

# AMAZON WEB SERVICES PROJECT- II

**AIM:** Create two EC2 web servers one with the apache http server installed on it and one with the nginx server installed on it and do all the installations using the user data section of ec2.

## **PROCEDURE:**

We'll proceed step by step with proper understanding, documentation and demonstration wherever possible

**1. Create EC2 Instances:** Go to the AWS Management Console and navigate to the EC2 dashboard. Click on "Launch Instance."

### Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

#### Name and tags [Info](#)

Name

arpits-apache-instance

[Add additional tags](#)

### Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

#### Name and tags [Info](#)

Name

arpits-nginx-instance

[Add additional tags](#)

- Instance “arpits-apache-instance” is for Apache server and instance “arpits-nginx-instance” is for Nginx server.

**2. Choose an Amazon Machine Image (AMI):** Select an Amazon Linux 2 or any other Linux-based AMI of your choice. Here, ubuntu server. We can choose any, only the later command will be different.

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q

Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Li

SUSE

Q

[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

Free tier eligible

ami-0f5ee92e2d63afc18 (64-bit (x86)) / ami-077053fb4029de92f (64-bit (Arm))

Virtualization: hvm    ENA enabled: true    Root device type: ebs

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2023-05-16

Architecture

AMI ID

Verified provider

64-bit (x86)

ami-0f5ee92e2d63afc18

Verified provider

**3. Choose an Instance Type:** Choose an instance type based on your requirements. Here, we have selected by-default instance type i.e., t2.micro.

▼ Instance type [Info](#)

Instance type

t2.micro

Free tier eligible

Family: t2    1 vCPU    1 GiB Memory    Current generation: true

On-Demand Linux base pricing: 0.0124 USD per Hour

On-Demand Windows base pricing: 0.017 USD per Hour

On-Demand RHEL base pricing: 0.0724 USD per Hour

On-Demand SUSE base pricing: 0.0124 USD per Hour

All generations

[Compare instance types](#)

[Additional costs apply for AMIs with pre-installed software](#)

#### 4.Configure Instance:

- Configure the network settings, subnet, and security group to allow HTTP traffic (port 80) and SSH access (port 22).
- In the "Advanced Details" section, add the following user data script for the Apache server:

##### Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

We'll create a new security group called '**launch-wizard-1**' with the following rules:

☒ Allow SSH traffic from  
Helps you connect to your instance

Anywhere  
0.0.0.0/0

☐ Allow HTTPS traffic from the internet  
To set up an endpoint, for example when creating a web server

☒ Allow HTTP traffic from the internet  
To set up an endpoint, for example when creating a web server

In the "Advanced Details" section, add the following user data script for the Apache server:

##### User data - optional [Info](#)

Upload a file with your user data or enter it in the field.

 Choose file

```
#!/bin/bash
apt-get update -y
apt-get install apache2 -y
echo "Welcome to Arpit's Apache Server!!!" > /var/www/html/index.html
systemctl start apache2
```

☐ User data has already been base64 encoded

- `apt-get update -y`: it will do all the updates required.
- `apt-get install apache2 -y`: it will install all the packages needed.
- `echo` means print. it is used to print any type of message in the web server as required.
- `systemctl start apache2`: start the services of apache2.

**5. Add Storage, Tags, and Configure Security Group:** Configure additional settings such as storage, tags, and security groups as needed. We have selected most of them as default.

**6. Review and Launch:** Review your configurations, and when you're ready, click "Launch."

**7. Key Pair:** Choose an existing key pair or create a new one for SSH access to your instances.

▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

my\_key\_pair

▼

↻

Create new key pair

We have created a key pair named `my_key_pair`. We'll use this for both the instances.

**8. Launch Instances:** Click "Launch Instances" to create your EC2 instance.

- Now as we can see in the below picture both our instances are now in running state.

**9. Access the Instances:** Wait for the instances to be launched and their status to become "running." We can now access your EC2 instances using SSH.

Instances (6) Info								
Find instance by attribute or tag (case-sensitive)								
☐	Name ▼	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability Zone ▼	Public IPv4 DNS ▼
☐	arpits-02-insta...	i-06402903566414d59	⊖ Terminated @ @	t2.micro	–	No alarms +	ap-south-1b	–
☐	arpits-02-insta...	i-00554e55a958069d	⊖ Terminated @ @	t2.micro	–	No alarms +	ap-south-1b	–
☐	arpits-instance	i-015038878fe0276de	⊖ Terminated @ @	t2.micro	–	No alarms +	ap-south-1b	–
☐	arpits-instance	i-02f8b0190e3c2df8c	⊖ Terminated @ @	t2.micro	–	No alarms +	ap-south-1b	–
☐	arpits-apache...	i-016fe73e77c8f3c10	⊕ Running @ @	t2.micro	⊕ 2/2 checks passed	No alarms +	ap-south-1b	ec2-52-66-102-50.ap-s...
☐	arpits-nginx-in...	i-0e6d51d3741a3d586	⊕ Running @ @	t2.micro	⊕ Initializing	No alarms +	ap-south-1b	ec2-3-110-219-129.ap-...

