

Create AWS EC2 instance and install Apache2 server into it

AWS Services used:

1> EC2 Instance - Amazon EC2 (Elastic Compute Cloud) is a core compute service provided by Amazon Web Services (AWS) that allows you to launch and manage virtual servers, known as EC2 instances, in the cloud.

2> Apache 2: Apache HTTP Server, commonly known as Apache, is a popular open-source web server software developed and maintained by the Apache Software Foundation. It is one of the most widely used web server solutions globally.

Steps involved in the deployment:

- 1> Created an EC2 instance
- 2> SSH to EC2
- 3> Installed Apache 2

1> Create an EC2 instance

Go to AWS services -> EC2 -> Instances -> Launch Instance

The image shows two screenshots of the AWS Management Console. The top screenshot displays the 'EC2 Global view' for the Asia Pacific (Mumbai) Region. It shows a summary of resources: 1 running instance, 0 auto scaling groups, 1 elastic IP, 1 key pair, 0 load balancers, 2 security groups, 0 placement groups, and 1 volume. The bottom screenshot shows the 'Instances (1)' page, listing one instance named 'my-workpress...' with ID 'i-04cc258791435da30', which is in a 'Running' state. The instance type is 't2.micro' and it has passed 2/2 status checks.

Resource	Count
Instances (running)	1
Dedicated Hosts	0
Instances	1
Load balancers	0
Security groups	2
Volumes	1
Auto Scaling Groups	0
Elastic IPs	1
Key pairs	1
Placement groups	0
Snapshots	0

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
my-workpress...	i-04cc258791435da30	Running	t2.micro	2/2 checks passed	No alarms

aws Services Search [Alt+S] Mumbai Anub

EC2 > Instances > Launch an 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

 [Add additional tags](#)

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

▼ Summary

Number of instances [Info](#)

Software Image (AMI)

Canonical, Ubuntu, 22.04 LTS, ...[read more](#)

ami-0f5ee92e2d63afc18

Virtual server type (instance type)

t2.micro

Cancel

Launch instance

[Review commands](#)

Give EC2 instance name as "myapachewebserver"

Recents Quick Start

Amazon Linux

macOS

Ubuntu

Windows

Red Hat

[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

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

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

Free tier eligible

Description

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

Select Ubuntu OS type

After that generate a key pair for Ubuntu EC2 instance which will help us to SSH to our EC2 server remotely.

Key pair name

Key pairs allow you to connect to your instance securely.

The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ RSA
RSA encrypted private and public key pair

☐ ED25519
ED25519 encrypted private and public key pair

Private key file format

☒ .pem
For use with OpenSSH

☐ .ppk
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn](#)

Cancel Create key pair

Auto-assign public IP [Info](#)
Enable

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 Anywhere (0.0.0.0/0)
Helps you connect to your instance
- ☒ 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

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)
ami-0f5ee92e2d63afc18

Virtual server type (instance type)
t2.micro

[Cancel](#) [Launch instance](#)
[Review commands](#)

Allow all firewall security groups and from anywhere. Click on launch instance.

Instances (1) [Info](#) [Refresh](#) [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

Find instance by attribute or tag (case-sensitive)

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Actions
<input type="checkbox"/>	my-workpress...	i-04cc258791435da30	Running	t2.micro	2/2 checks passed	No alarms	Refresh Connect Instance state Actions

Instances

- Instances
- Instance Types
- Launch Templates
- Spot Requests

No, we have to assign elastic IP to our EC2 server or else Public IP V4 will keep on changing after restarting the instance.

Go to EC2 instance dashboard -> Elastic IP

Elastic IP addresses (1/1) [Refresh](#) [Actions](#) [Allocate Elastic IP address](#)

Filter Elastic IP addresses

<input checked="" type="checkbox"/>	Name	Allocated IPv4 address	Type	Allocation ID
<input checked="" type="checkbox"/>	-	15.207.192.248	Public IP	eipalloc-0996468e40ac39990

