guide will walk through the steps of creating a VM on Azure, setting up a web server, and deploying a static website. Additionally, it will cover optional advanced topics, such as configuring a custom domain and enabling SSL certificates for enhanced security. These steps ensure that the website is not only accessible but also secure, providing visitors with a trustworthy experience.

By following this guide, users will gain practical experience in cloud computing, web development, and web hosting. The skills acquired will be valuable for personal projects and professional endeavors in the ever-evolving digital world.

METHODOLOGY

The methodology for this project involves a systematic approach to create and host a static website on an Azure Virtual Machine (VM), with optional configurations for a custom domain and SSL.

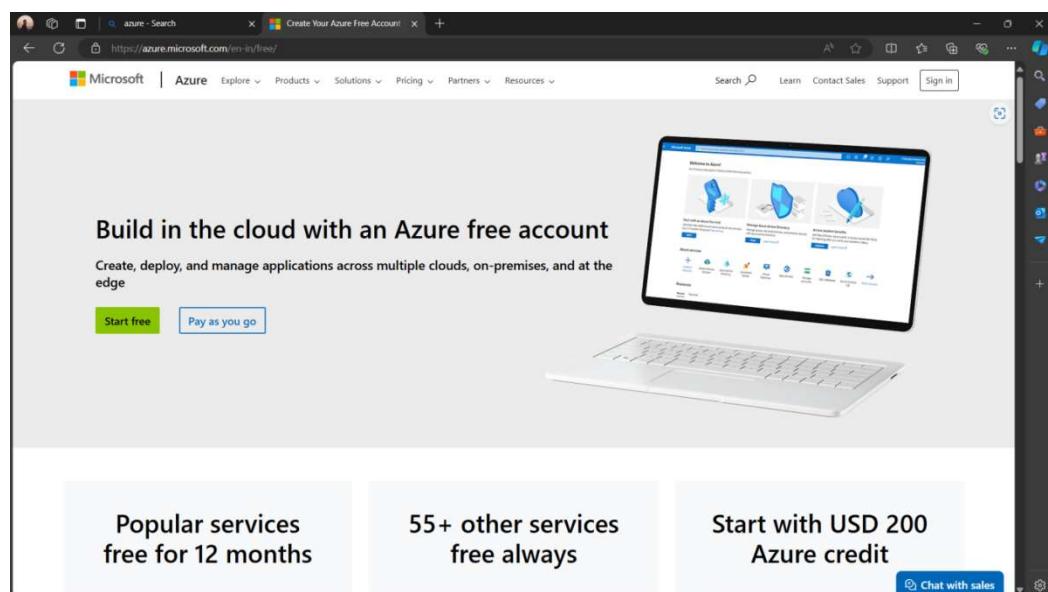
In Microsoft Azure, you can easily upload your static website and share the link anywhere and anyone can see your website, you can manage your access control. Html, CSS you can put your code on the storage container. Azure provides a static website option for user can serve their website in azure.

You can enable your static website in azure free of cost but only you will pay for the storage account. Azure uses the encryption technique for storing the data. The process is divided into distinct phases, each with specific steps and proper screenshots.

Phase 1: Setting Up the Server:

1. Create an Azure Account:

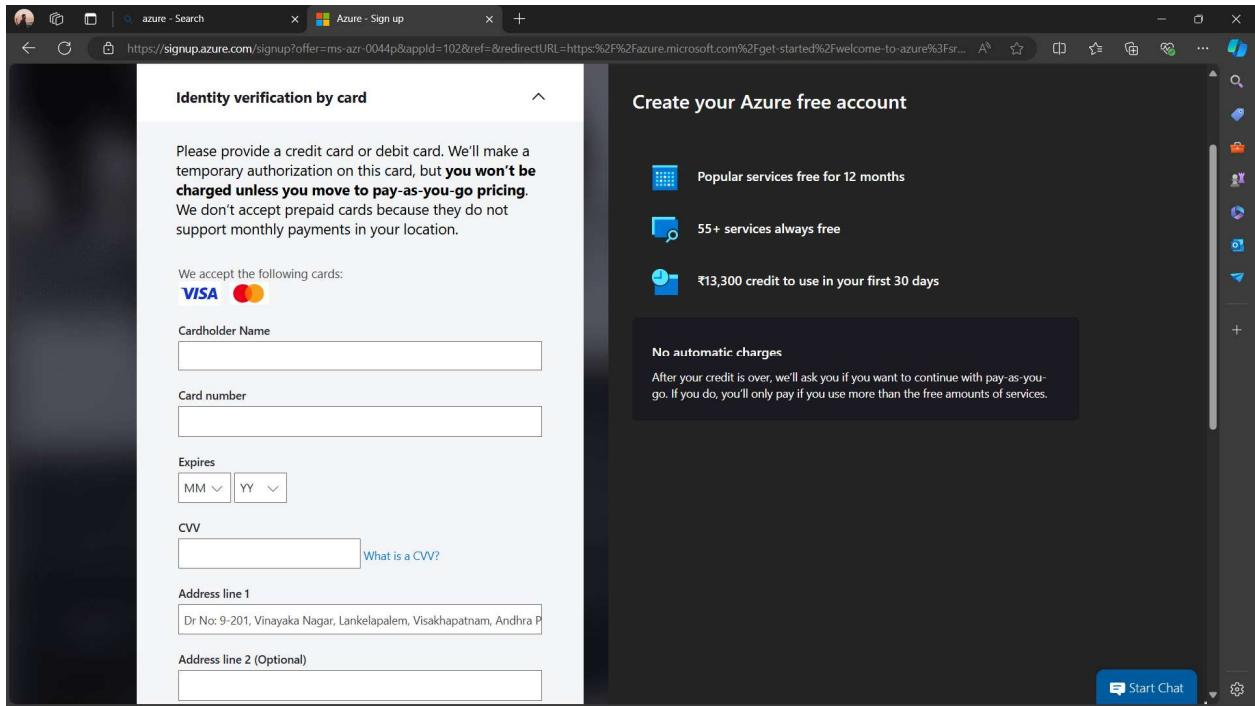
- Sign up for an Azure account at the Azure portal (<https://portal.azure.com>).



- Set up billing information and explore the available free tier options.

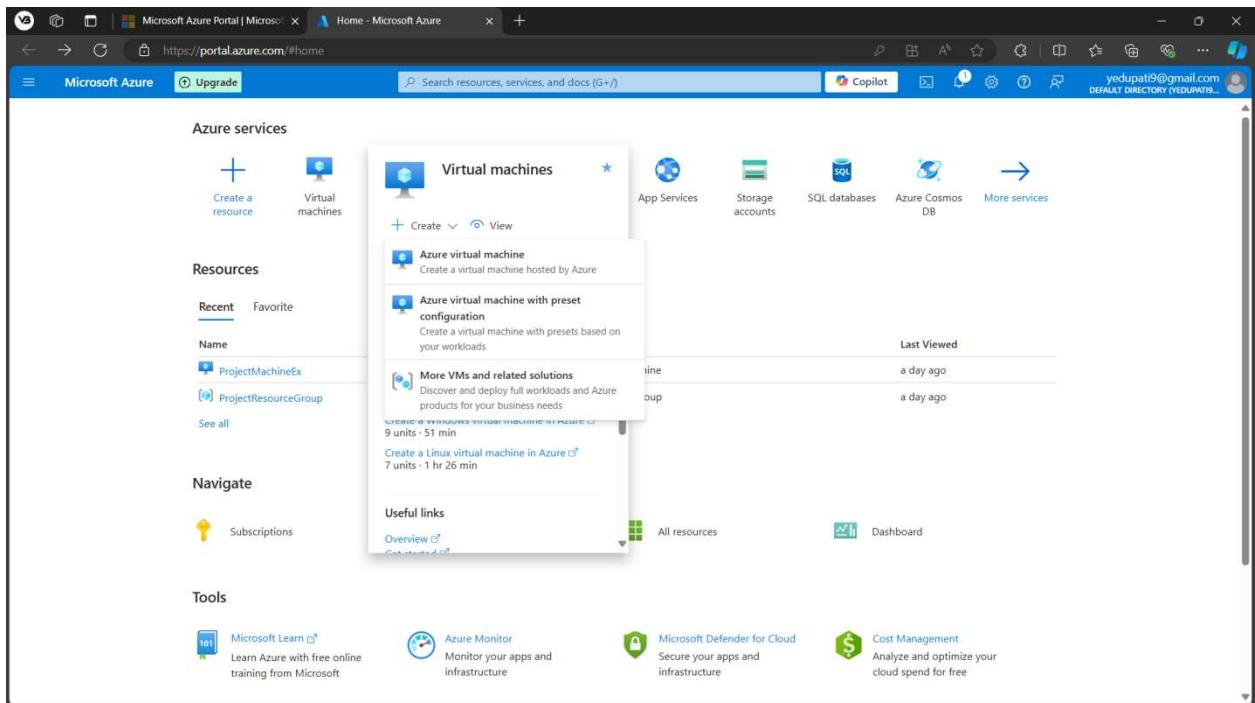
The screenshot shows the Microsoft Azure Sign Up interface. On the left, there's a sidebar titled "Your profile" with fields for Country/Region (set to India), First name, Middle name (Optional), Last name, Email address (unbishav8@gmail.com), and Phone (Example: 81234-56789). Below these fields is a checkbox for "Use a different phone number to verify your identity." On the right, there's a summary titled "Create your Azure free account" which highlights "Popular services free for 12 months", "55+ services always free", and "₹13,300 credit to use in your first 30 days". A note below states "No automatic charges" and "After your credit is over, we'll ask you if you want to continue with pay-as-you-go. If you do, you'll only pay if you use more than the free amounts of services." At the bottom right of the main area is a "Start Chat" button.