For **Nginx installation** for `arpits-nginx-instance` do the following:

- Go to your instance.
- Right click on it. A dropdown list appears as shown in the below picture.
- Click on connect.
- In **connection type** select **Connect using EC2 Instance Connect** and connect.

☑ arpits-nginx-in... i-0e6d51d37

Launch instances

Launch instance from template

Migrate a server

Connect

Stop instance

Start instance

Reboot instance

Hibernate instance

Terminate instance

Instance settings

Networking

Security

Image and templates

Monitor and troubleshoot

Instance: i-0e6d51d3741a3d586

Details

Security

Networking

▼ Instance summary Info

Instance ID

i-0e6d51d3741a3d586 (arpits-nginx-i)

IPv6 address

–

t2.micro

⊕ 2/2 checks passed

No alarms +

ap-south-1b

ec2-3-110-219-129.ap-...

Monitoring

Tags

v4 address

0.219.129 | open address

state

ing

Private IPv4 addresses

172.31.10.60



Public IPv4 DNS

ec2-3-110-219-129.ap-south-1.compute.amazonaws.com | open address

[EC2](#) > [Instances](#) > [i-0e6d51d3741a3d586](#) > [Connect to instance](#)

## Connect to instance [Info](#)

Connect to your instance i-0e6d51d3741a3d586 (arpits-nginx-instance) using any of these options

EC2 Instance Connect	Session Manager	SSH client	EC2 serial console
<p>Instance ID  i-0e6d51d3741a3d586 (arpits-nginx-instance)</p> <p>Connection Type</p> <div><input checked="" type="radio"/> <b>Connect using EC2 Instance Connect</b> Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.</div> <div><input type="radio"/> <b>Connect using EC2 Instance Connect Endpoint</b> Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.</div> <p>Public IP address  3.110.219.129</p> <p>User name Enter the user name defined in the AMI used to launch the instance. If you didn't define a custom user name, use the default user name, ubuntu.</p> <div>ubuntu</div> <div><p><b>Note:</b> In most cases, the default user name, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.</p></div>			

[Cancel](#) [Connect](#)

After clicking on **Connect**, a ubuntu powershell will appear. Here, we have to write the user data commands.

Now type the following commands:

- **sudo apt-get update** : It will do all the updates required for the server
- **sudo apt-get install nginx** : It will install all the necessary packages needed.
- Then it will ask for extra disk space. **Type Y.**
- **sudo systemctl status nginx** : It'll show the current status of the server.

In the status section, it'll show **active (running)**. That means it is ready to serve.



```
sudo apt-get update:
```

```
ubuntu@ip-172-31-10-60:~$ sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [966 kB]
Get:12 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [222 kB]
Get:13 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [15.6 kB]
Get:14 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [830 kB]
Get:15 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [133 kB]
Get:16 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [536 B]
Get:17 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [979 kB]
Get:18 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [213 kB]
Get:19 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [21.8 kB]
Get:20 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [41.6 kB]
Get:21 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [9768 B]
Get:22 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [476 B]
Get:23 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [41.7 kB]
Get:24 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [10.5 kB]
Get:25 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:26 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:27 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [24.3 kB]
Get:28 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.4 kB]
Get:29 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [640 B]
Get:30 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
```

```
sudo apt-get install nginx:
```

```
ubuntu@ip-172-31-10-60:~$ sudo apt-get install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done

The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip2 libnginx-mod-http-image-filter
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2 libtiff5 libwebp7 libxpm4 nginx-common nginx-core
Suggested packages:
  libgd-tools fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip2 libnginx-mod-http-image-filter
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2 libtiff5 libwebp7 libxpm4 nginx nginx-common nginx-core
0 upgraded, 20 newly installed, 0 to remove and 113 not upgraded.
Need to get 2691 kB of archives.
After this operation, 8335 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 fonts-dejavu-core all 2.37-2build1 [1041 kB]
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 fontconfig-config all 2.13-1.4.2ubuntu5 [29.1 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libdeflate0 amd64 1.10-2 [70.9 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libfontconfig1 amd64 2.13-1.4.2ubuntu5 [131 kB]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libjpeg-turbo8 amd64 2.1.2-0ubuntu1 [134 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libjpeg8 amd64 8c-2ubuntu1 [2264 B]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libgd3 amd64 2.1-3.1ubuntu0.22.04.1 [29.2 kB]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libwebp7 amd64 1.2.2-2ubuntu0.22.04.1 [206 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libtiff5 amd64 4.3.0-6ubuntu0.5 [184 kB]
```

```
sudo systemctl status nginx:
```

