

Launch Apache2 Server in AWS EC2 Instance

2 ways to hold web page on a server: AWS S3 or AWS EC2.

S3 needs you to store your website files (js, html, css) in its "S3 bucket".

EC2 needs you to run a server on the system and put website files in certain folder.

Before launching:

A running AWS EC2 instance (in **Ubuntu** system)

Launch Instance ▾ Connect Actions ▾									
Filter by tags and attributes or search by keyword								1 to 2 of 2	
Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)		
WebServer	i-095ea1b771eceb744	t2.micro	us-east-1a	running	2/2 checks passed	None	ec2-54-175-119-77.compute-1.amazonaws.com		

keypair file, e.g. my_first_aws_keypair.pem

Filezilla

Steps:

1. Connect remote system with SSH

```
xingjis-air:~ xingjichen$ ssh -i my_first_aws_keypair.pem ubuntu@54.***.****.77
```

-> Make sure we have the keypair file in current directory.

2. (Optional) Switch to root user (in case permission denied)

```
$ sudo su
```

3. Install Apache2 and check status

```
$ sudo apt-get install apache2
```

-> This may take a few seconds

```
$ sudo systemctl status apache2.service
```

-> check server status

```
[ubuntu@ip-172-31-18-10:~]$ sudo systemctl status apache2.service
● apache2.service - LSB: Apache2 web server
   Loaded: loaded (/etc/init.d/apache2; bad; vendor preset: enabled)
   Drop-In: /lib/systemd/system/apache2.service.d
            └─apache2-systemd.conf
   Active: active (running) since Tue 2018-12-18 22:55:36 UTC; 4h 1min ago
     Docs: man:systemd-sysv-generator(8)
    CGroup: /system.slice/apache2.service
            └─2111 /usr/sbin/apache2 -k start
               2114 /usr/sbin/apache2 -k start
               2115 /usr/sbin/apache2 -k start

Dec 18 22:55:35 ip-172-31-18-10 systemd[1]: Starting LSB: Apache2 web server...
Dec 18 22:55:35 ip-172-31-18-10 apache2[2094]: * Starting Apache httpd web serv
Dec 18 22:55:36 ip-172-31-18-10 apache2[2094]: *
Dec 18 22:55:36 ip-172-31-18-10 systemd[1]: Started LSB: Apache2 web server.
```

```
$ sudo systemctl stop apache2.service
```

-> Stop server, and we will see

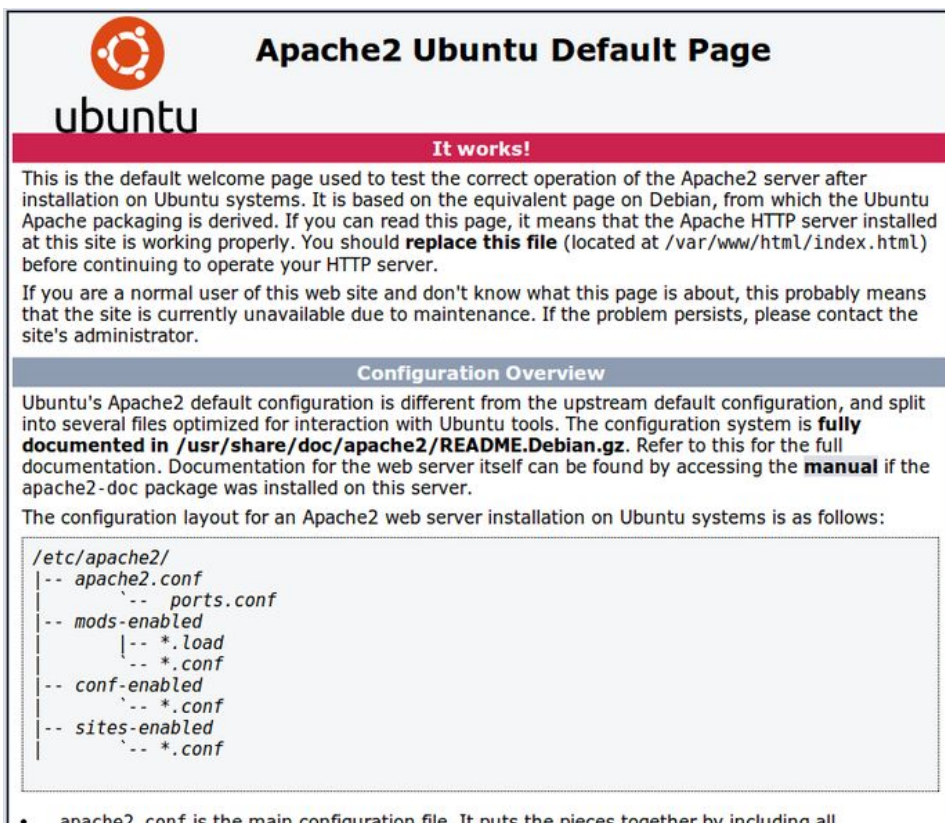
```
Drop-In: /lib/systemd/system/ap
└─apache2-systemd.conf
Active: inactive (dead) since 1
Docs: man:systemd-sysv-gener
Process: 2700 ExecStop=/etc/ini
```

```
$ sudo systemctl start apache2.service
```

-> Start server.