The screenshot shows the continuation of the Azure Sign Up process. It includes fields for State (dropdown menu showing "--Select--") and Postal Code (text input field). Below these fields is a consent section with a note: "I understand that Microsoft may contact me about my free account." and a checkbox for "I agree to the [subscription agreement, offer details](#)". Further down, another checkbox allows users to "I would like Microsoft to share my information with select partners so I can receive relevant information about their products and services." At the bottom of this section is a link to the "Privacy Statement". A "Next" button is located at the bottom of this panel. At the very bottom of the page, there's a "Sign up" button and a "Start Chat" button.



2. Provision the Virtual Machine:

- Navigate to the Azure portal and create a new Virtual Machine



➤ Select Azure virtual machine

The screenshot shows the Microsoft Azure Portal home page. In the top navigation bar, there is a dropdown menu labeled "Virtual machines". A tooltip is displayed over this menu, listing three options: "Azure virtual machine" (Create a virtual machine hosted by Azure), "Azure virtual machine with preset configuration" (Create a virtual machine with presets based on your workloads), and "More VMs and related solutions" (Discover and deploy full workloads and Azure products for your business needs). Below the dropdown, there is a "Last Viewed" section with two items: "Windows virtual machine in Azure" (a day ago) and "Create a Linux virtual machine in Azure" (a day ago). The main content area includes sections for "Azure services", "Resources" (Recent and Favorite), "Navigate" (Subscriptions), "Tools" (Microsoft Learn, Azure Monitor, Microsoft Defender for Cloud, Cost Management), and "Useful links" (Overview, All resources, Dashboard).

- Select the appropriate VM configuration, including the operating system (e.g., Ubuntu), size, and region.

The screenshot shows the "Create a virtual machine" wizard, step 1: Project details. The title bar says "Create a virtual machine - Microsoft Virtual Machine - ARM". The page has a header with "Help me create a low cost VM", "Help me create a VM optimized for high availability", and "Help me choose the right VM size for my workload". The "Project details" section asks to select a subscription and resource group. The "Subscription" dropdown is set to "free trial" and the "Resource group" dropdown shows "(New) Resource group" with "Create new" option. The "Instance details" section includes fields for "Virtual machine name" (mandatory), "Region" (set to "(Asia Pacific) South India"), "Availability options" (set to "No infrastructure redundancy required"), "Security type" (set to "Trusted launch virtual machines"), and "Image" (set to "Ubuntu Server 24.04 LTS - x64 Gen2 (free services eligible)"). At the bottom, there are buttons for "< Previous", "Next : Disks >", "Review + create", and "Give feedback".

- Configure administrative access using SSH keys or a password.

Administrator account

Authentication type: SSH public key Password

Azure now automatically generates an SSH key pair for you and allows you to store it for future use. It is a fast, simple, and secure way to connect to your virtual machine.

Username *: azureuser

SSH public key source: Generate new key pair

SSH Key Type: RSA SSH Format Ed25519 SSH Format
Ed25519 offers better performance and security with a smaller key size, while RSA is still widely used particularly for legacy systems and applications.

Key pair name *: VishnuProjectKeyPair

< Previous | Next : Disks > | Review + create | Give feedback

- Configure network security groups (NSGs) to allow HTTP/HTTPS traffic and secure SSH access.

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network *: (new) VishnuProject-vnet

Subnet *: (new) default (10.0.0.0/24)

Public IP: (new) VishnuProject-ip

NIC network security group: Basic Advanced

Public inbound ports *: Allow selected ports

Select inbound ports *: HTTP (80), HTTPS (443), SSH (22)

< Previous | Next : Management > | Review + create | Give feedback

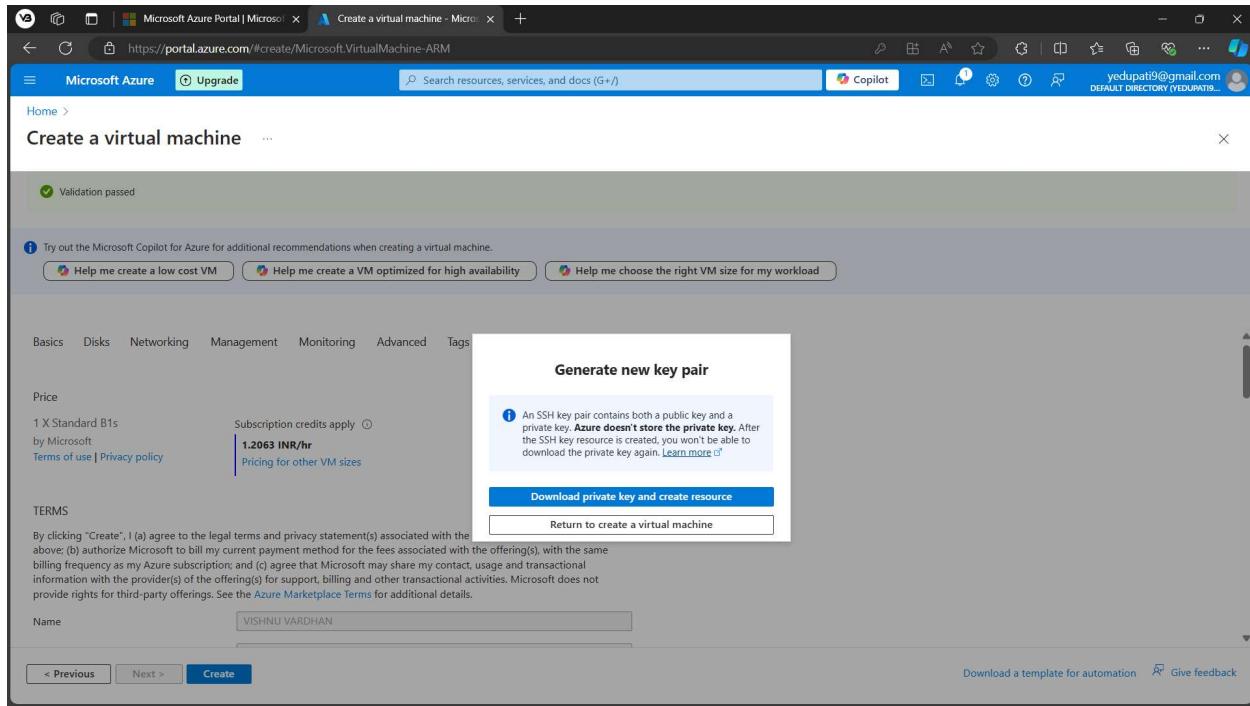
- Set up the Management, Monitoring, Advanced, Tags of a virtual machine.

The screenshot shows the Microsoft Azure Portal interface for creating a virtual machine. The URL is https://portal.azure.com/#create/Microsoft.VirtualMachine-ARM. The 'Advanced' tab is currently selected. The page displays several configuration sections: 'Extensions' (with a note about post-deployment automation), 'VM applications' (with a note about secure download and provisioning scripts), and 'Custom data and cloud init' (with a note about passing configuration data during provisioning). At the bottom, there are navigation buttons ('< Previous', 'Next : Tags >', 'Review + create') and a feedback link.

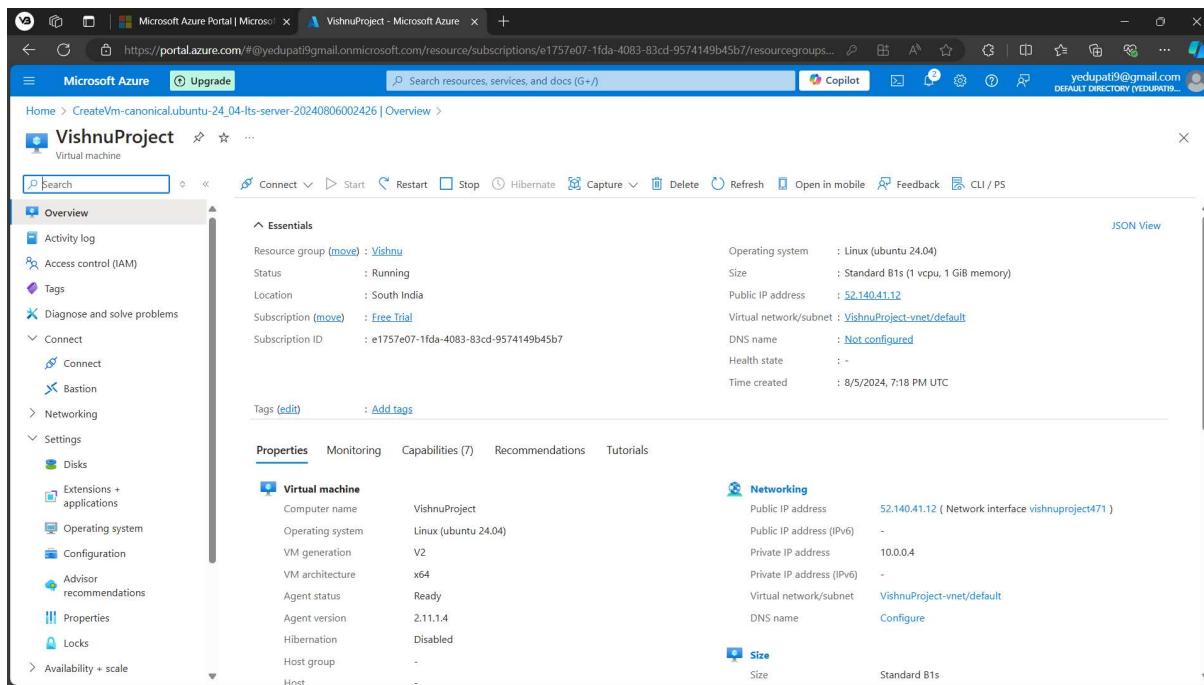
- Tap Review + Create and verify all and tap create.

The screenshot shows the Microsoft Azure Portal interface for creating a virtual machine, with the 'Review + create' tab selected. A green bar at the top indicates 'Validation passed'. Below it, a summary section shows the configuration: '1 X Standard B1s by Microsoft' and a price of '1.2063 INR/hr'. There is also a note about 'Subscription credits apply'. The 'TERMS' section contains the Microsoft Marketplace Terms of Use. At the bottom, there is a 'Name' input field containing 'VISHNU VARDHAN', a 'Create' button, and links for 'Download a template for automation' and 'Give feedback'.

