



Assignment 2 - Cloud Server Project Documentation

Created by Tom Dutch 34089911 for ICT171 Assignment 2: Cloud Server Project

This documentation will contain a list of steps taken to deploy the cloud server, along with the development of the script on the website.

Website located at studylist.space

Deploying an Amazon Web Services EC2 Instance

The first step required is the launching of a new Amazon EC2 Ubuntu instance.

Go to [AWS EC2 Console](#) and log in using your existing account.

From the EC2 Console, in the navigation pane, click on "Instances" and then "Launch Instances" on the top right.

Launching an Instance

Follow the steps below for launching a Ubuntu free tier instance:

- Name the instance accordingly - *Studylist.space web server*
- Pick the Linux Distribution as Ubuntu and select **Ubuntu Server 24.04 LTS (HVM)**
 - Version 24.04 LTS is used instead of 22.04 LTS as it has improved security and better long-term support.
- Choose the *t2.micro* instance type - ensure settings match the image below.

▼ **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

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

Recents **Quick Start**

Amazon Linux

macOS

Ubuntu

Windows

Red Hat

SUSE Linux

Debian

[Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type Free tier eligible

ami-0f5d1713c9af4fe30 (64-bit (x86)) / ami-099eeb58169040255 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Canonical, Ubuntu, 24.04, amd64 noble image

Architecture	AMI ID	Publish Date	Username	
64-bit (x86)	ami-0f5d1713c9af4fe30	2025-03-05	ubuntu	Verified provider

▼ **Instance type** [Info](#) | [Get advice](#)

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true On-Demand SUSE base pricing: 0.0146 USD per Hour On-Demand Linux base pricing: 0.0146 USD per Hour On-Demand Windows base pricing: 0.0192 USD per Hour On-Demand RHEL base pricing: 0.029 USD per Hour On-Demand Ubuntu Pro base pricing: 0.0164 USD per Hour

Free tier eligible

☐ All generations [Compare instance types](#)

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

- Create a new Key Pair - name it LoginKey and click "Create key pair" - download the Key Pair.
- Click "Create Security Group" and select the following:
 - Allow SSH traffic from - set IP Address to Anywhere (0.0.0.0/0).
 - Allow HTTPS traffic from the internet.
 - Allow HTTP traffic from the internet.
- Configure storage as 1x 30GiB gp3 Root volume, 3000 IOPS, Not encrypted .
- Once the above steps are complete, click "Launch Instance".
- Once done, click "View All Instances" and check the instance's state shows "Running".
 - To preserve free tier hours, stop the instance until ready to proceed to the next step.
 - To stop: Select the instance, click instance state and 'Stop Instance'.

Accessing the Instance

On the Amazon EC2 Instances page, in the list of instances, select the web server and in the bottom pop up, note down the Public IPv4 Address for later use.

To access the instance via terminal access, use SSH via Windows PowerShell.

Go to the directory containing 'LoginKey.pem' and in the directory bar type in 'powershell' to launch Windows PowerShell.

Paste the below command, replacing <IP_ADDRESS> with the one gathered earlier.

```
ssh -i ./LoginKey.pem ubuntu@<IP_ADDRESS>
```

When prompted with "Are you sure you want to continue connecting (yes/no/[fingerprint])?", type `yes` and hit enter.

Your CLI output should match below if successfully connected.

```
PS C:\Users\thoma\OneDrive\Desktop> ssh -i .\LoginKey.pem ubuntu@3.27.208.31
The authenticity of host '3.27.208.31 (3.27.208.31)' can't be established.
ED25519 key fingerprint is SHA256:ee2SgQ0a438Ne5hNB0joAfZY+MNeHJlFGf8diEmUak.
This host key is known by the following other names/addresses:
  C:\Users\thoma/.ssh/known_hosts:13: 13.211.77.106
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.27.208.31' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)

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

System information as of Wed Jun  4 13:47:32 UTC 2025

System load:  0.0      Processes:            106
Usage of /:   12.7% of 18.33GB   Users logged in:     0
Memory usage: 21%      IPv4 address for enx0: 172.31.6.10
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

10 updates can be applied immediately.
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

Last login: Tue Jun  3 15:08:51 2025 from [REDACTED]
```

Misc Configuration

Updating Packages

For good practice, upgrade all system packages using the following commands:

```
sudo apt update
```

Then:

```
sudo apt upgrade -y
```

Once complete, all system packages should be up to date.