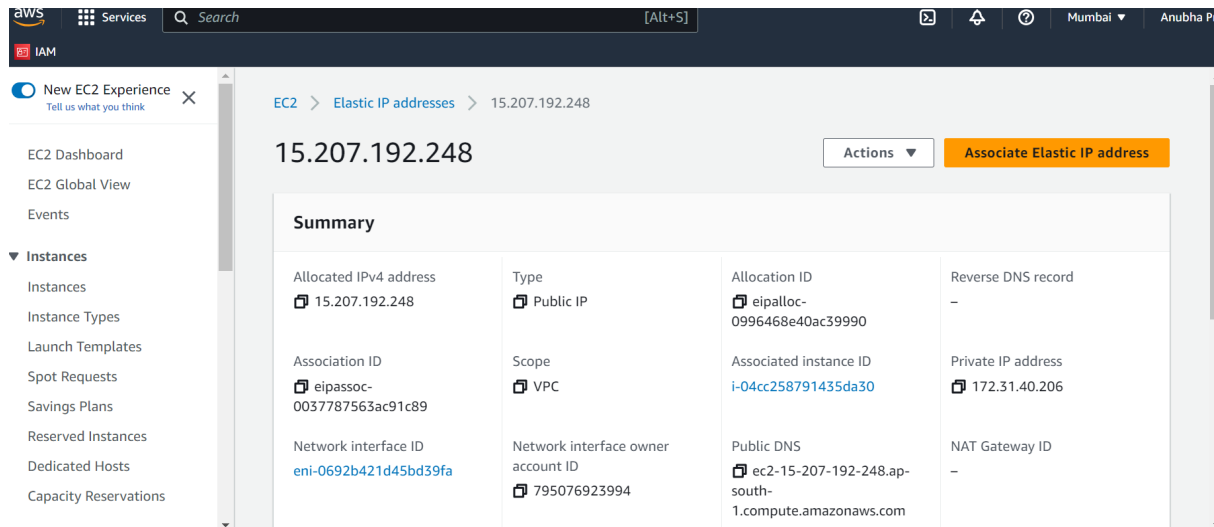
15.207.192.248

[Summary](#) [Tags](#)

Summary

Click on allocate elastic IP address and allocate elastic IP with all the default settings.

Now assign this elastic IP to our EC2 instance by associating the Elastic IP option -> Save



2> SSH to EC2

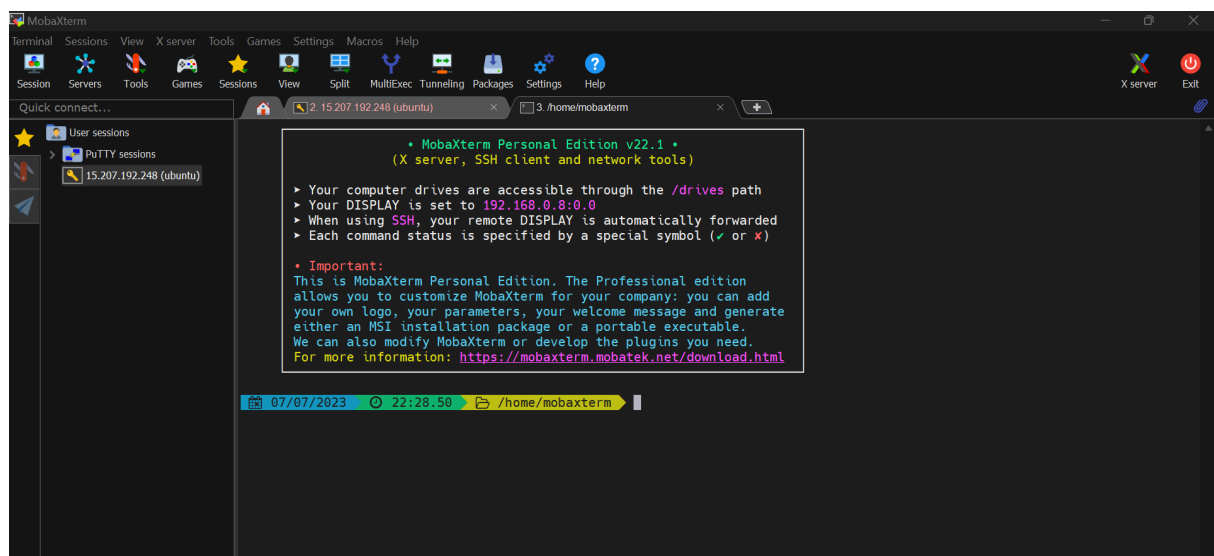
Go to our EC2 instance and copy Public IP V4 which will be the same to elastic IP this time.