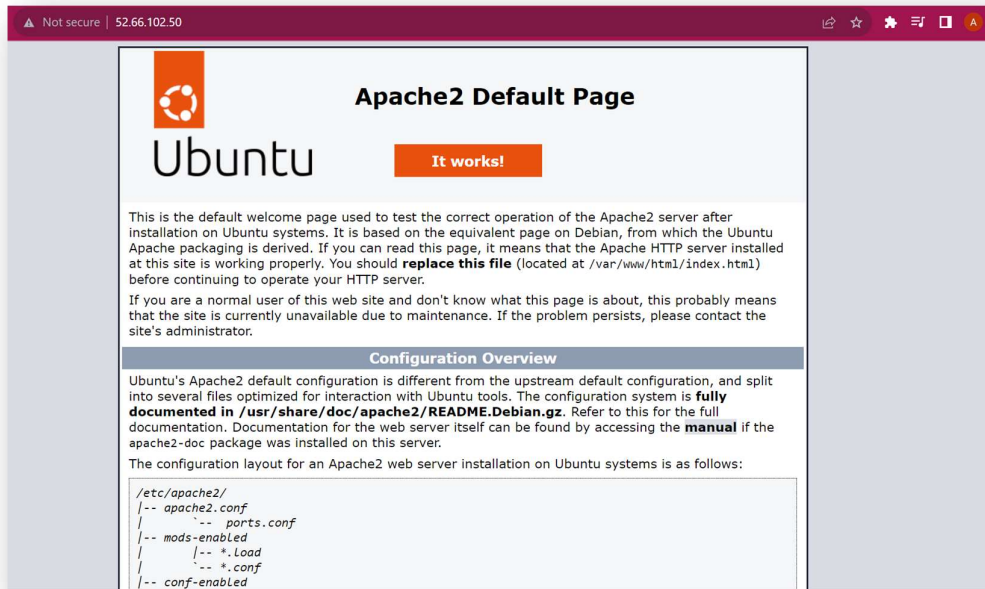
[illegible]

i-0e6d51d3741a3d586 (arpits-nginx-instance)

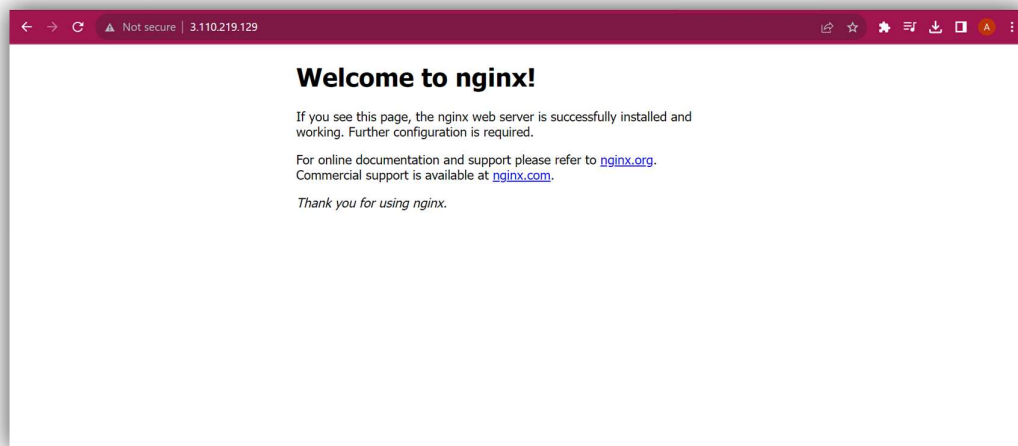
PublicIPs: 3.110.219.129 PrivateIPs: 172.31.10.60

Now, our Nginx installation is done and server is ready to use.

**10. Test the Web Servers:** For the Apache server, open a web browser and enter the public IP address or DNS name of the instance. For the Nginx server, do the same with its public IP address or DNS name.



This shows that **apache** server is **installed successfully**.



This shows that **nginx** server is **installed successfully**.

**RESULT:** Creation of two EC2 web servers one with the apache http server installed on it and one with the nginx server installed on it and all the installations using the user data section of ec2 is done successfully.