➤ Download the Private Key

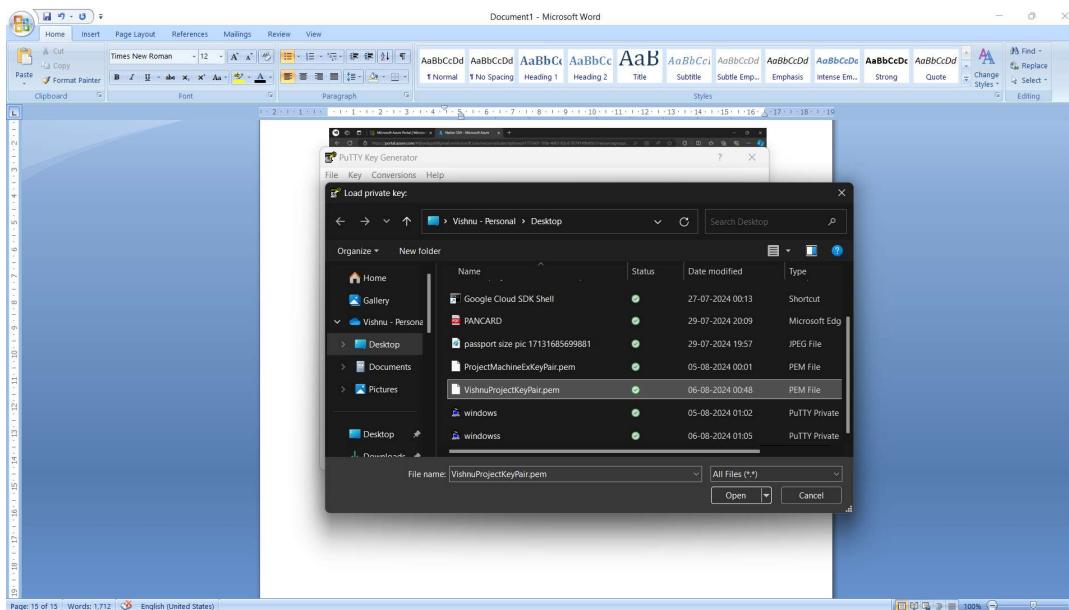


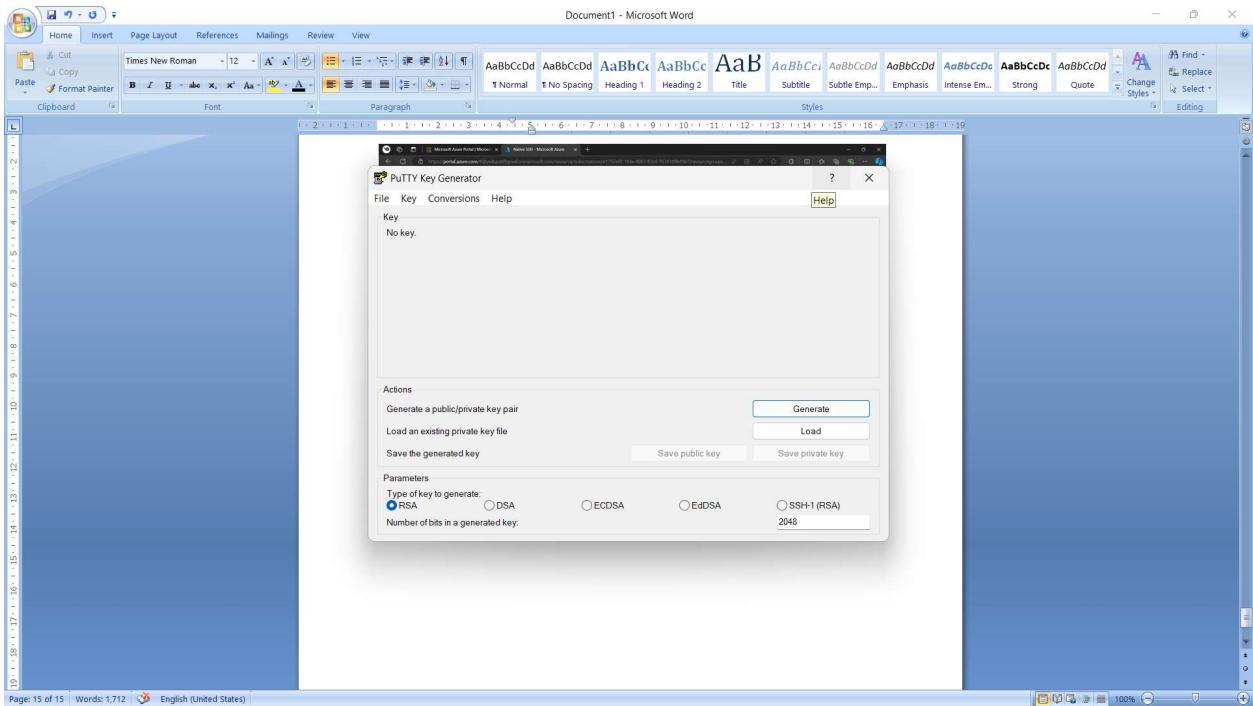
- Wait for few minutes to create resource and virtual machine.
➤ After that click on Go to resource and it will open the resource and virtual machine.



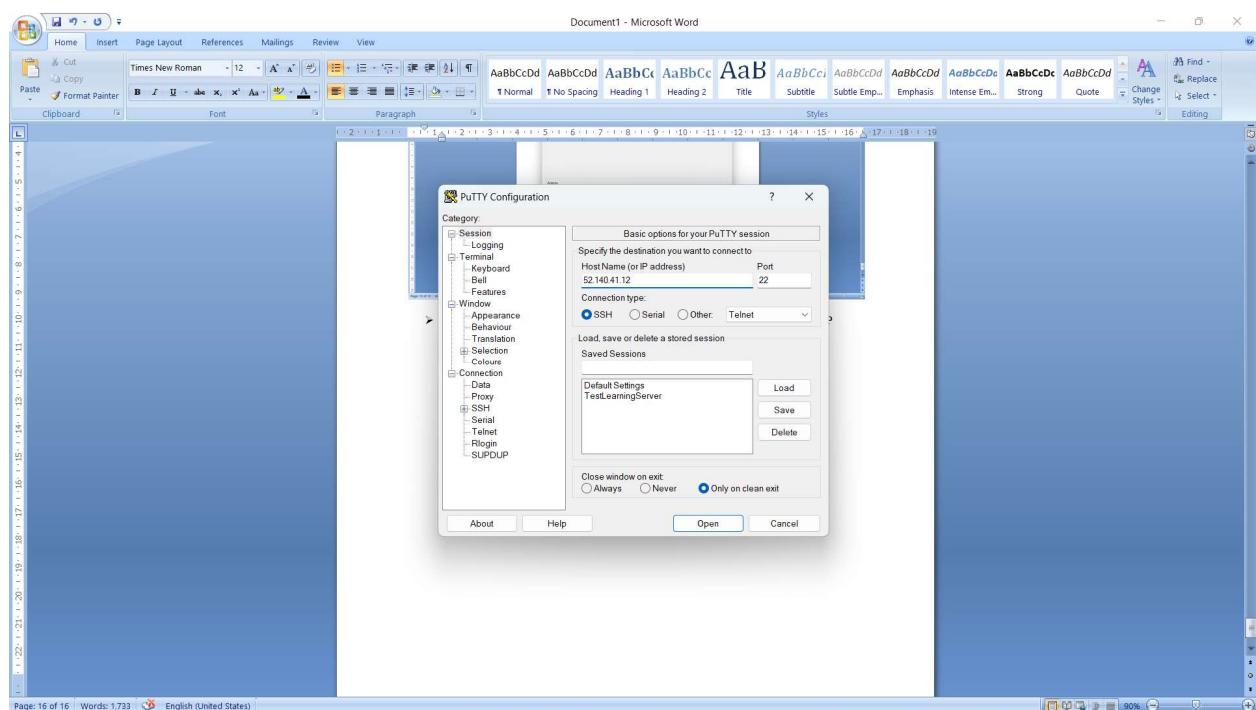
- Click the connect to connect the virtual machine to server select SSH.

- Now install the PuttY from the browser you using and open PuttY key generator and click on the Load.
- Now select the private key which you have downloaded.
- Then successfully it will generate the key in putty and click on “**save private key**” commad.