Enabling Ubuntu Firewall

For security, it's a good idea to enable Ubuntu Firewall to minimise vulnerability.

Run the following commands:

```
sudo ufw allow ssh
```

Then:

```
sudo ufw allow http
```

Then:

```
sudo ufw allow https
```

Finally:

```
sudo ufw enable
```

Verify the correct setup of ubuntu firewall by running the below command:

```
sudo ufw status
```

The output of this command should match the below.

```
ubuntu@ip-172-31-6-10:~$ sudo ufw status
Status: active

To                Action            From
--                -
22/tcp            ALLOW            Anywhere
80/tcp            ALLOW            Anywhere
443              ALLOW            Anywhere
22/tcp (v6)       ALLOW            Anywhere (v6)
80/tcp (v6)       ALLOW            Anywhere (v6)
443 (v6)          ALLOW            Anywhere (v6)
```

Installing Apache Web Server

Apache web server is required to host the web page and can be installed by the following steps.

```
sudo apt install apache2 -y
```

To verify installation, in your own internet browser, navigate to `http://<IP_ADDRESS>/`

This should load the Apache2 Default Page shown below.



Apache2 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
```

Setting up DNS Record

Log into [Namecheap](#) open the 'Dashboard' tab when hovering over the username.

Next to the domain name 'studylist.space', click 'MANAGE'. Navigate to the 'Advanced DNS' Tab.

Under 'Host Records' create the following records:

Type	Host	Value	TTL
A Record	www	<IP_ADDRESS>	Automatic
CNAME Record	@	www.studylist.space	Automatic

Setting up HTTPS

HTTPS must be set up for secure connection to the website. This will be set up using the free service called Let's Encrypt.

Run the following commands once logged into server via SSH.

Install Certbot to set up SSL/TLS certificate.

```
sudo snap install --classic certbot
```

This command will take several minutes due to the limited bandwidth of the t2.micro instance.

Once complete, you should see the image below as output.

```
ubuntu@ip-172-31-6-10:~$ sudo snap install --classic certbot
certbot 4.0.0 from Certbot Project (certbot-eff/) installed
ubuntu@ip-172-31-6-10:~$ |
```

Then run the below command to allow certbot to be run:

```
sudo ln -s /snap/bin/certbot /usr/bin/certbot
```

To set up certbot, run:

```
sudo certbot --apache
```

When prompted for an email, hit 'Enter' and enter the domain name `studylist.space`.

The following output should be displayed.

```

ubuntu@ip-172-31-6-10:~$ sudo certbot --apache
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Enter email address or hit Enter to skip.
(Enter 'c' to cancel):

-----
Please read the Terms of Service at:
https://letsencrypt.org/documents/LE-SA-v1.5-February-24-2025.pdf
You must agree in order to register with the ACME server. Do you agree?
-----
(Y)es/(N)o: Y
Account registered.
Please enter the domain name(s) you would like on your certificate (comma and/or
space separated) (Enter 'c' to cancel): studylist.space
Requesting a certificate for studylist.space

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/studylist.space/fullchain.pem
Key is saved at: /etc/letsencrypt/live/studylist.space/privkey.pem
This certificate expires on 2025-09-02.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

Deploying certificate
Successfully deployed certificate for studylist.space to /etc/apache2/sites-available/000-default-le-ssl.conf
Congratulations! You have successfully enabled HTTPS on https://studylist.space

-----
If you like Certbot, please consider supporting our work by:
 * Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate
 * Donating to EFF: https://eff.org/donate-le
-----
ubuntu@ip-172-31-6-10:~$

```

Once complete, SSL should be set up and HTTPS should work. Navigate to <https://studylist.space> to verify the operation of a HTTPS connection.

That is the setup complete