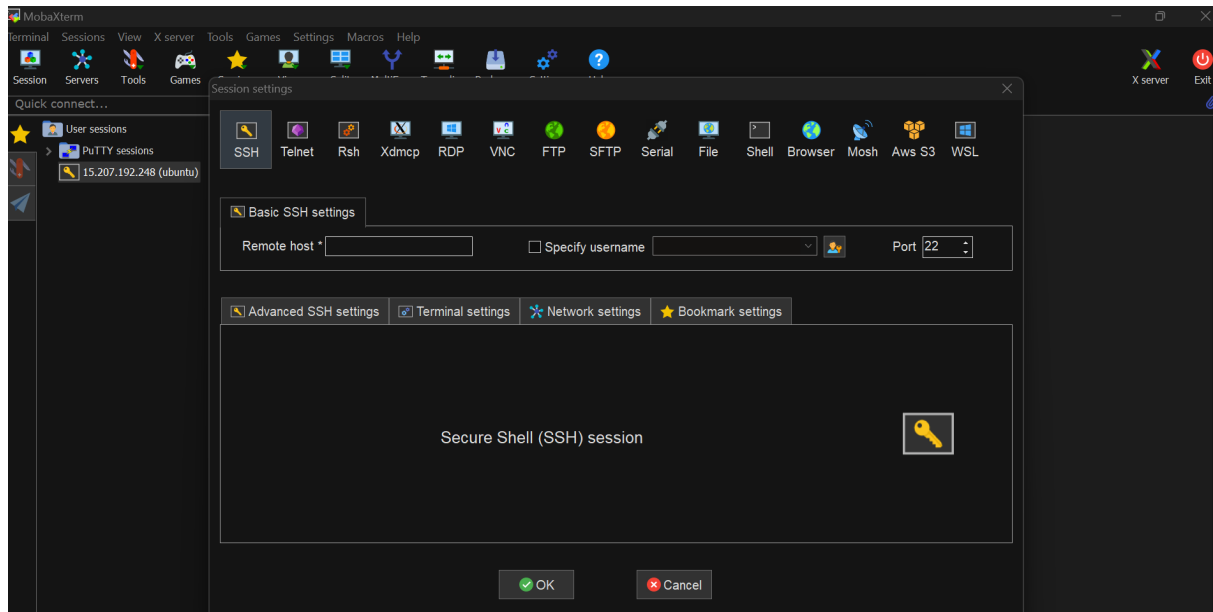
SSH: SSH (Secure Shell) is a network protocol that provides a secure and encrypted means of accessing and managing remote systems. It allows you to securely connect to a remote server or computer over an unsecured network, such as the internet.

Now in order to SSH to our EC2 instance we need one SSH agent if we are windows users. I am using MobaXterm here. It provides access to a Linux terminal on a Windows PC, allowing you to work with the Linux OS without downloading it on your system.

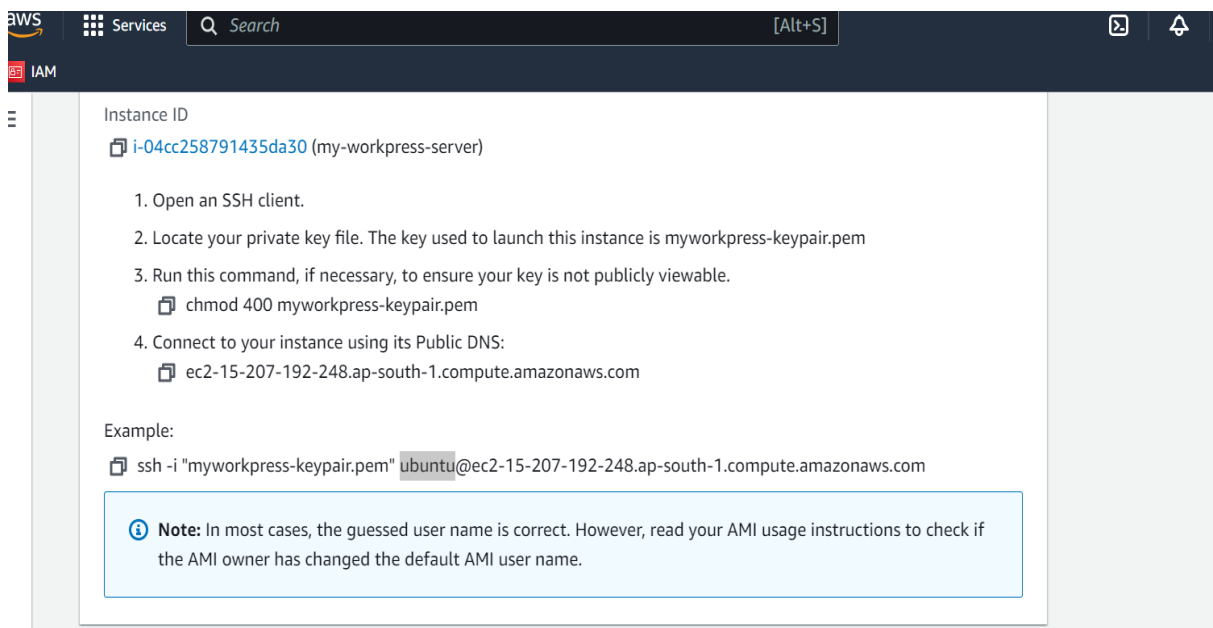
URL to download MobaXterm: <https://downloads.digitaltrends.com/moba/windows>