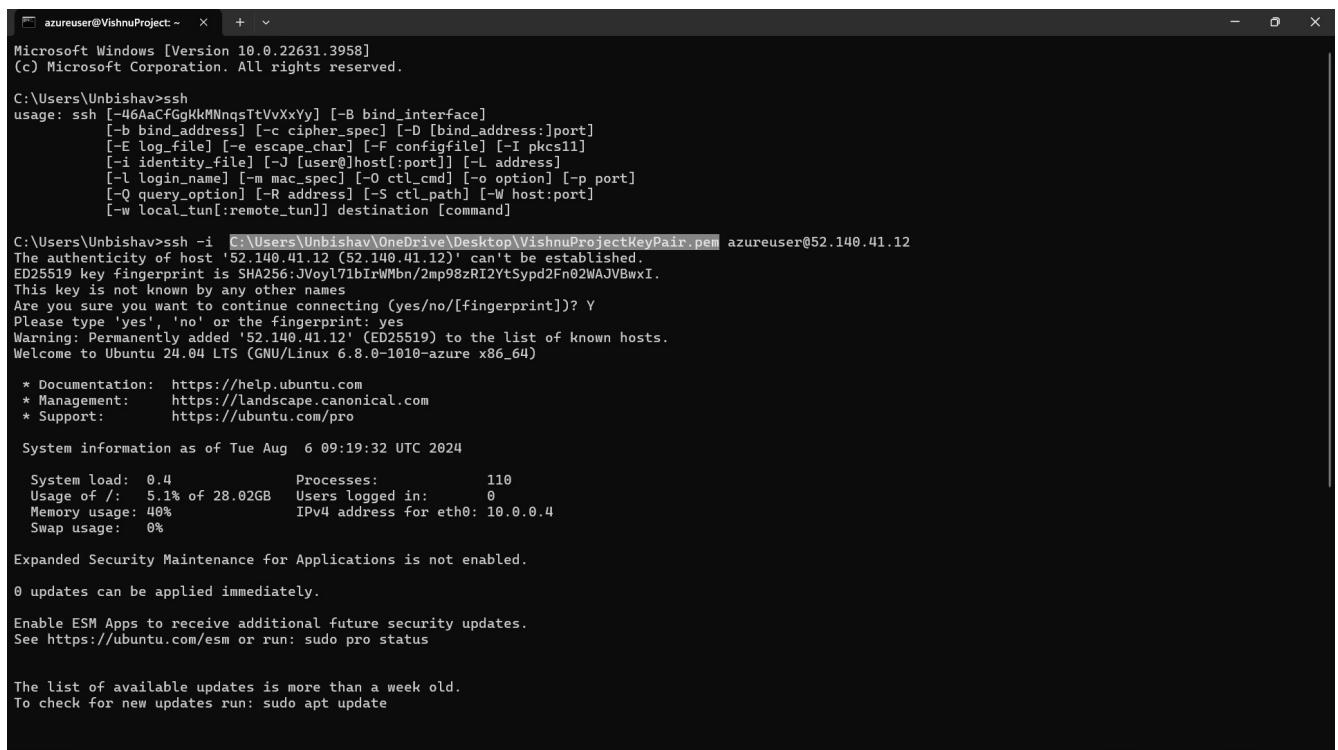
- Use the PuttY to connect the VM(Virtual Machine) by entering the host name as IP Address(Public IP Address from VM).



- Now go to SSH and go to AUTH and select Credentialas browse your private key select open command and it will connect to VM machine. If it doesn't connect then close the PuttY.

Phase 2: Setting Up the Nginx and PHP:

- Now open the command prompt in your windows check whether SSH is there or not type the command “ssh” in cmd. If it is not there just install through command prompt
- Now go to Azure virtual machine where we have selected the SSH in connect service and copy the command which is in the native SSH step-3 to command prompt. The command is “ssh -i <privatekey> azureuser@52.140.41.12”. In the place of private key copy the path of private key which you had downloaded before and click enter then type yes on the command prompt to connect the VM machine and windows as shown in below screenshot:



```
azureuser@VishnuProject: ~ + ^ x
Microsoft Windows [Version 10.0.22631.3958]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Unbishav>ssh
usage: ssh [-46aCfgKkMMnqsTtVvXxYy] [-B bind_interface]
           [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
           [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]
           [-i identity_file] [-J [user@]host[:port]] [-L address]
           [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]
           [-Q query_option] [-R address] [-S ctl_path] [-W host:port]
           [-w local_tun[:remote_tun]] destination [command]

C:\Users\Unbishav>ssh -i "C:\Users\Unbishav\OneDrive\Desktop\VishnuProject\KeyPair.pem" azureuser@52.140.41.12
The authenticity of host '52.140.41.12 (52.140.41.12)' can't be established.
ED25519 key fingerprint is SHA256:Voy7ibirWMbn/2mp98zR12Yt5ypd2n02WAJVbwxI.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? Y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '52.140.41.12' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1010-azure x86_64)

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

System information as of Tue Aug  6 09:19:32 UTC 2024

  System load:  0.4          Processes:           110
  Usage of /:   5.1% of 28.02GB  Users logged in:     0
  Memory usage: 40%          IPv4 address for eth0: 10.0.0.4
  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 list of available updates is more than a week old.
To check for new updates run: sudo apt update
```

- Now you can see the windows and the VM machine are connected.

```

azureuser@VishnuProject:~ + 
System information as of Tue Aug 6 09:19:32 UTC 2024
System load: 0.4 Processes: 110
Usage of /: 5.1% of 28.02GB Users logged in: 0
Memory usage: 40% IPv4 address for eth0: 10.0.0.4
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 list of available updates is more than a week old.
To check for new updates run: sudo apt update

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@VishnuProject:~$ |

```

- Use the “ls -la” command to view displays the contents of the current directory.
- Use the “pwd” to print working directory.

```

azureuser@VishnuProject:~ + 
System information as of Tue Aug 6 09:19:32 UTC 2024
System load: 0.4 Processes: 110
Usage of /: 5.1% of 28.02GB Users logged in: 0
Memory usage: 40% IPv4 address for eth0: 10.0.0.4
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 list of available updates is more than a week old.
To check for new updates run: sudo apt update

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@VishnuProject:~$ ls -la
total 28
drwxr-x--- 4 azureuser azureuser 4096 Aug 6 09:19 .
drwxr-xr-x 3 root      root    4096 Aug 5 19:19 ..
-rw-r--r-- 1 azureuser azureuser 220 Mar 31 08:41 .bash_logout
-rw-r--r-- 1 azureuser azureuser 3771 Mar 31 08:41 .bashrc
drwx----- 2 azureuser azureuser 4096 Aug 6 09:19 .cache
-rw-r--r-- 1 azureuser azureuser 897 Mar 31 08:41 .profile
drwx----- 2 azureuser azureuser 4096 Aug 5 19:19 .ssh
azureuser@VishnuProject:~$ pwd
/home/azureuser
azureuser@VishnuProject:~$ |

```

- Use the following commands to perform update and upgrade the package lists.

“sudo apt-get update -y”
“sudo apt-get upgrade -y”

- Now install the web server Nginx using the following command

“sudo apt-get install nginx -y”

- Start and enable the web server to run using the following commands

“sudo systemctl start nginx”

“sudo systemctl enable nginx”

You can observe through following image:

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
azureuser@VishnuProject:~$ sudo systemctl start nginx  
azureuser@VishnuProject:~$ sudo systemctl enable nginx  
Synchronizing state of nginx.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.  
Executing: /usr/lib/systemd/systemd-sysv-install enable nginx  
azureuser@VishnuProject:~$
```

- Install the PHP and check the version of the PHP by using the following commands:

“sudo apt-get install php”

“php -v”

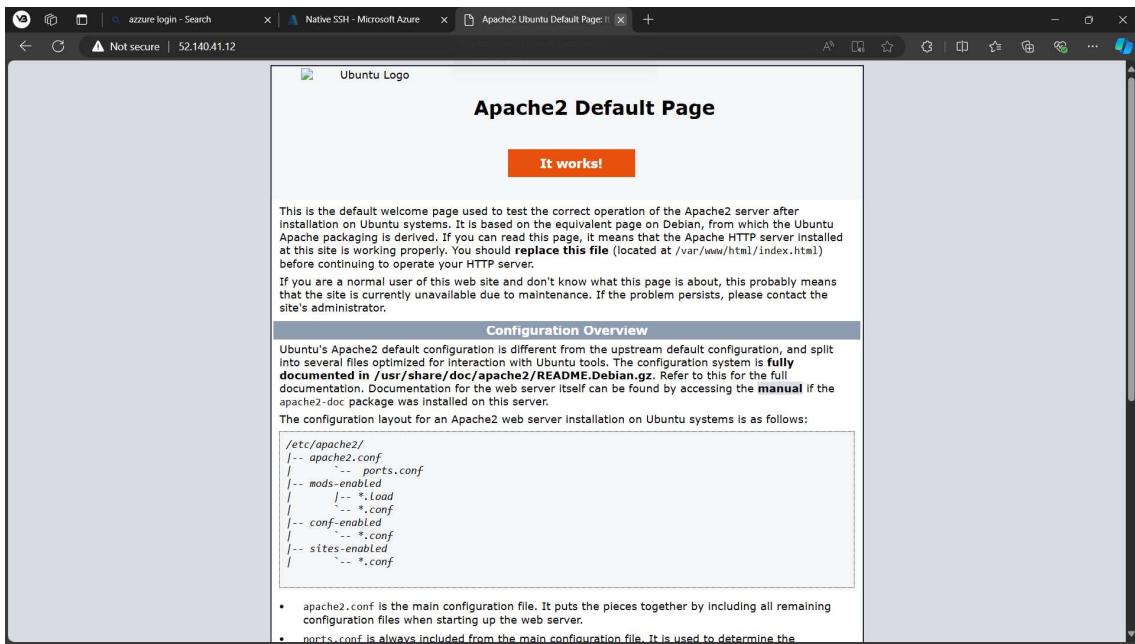
- Remove the apache 2 which we doesn't need by using the following command

“sudo apt-get remove apache2”

- Now check the nginx is running or not by using the following command:

“sudo systemctl status nginx”

- To check the nginx web server is working or not. Search the public IP Address on the browser then we will observe that apache2 ubuntu page will be displayed as shown in the below screenshot:



- Now change the directory to the /etc folder and get list of the folder by using the following commands:

“cd /etc” and “ls”

- Then you can observe nginx in the list. Now add the nginx folder get list of the folder by using the following commands:

“cd /nginx” and “ls”

- Then you can observe sites-enabled in the list. Now add the sites-enabled folder get list of the folder by using the following commands:

“cd / sites-enabled” and “ls”

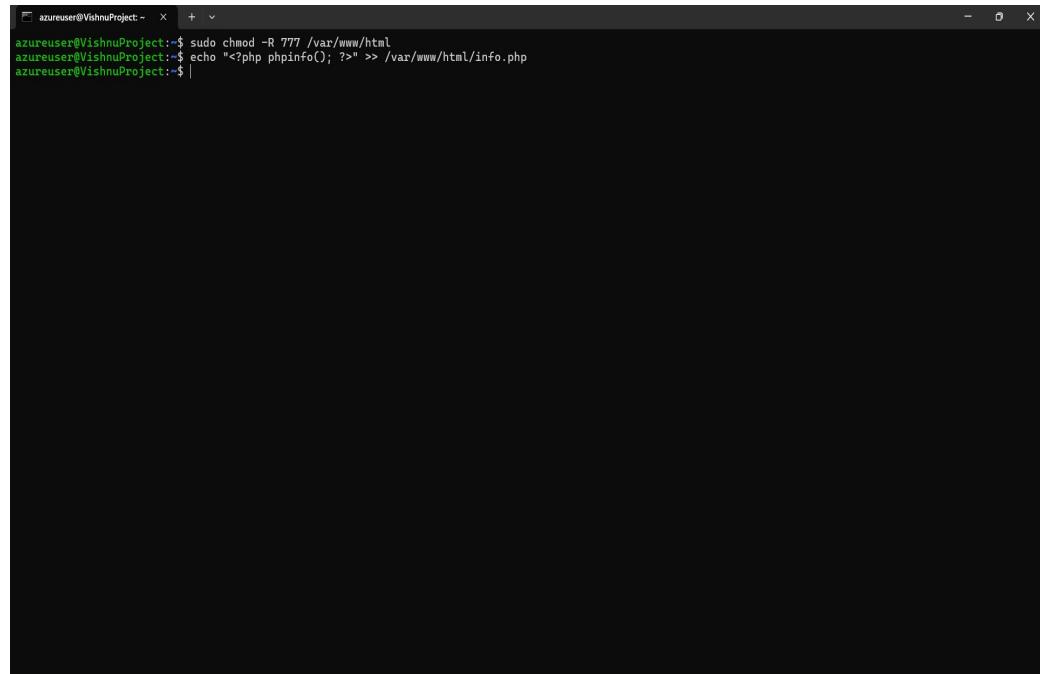
- You can observe a default file and to open it use the command:

“sudo nano default”

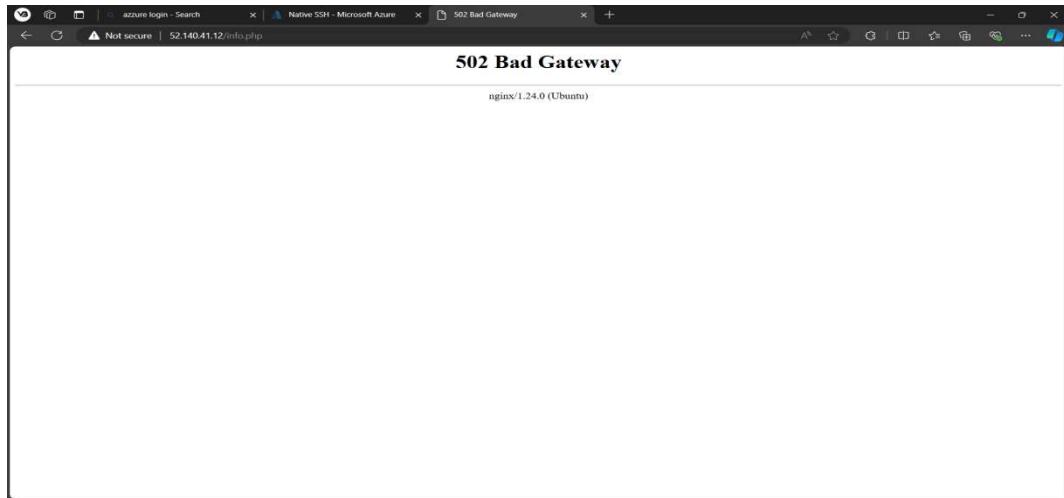
and add the **index.php** in the file before index.html and uncomment the location of the php line and fastcgi-pass unix line. After that save the file and exit from the file.

- Now verify the syntax using the following command **“sudo nginx -t”**
- You can observe that the system is proper. Then restart the nginx by using the following command: **“sudo systemctl restart nginx”**

- Now go to root folder using the command “**cd ~**” and observe the following screenshots to get the info.php in the browser.