4. Deploy webpage

After step 3, we should see default Apache server page [index.html](#) if we enter the public IP address of our instance in browser



Apache2 Ubuntu Default Page

ubuntu

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

Now we need to replace it with our own webpage/files

```
$ cd /var/www/html
```

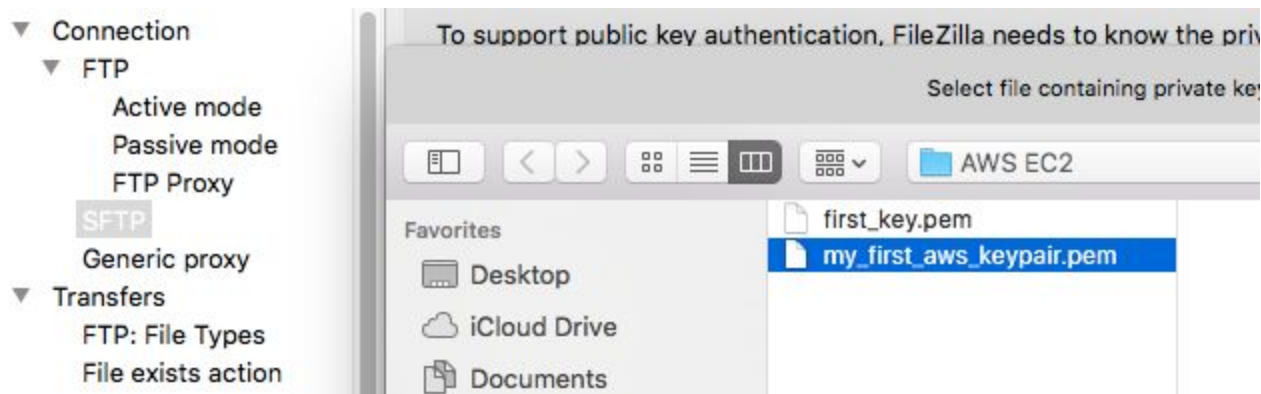
```
$ ls
```

-> you will see [index.html](#) in this directory

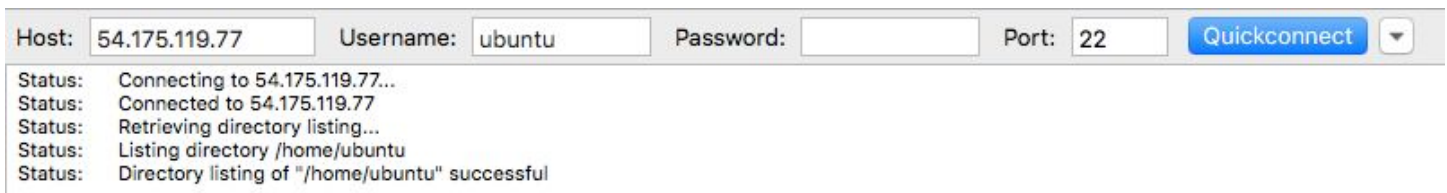
In order to move files to this folder, we need to change owner of `/www` to “ubuntu” (default owner = “root” and Filezilla doesn’t have root access)

```
$ sudo chown -R ubuntu /var/www
```

Start Filezilla, go to settings, click SFTP, click “add key file”
Then choose the keypair file in our PC



Connect to remote system



Redirect to /var in Filezilla and see /www has owner “ubuntu”

Remote site: /var					
Filename	Filesize	Filetype	Last modified	Permissions	Owner/Group
..					
backups		Directory	11/14/20...	drwxr-xr-x	root root
cache		Directory	12/18/20...	drwxr-xr-x	root root
lib		Directory	10/18/20...	drwxr-xr-x	root root
opt		Directory	10/12/20...	drwxr-xr-x	root root
snap		Directory	10/17/20...	drwxr-xr-x	root root
spool		Directory	10/12/20...	drwxr-xr-x	root root
www		Directory	10/18/20...	drwxr-xr-x	ubuntu root
local		Directory	04/12/20...	drwxrwsr-x	root staff
mail		Directory	10/12/20...	drwxrwsr-x	root mail
log		Directory	12/18/20...	drwxrwxr-x	root syslog
crash		Directory	10/12/20...	drwxrwxrwt	root root
tmp		Directory	12/18/20...	drwxrwxrwt	root root
lock		Directory	10/12/20...	lrwxrwxrwx	root root
run		Directory	10/12/20...	lrwxrwxrwx	root root

Go to /var/www/html and drag your webpage files (js,css,html) into the directory.