After installing it ->open ->session





Session -> SSH -> Remote host: Public IPV4 of our EC2 instance and Specify username: ubuntu -> Okay and it will connect to our EC2 instance.



3> Install Apache 2

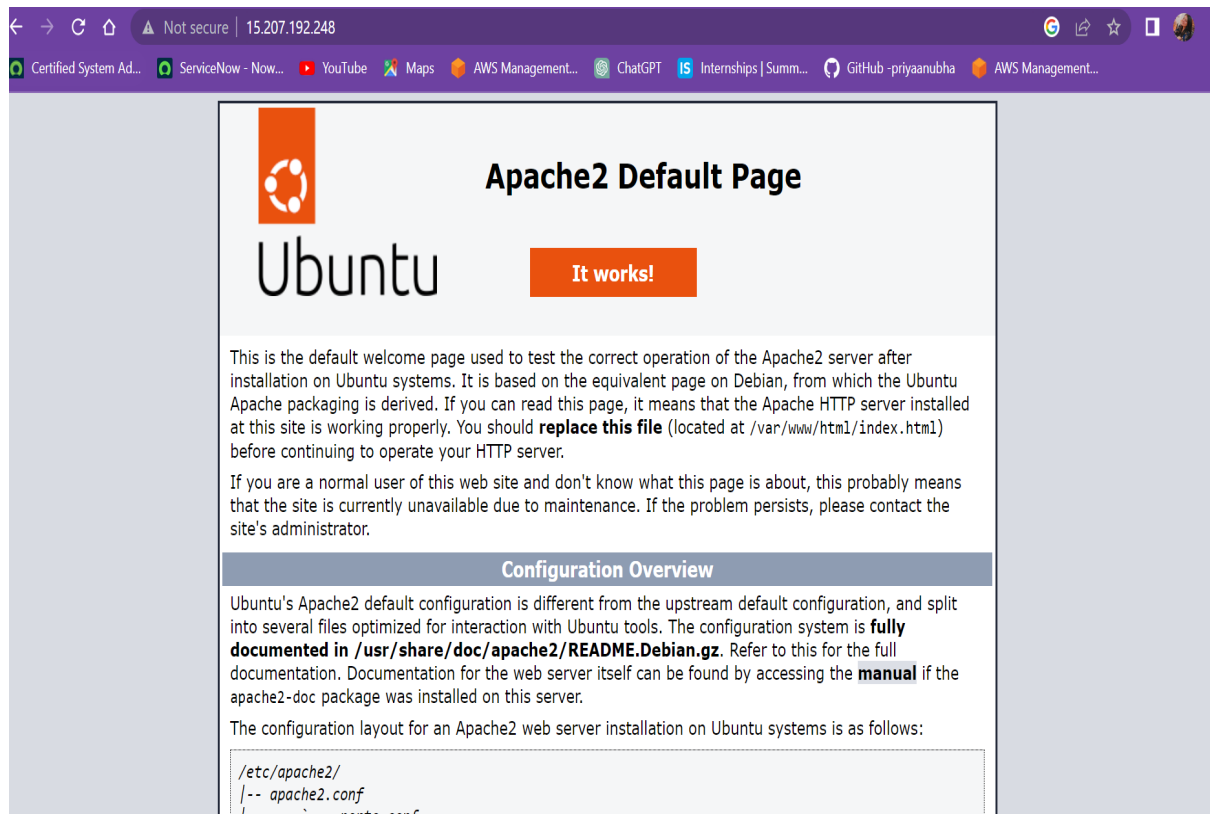
Command to install Apache 2 web server on our EC2 instance:

```
sudo apt install apache2
```

Here we have installed Apache 2 web server successfully to our EC2 instance which we can check via the public IPV4 of our EC2 instance

URL: `http:// <<Public IPV4 of EC2 instance>>`

Output:



So we have successfully created our EC2 server and installed Apache 2 web server on it via SSH.