```
azureuser@VishnuProject:~$ sudo chmod -R 777 /var/www/html
azureuser@VishnuProject:~$ echo "<?php phpinfo(); ?>" >> /var/www/html/info.php
azureuser@VishnuProject:~$ |
```

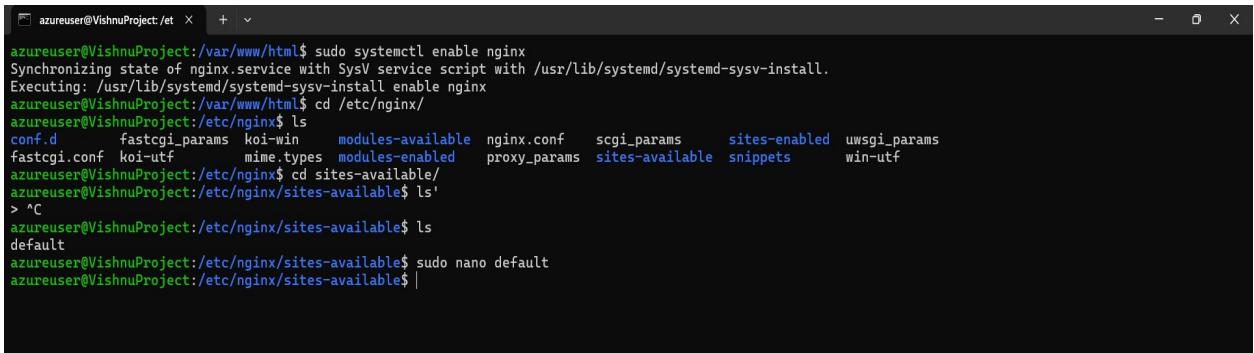


- By observing the above screenshot we are added /info.php to the ip address. By observing that we know some error is there. To find error go to the folder var, www, html by using the command “**cd /var/www/html/**” and use **ls** command to view the files in the folder.
- Now enter into the folder by using the command “**nano info.php**”. If the details in the folder are correct then reverify. To reverify just refresh the page of the browser.
- If the page is not loading then check the status of the nginx by the command “**sudo systemctl status nginx**”
- To check the error in nginx use the command “**tail -f /var/log/nginx/error.log**”

```
azureuser@VishnuProject:/var/www/html$ tail -f /var/log/nginx/error.log
2024/08/06 10:53:53 [alert] 9009#9009: *34 open socket #7 left in connection 3
2024/08/06 10:53:53 [alert] 9009#9009: *35 open socket #11 left in connection 4
2024/08/06 10:53:53 [alert] 9009#9009: aborting
2024/08/06 11:05:57 [crit] 17830#17830: *9 connect() to unix:/run/php/php7.4-fpm.sock failed (2: No such file or directory) while connecting to upstream, cl
ient: 43.225.25.129, server: _, request: "GET /info.php HTTP/1.1", upstream: "fastcgi://unix:/run/php/php7.4-fpm.sock:", host: "52.140.41.12"
2024/08/06 12:00:24 [crit] 17830#17830: *36 connect() to unix:/run/php/php7.4-fpm.sock failed (2: No such file or directory) while connecting to upstream, c
lient: 43.225.25.129, server: _, request: "GET /info.php HTTP/1.1", upstream: "fastcgi://unix:/run/php/php7.4-fpm.sock:", host: "52.140.41.12"
^C
azureuser@VishnuProject:/var/www/html$ |
```

We can notice the error in nginx that is not having the php7.4-fpm installed. So now uninstall the php using the following command “**`sudo apt-get purge php`**”.

- Once update the folder and install the software-properties-common and add the repository php:onderj/php to modify the error by following commands
“`sudo apt -y install software-properties-common`” & “`sudo add-apt-repository ppa:ondrej/php`”.
- Now update and upgrade the system and install php8.1 by command “**`sudo apt -y install php8.1`**” and install php-fpm by command “**`sudo apt -y install php8.1-fpm`**”.
- To start and enable the php8.1-fpm use the command “**`sudo systemctl start php8.1-fpm`**” and “**`sudo systemctl enable php8.1-fpm`**”.
- Now enable the nginx webserver and verify the default file again.



```
azureuser@VishnuProject:/var/www/html$ sudo systemctl enable nginx
Synchronizing state of nginx.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable nginx
azureuser@VishnuProject:/var/www/html$ cd /etc/nginx/
azureuser@VishnuProject:/etc/nginx$ ls
conf.d      fastcgi_params  koi-win    modules-available  nginx.conf    scgi_params   sites-enabled  uwsgi_params
fastcgi.conf  koi-utf       mime.types  modules-enabled proxy_params  sites-available snippets    win-utf
azureuser@VishnuProject:/etc/nginx$ cd sites-available/
azureuser@VishnuProject:/etc/nginx/sites-available$ ls
> ^C
azureuser@VishnuProject:/etc/nginx/sites-available$ ls
default
azureuser@VishnuProject:/etc/nginx/sites-available$ sudo nano default
azureuser@VishnuProject:/etc/nginx/sites-available$ |
```

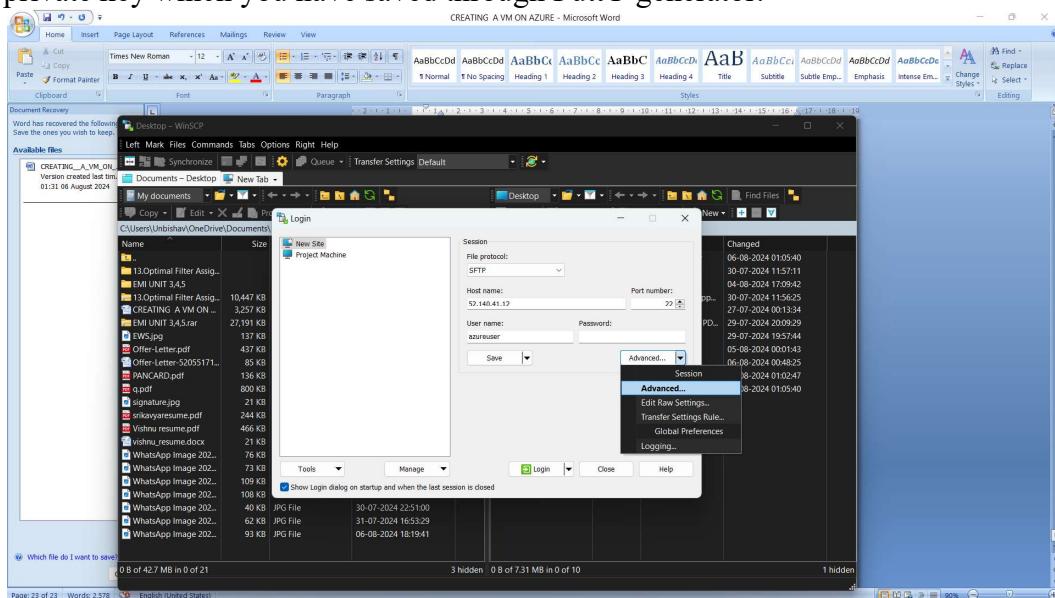
Once you enter into the default file change version of php7.4 to php8.1 and save and exit.

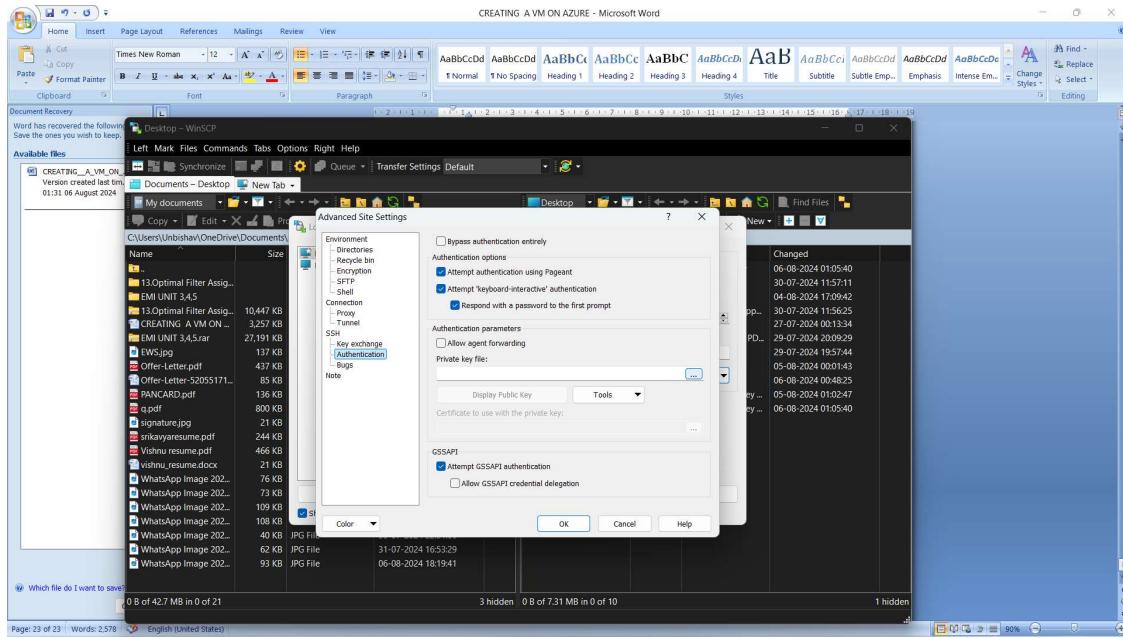
- Now verify the syntax of nginx by command “**`sudo nginx -t`**” and restart the nginx and go to the browser refresh and you it will load following page.

PHP Version 8.1.20	
System	Linux ProjectMachineOne 5.15.0-1040-azure #47~20.04.1-Ubuntu SMP Fri Jun 2 21:38:08 UTC 2023 x86_64
Build Date	Jun 8 2023 15:25:40
Build System	Linux
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/fpm
Loaded Configuration File	/etc/php/8.1/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/fpm/conf.d
Additional .ini files parsed	/etc/php/8.1/fpm/conf.d/10-opcache.ini, /etc/php/8.1/fpm/conf.d/10-pdo.ini, /etc/php/8.1/fpm/conf.d/20-calendars.ini, /etc/php/8.1/fpm/conf.d/20-cgiINI.ini, /etc/php/8.1/fpm/conf.d/20-exif.ini, /etc/php/8.1/fpm/conf.d/20-fil.ini, /etc/php/8.1/fpm/conf.d/20-finfo.ini, /etc/php/8.1/fpm/conf.d/20-ftp.ini, /etc/php/8.1/fpm/conf.d/20-gettext.ini, /etc/php/8.1/fpm/conf.d/20-iconv.ini, /etc/php/8.1/fpm/conf.d/20-phar.ini, /etc/php/8.1/fpm/conf.d/20-posix.ini, /etc/php/8.1/fpm/conf.d/20-readline.ini, /etc/php/8.1/fpm/conf.d/20-shmop.ini, /etc/php/8.1/fpm/conf.d/20-sockets.ini, /etc/php/8.1/fpm/conf.d/20-sysvmsg.ini, /etc/php/8.1/fpm/conf.d/20-sysvsem.ini, /etc/php/8.1/fpm/conf.d/20-tokenizer.ini, /etc/php/8.1/fpm/conf.d/20-xsl.ini, /etc/php/8.1/fpm/conf.d/20-ziptokenizer.ini
PHP API	20210902
PHP Extension	20210902
Zend Extension	420210902
Zend Extension Build	API420210902.NTS
PHP Extension Build	API420210902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
Zend Max Execution Timers	disabled
IPv6 Support	enabled
Intl Support	available, disabled

Phase 3: Website Deployment and Configuration:

- Now install WinSCP from the browser and open it and give the host name as your public ip address and your username and go advanced option select authentication browse the private key which you have saved through PuttY generator.





Then click OK and click save option and login and accept the conditions.

- After login upload the project file and now open the folders var and www in command prompt by using “**cd /var/www/**” and use **ls**.
- Make new folder **project.unbishavprojects.shop** in command prompt and give the permissions by using the command “**sudo chmod 777 /var/www/**”.
- Make new folder by “**mkdir 1stopproject.unbishavprojects.shop**”.
- Now upload the project zip file into www folder and unzip through in cmd by using following commands

“cd 1stopproject.unbishavprojects.shop/”

“ls”

“sudo apt-get install unzip”

“unzip 1stopproject.zip”

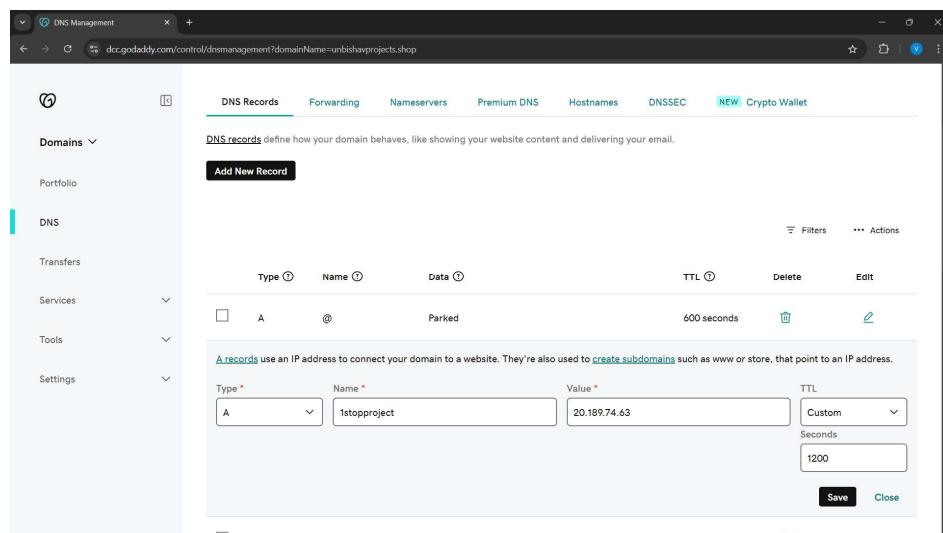
- After unzipping the files in cmd check in WinSCP the zip file is unzipped. Now move the HTML and CSS files to the **project.unbishavprojects.shop** folder from cmd.

```
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ mv /var/www/1stopproject.unbishavprojects.shop/* /var/www/1stopproject.unbishavprojects.shop/
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ ls
1stopproject index.html styles.css
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ rm -rf 1stopproject
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ ls
index.html styles.css
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ sudo |
```

- Now copy the 1stopproject.unbishavprojects.shop file into nginx folder from cmd and use **ls** command to see the list in the site-available folder and use “**sudo nano 1stopproject.unbishavprojects.shop**” command to enter and modify the root path as “root /var/www/1stopproject.unbishavprojects.shop;” save and exit from it.

```
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ mv /var/www/1stopproject.unbishavprojects.shop/* /var/www/1stopproject.unbishavprojects.shop/
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ ls
1stopproject index.html styles.css
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ rm -rf 1stopproject
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ ls
index.html styles.css
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ cp /etc/nginx/sites-available/default /etc/nginx/sites-available
cp: '/etc/nginx/sites-available/default' and '/etc/nginx/sites-available/default' are the same file
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ cd /etc/nginx/sites-available/
projectuser@My1stopProject:/etc/nginx/sites-available$ ls
default
projectuser@My1stopProject:/etc/nginx/sites-available$ cd /var/www/1stopproject.unbishavprojects.shop/
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ sudo cp /etc/nginx/sites-available/default /etc/nginx/sites-available/1stopproject.unbishavprojects.shop
projectuser@My1stopProject:~/var/www/1stopproject.unbishavprojects.shop$ cd /etc/nginx/sites-available/
projectuser@My1stopProject:/etc/nginx/sites-available$ ls
1stopproject.unbishavprojects.shop default
projectuser@My1stopProject:/etc/nginx/sites-available$ sudo nano 1stopproject.unbishavprojects.shop
projectuser@My1stopProject:/etc/nginx/sites-available$ sudo nano 1stopproject.unbishavprojects.shop
```

- Now go to GoDaddy domain website and purchase a domain for our website and go to DNS management and click on add new record and select types as A and enter name as 1stopproject and give the value as your public ip address and give the TTL and save it as shown in figure

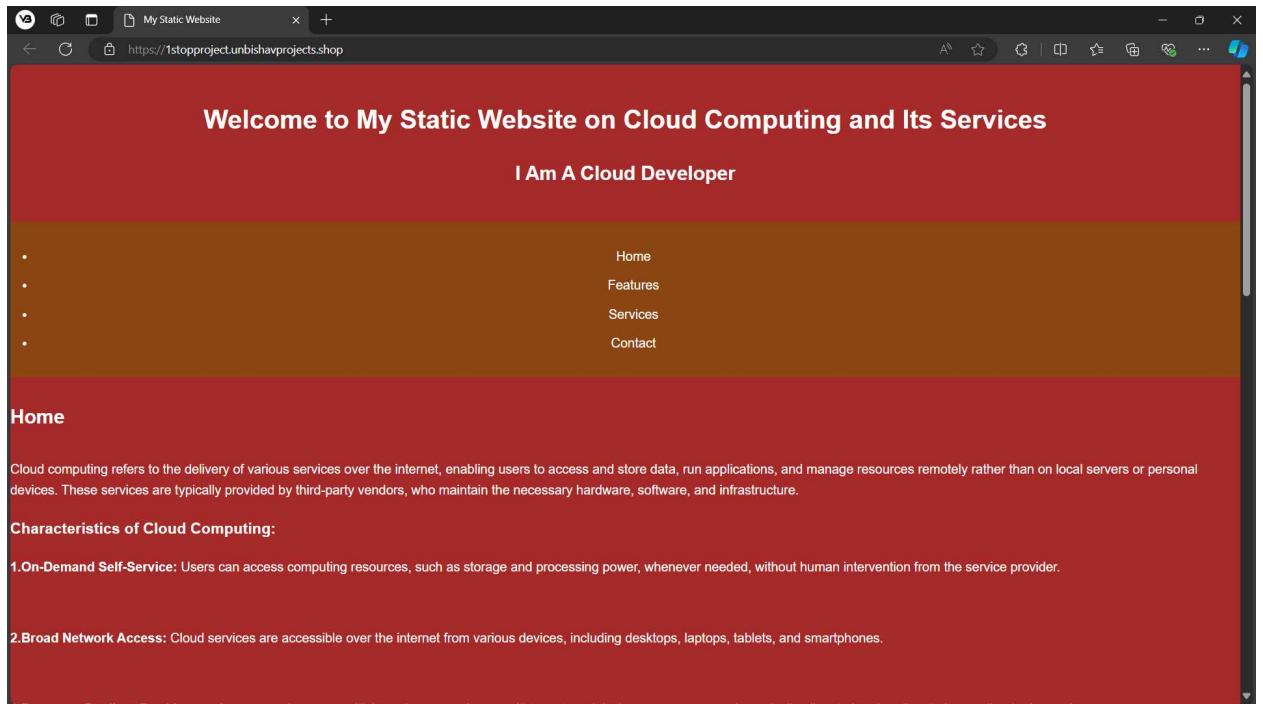


It is useful to connect your web server to the browser.

- Now use the following commands and reload the nginx and check the syntax as shown in below images to open your static website.

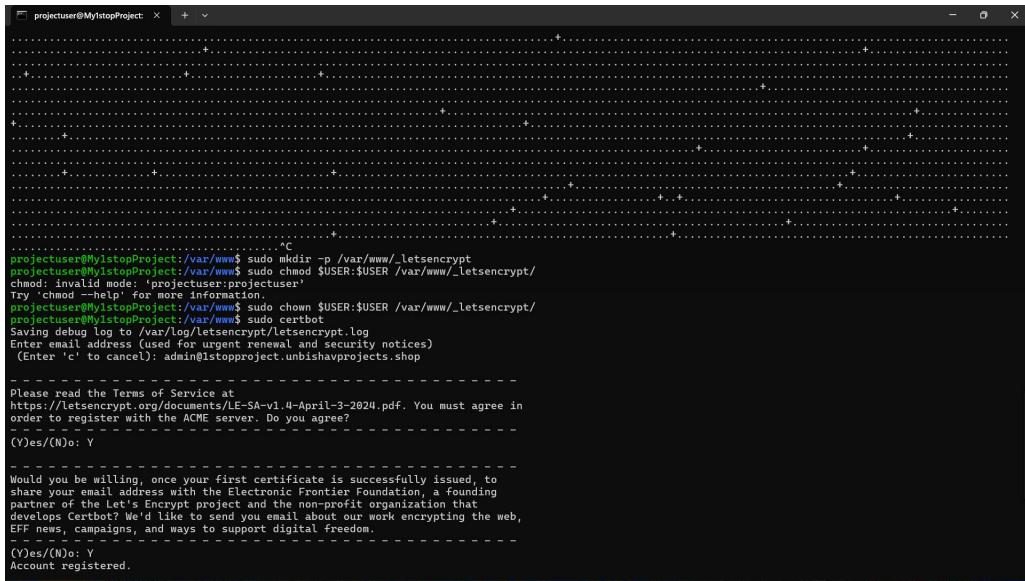
```
projectuser@My1stopProject:/etc/nginx/sites-available$ sudo chown -R $USER:$USER /var/www/1stopproject.unbishavprojects.shop/
projectuser@My1stopProject:/etc/nginx/sites-available$ sudo chmod -R g+rw /var/www/1stopproject.unbishavprojects.shop/
projectuser@My1stopProject:/etc/nginx/sites-available$ sudo ln -s /etc/nginx/sites-available/1stopproject.unbishavprojects.shop /etc/nginx/sites-enabled/
projectuser@My1stopProject:/etc/nginx/sites-available$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
projectuser@My1stopProject:/etc/nginx/sites-available$ sudo systemctl reload nginx
```

- Now go to the browser search as `1stop.unbishavprojects.shop` to load the static website as shown in figure:



- Let's encrypt the certificates to encrypt install the `python3-certbot-nginx` by entering the following command “`sudo apt-get install python3-certbot-nginx`”.
- After that generate the DH parameter by entering the command “`sudo openssl dhparam -out /etc/nginx/dhparam.pem 2048`” and to use the following commands to get the

certificates for the website as shown below:

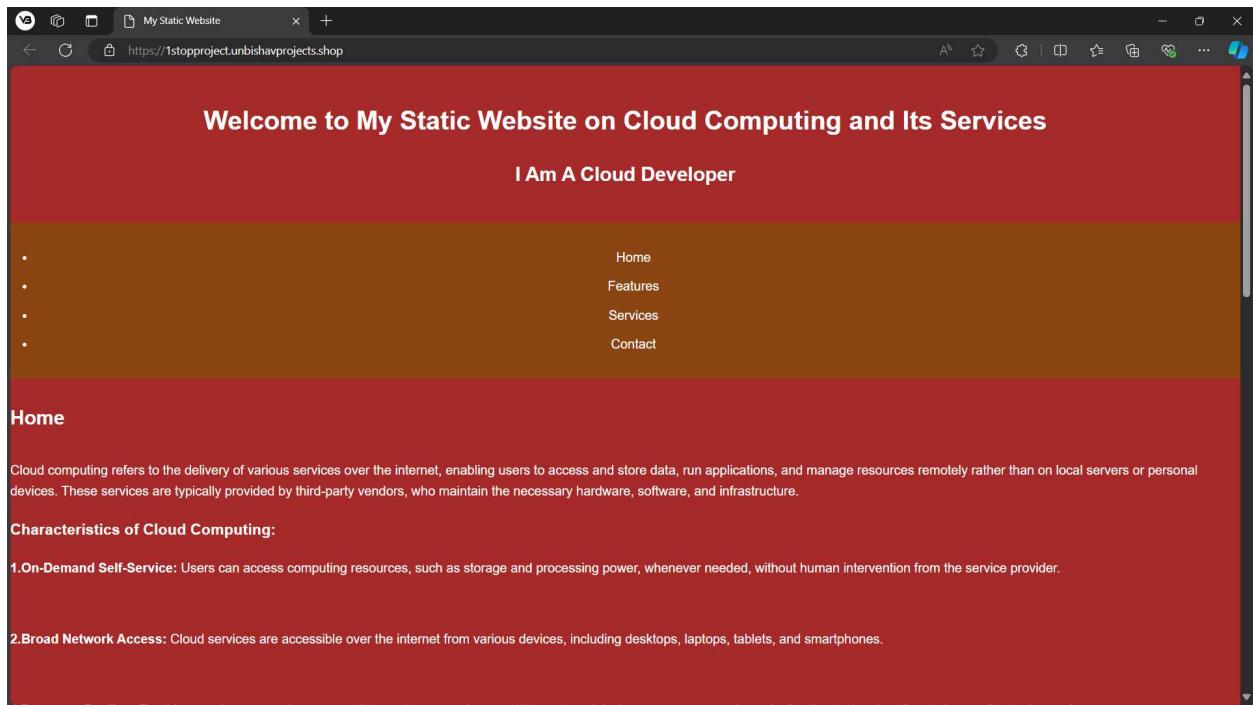


```
projectuser@My1stopProject:~/var/www$ sudo mkdir -p /var/www/_letsencrypt
projectuser@My1stopProject:~/var/www$ sudo chmod $USER:$USER /var/www/_letsencrypt/
chmod: invalid mode: 'projectuser:projectuser'
Try 'chmod --help' for more information.
projectuser@My1stopProject:~/var/www$ sudo chown $USER:$USER /var/www/_letsencrypt/
projectuser@My1stopProject:~/var/www$ sudo certbot
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Enter email address (used for urgent renewal and security notices)
(Enter 'c' to cancel): admin@1stopproject.unbishavprojects.shop

Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.4-April-3-2024.pdf. You must agree in
order to register with the ACME server. Do you agree?
-- -- -- -- --
(Y)es/(N)o: Y

Would you be willing, once your first certificate is successfully issued, to
share your email address with the Electronic Frontier Foundation, a founding
partner of the Let's Encrypt project and the non-profit organization that
develops Certbot? We'd like to send you email about our work encrypting the web,
EFF news, campaigns, and ways to support digital freedom.
-- -- -- -- --
(Y)es/(N)o: Y
Account registered.
```

- After this go to the browser and refresh the static website will be loaded.



CODE

HTML CODE:

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>My Static Website</title>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <header>

        <h1>Welcome to My Static Website on Cloud Computing and Its Services</h1>

        <h2>I Am A Cloud Developer</h2>

    </header>

    <nav>

        <ul>

            <li><a href="#home">Home</a></li>

            <li><a href="#features">Features</a></li>

            <li><a href="#services">Services</a></li>

            <li><a href="#contact">Contact</a></li>

        </ul>

    </nav>
```

```
<main>

<section id="home">

    <h2>Home</h2>

    <p>Cloud computing refers to the delivery of various services over the internet, enabling users to access and store data, run applications, and manage resources remotely rather than on local servers or personal devices. These services are typically provided by third-party vendors, who maintain the necessary hardware, software, and infrastructure.</p>

    <h3>Characteristics of Cloud Computing:</h3>

    <p><b>1.On-Demand Self-Service:</b> Users can access computing resources, such as storage and processing power, whenever needed, without human intervention from the service provider.</p><br>

    <p><b>2.Broad Network Access:</b> Cloud services are accessible over the internet from various devices, including desktops, laptops, tablets, and smartphones.</p><br>

    <p><b>3.Resource Pooling:</b> Providers pool resources to serve multiple customers, using a multi-tenant model where resources are dynamically allocated and reallocated according to demand.</p><br>

    <p><b>4.Rapid Elasticity:</b> Resources can be quickly scaled up or down to accommodate changes in workload or demand.</p><br>

    <p><b>5.Measured Service:</b> Cloud systems automatically control and optimize resource usage by leveraging a metering capability, so customers are billed based on their actual usage.</p>

    <p>Types of Cloud Services:</p>

    <ul>

        <li><a href="https://aws.amazon.com/">Amazon Web Services (AWS)</a></li>

        <li><a href="https://azure.microsoft.com/">Microsoft Azure</a></li>

        <li><a href="https://console.cloud.google.com/">Google Cloud Services (GCP)</a></li>

    </ul>

</section>
```

```
<section id="features">

<h2>Features</h2>

<ul>

    <li>Scalable Infrastructure: Dynamically adjust resources based on demand.</li>
    <li>High Availability: Ensure uptime and reliability with redundant systems.</li>
    <li>Cost Efficiency: Pay-as-you-go model to optimize expenses.</li>
    <li>Robust Security: Advanced encryption and access control to protect your data.</li>
    <li>Easy Integration: Seamlessly connect with existing tools and services.</li>
</ul>

</section>

<section id="services">

<h2>Cloud Services</h2>

<ul>

    <li><a><u>Amazon Web Services (AWS)</u></a><p>Amazon Web Services (AWS) is a comprehensive and widely adopted cloud platform provided by Amazon. Launched in 2006, AWS offers a broad set of cloud-based services, ranging from computing power, storage, and databases to machine learning, artificial intelligence, analytics, and more. It has become a dominant player in the cloud computing industry, powering millions of businesses worldwide, from startups to large enterprises.</p></li>
    <li><a><u>Microsoft Azure</u></a><p>Microsoft Azure is a cloud computing platform and service provided by Microsoft. It offers a wide range of cloud services, including those for computing, analytics, storage, and networking. Users can choose and configure these services to meet their specific needs, ranging from running existing applications in the cloud to developing and deploying new ones. Since its launch in 2010, Azure has become one of the leading cloud platforms globally, alongside AWS and Google Cloud.</p></li>
    <li><a><u>Google Cloud Services (GCP)</u></a><p>Google Cloud Platform (GCP) is a suite of cloud computing services offered by Google. It provides a range of infrastructure, platform, and software services that allow businesses, developers, and organizations to build, deploy, and scale applications, websites, and services using Google's powerful infrastructure. Launched in 2008, GCP has grown to become one of the top cloud
</ul>
</section>
```

platforms alongside AWS and Microsoft Azure, known for its strength in data analytics, machine learning, and its global network.</p>

</section>

<section id="contact">

<h2>Contact</h2>

<p>Phone: +918919739983</p>

<p>Email: yedupati8@gmail.com</p>

<p>Linkidin: Vishnu Vardhan Yedupati</p>

</section>

</main>

<footer>

<p>© 2024 CloudCompute.</p>

</footer>

</body>

</html>

CSS CODE:

/* General Reset and Body Styling */

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

background-color: Brown; /* Red background */

```
color: white; /* White text color for contrast */  
line-height: 1.6;  
}  
  
/* Header Styling */  
  
header {  
background-color: brown; /* Black header for contrast */  
color: white;  
padding: 20px 0;  
text-align: center;  
}  
  
/* Navigation Menu Styling */  
  
nav {  
background: #8B4513; /* SaddleBrown for navigation background */  
color: white;  
padding: 10px;  
text-align: center;  
}  
  
nav ul li a {  
color: white;  
text-decoration: none;  
padding: 5px 15px;
```

```
}
```

```
nav ul li a:hover {
```

```
background-color: brown; /* Hover effect with brown color */
```

```
border-radius: 4px;
```

```
}
```

```
/* Main Content Styling */
```

```
main h2 {
```

```
border-bottom: 2px solid Brown; /* Brown border for section headings */
```

```
padding-bottom: 10px;
```

```
margin-top: 30px;
```

```
}
```

```
main p {
```

```
margin-bottom: 20px;
```

```
}
```

```
/* List Styling */
```

```
ul {
```

```
margin: 20px 0;
```

```
padding-left: 20px;
```

```
}
```

```
li {
```

```
margin-bottom: 10px;
```

```
}

section#services li u {

    font-weight: bold;

}

/* Link Styling */

a {

    color: white; /* White color for links */

    text-decoration: none;

}

a:hover {

    text-decoration: underline;

}

/* Contact Section Styling */

section#contact a {

    color: White; /* Consistent White for links */

    text-decoration: none;

}

section#contact a:hover {

    text-decoration: underline;

}

/* Footer Styling */

footer {
```

```
background-color: Black; /* Black footer for consistency */  
color: white;  
text-align: center;  
padding: 10px 0;  
margin-top: 40px;  
}  
  
/* Responsive Design */  
  
@media (max-width: 600px) {  
    header h1 {  
        font-size: 2em;  
    }  
    header h2 {  
        font-size: 1.2em;  
    }  
    nav ul li {  
        display: block;  
        margin: 5px 0;  
    }  
    nav ul li a {  
        display: block;  
        padding: 10px;  
    }  
}
```

OUTPUT:

The screenshot shows a web browser window with the title bar "My Static Website". The URL in the address bar is "https://1stopproject.unblishavprojects.shop". The main content area has a red header with the text "Welcome to My Static Website on Cloud Computing and Its Services" and "I Am A Cloud Developer". Below the header is a navigation menu with four items: Home, Features, Services, and Contact. The "Home" item is currently selected. The main content section is titled "Home" and contains the following text:

Cloud computing refers to the delivery of various services over the internet, enabling users to access and store data, run applications, and manage resources remotely rather than on local servers or personal devices. These services are typically provided by third-party vendors, who maintain the necessary hardware, software, and infrastructure.

Characteristics of Cloud Computing:

1. **On-Demand Self-Service:** Users can access computing resources, such as storage and processing power, whenever needed, without human intervention from the service provider.
2. **Broad Network Access:** Cloud services are accessible over the internet from various devices, including desktops, laptops, tablets, and smartphones.

The screenshot shows the same web browser window with the title bar "My Static Website". The URL is "https://1stopproject.unblishavprojects.shop". The main content area has a red header with the text "Cloud Services". Below the header is a list of three cloud service providers:

- **Amazon Web Services (AWS):** Amazon Web Services (AWS) is a comprehensive and widely adopted cloud platform provided by Amazon. Launched in 2006, AWS offers a broad set of cloud-based services, ranging from computing power, storage, and databases to machine learning, artificial intelligence, analytics, and more. It has become a dominant player in the cloud computing industry, powering millions of businesses worldwide, from startups to large enterprises.
- **Microsoft Azure:** Microsoft Azure is a cloud computing platform and service provided by Microsoft. It offers a wide range of cloud services, including those for computing, analytics, storage, and networking. Users can choose and configure these services to meet their specific needs, ranging from running existing applications in the cloud to developing and deploying new ones. Since its launch in 2010, Azure has become one of the leading cloud platforms globally, alongside AWS and Google Cloud.
- **Google Cloud Services (GCP):** Google Cloud Platform (GCP) is a suite of cloud computing services offered by Google. It provides a range of infrastructure, platform, and software services that allow businesses, developers, and organizations to build, deploy, and scale applications, websites, and services using Google's powerful infrastructure. Launched in 2008, GCP has grown to become one of the top cloud platforms alongside AWS and Microsoft Azure, known for its strength in data analytics, machine learning, and its global network.

Contact

Phone: +918919739983
Email: yedupati8@gmail.com
LinkedIn: Vishnu Vardhan Yedupati

CONCLUSION

In this project, we successfully demonstrated the process of deploying a static website using Azure's cloud infrastructure. By creating a virtual machine (VM) and hosting the website, we explored the essential steps involved in setting up a server environment and managing web hosting.

This project provided hands-on experience with cloud infrastructure, web development, and server management. By leveraging Azure's capabilities, we demonstrated how to efficiently deploy and host a website. Additionally, configuring a domain and SSL added value by enhancing the website's professionalism and security.

Future Considerations:

- **Scaling:** Consider exploring Azure's options for scaling the website as traffic increases, such as load balancing and auto-scaling.
- **Continuous Integration/Deployment (CI/CD):** Implement CI/CD pipelines for automating deployments and updates to the website.
- **Security Enhancements:** Continue to monitor and enhance security measures to protect against potential vulnerabilities.

By completing this project, you have gained valuable skills in cloud computing and web development, preparing you for more advanced projects in the future. The knowledge acquired here forms a solid foundation for further exploration into dynamic web applications and cloud services.