35162293 – Cloud Server Project Documentation

Student Name: Nayeem Faisal

Student ID: 35162293

Unit: ICT171 – Introduction to Server Environments and Architectures

Project: 35162293 – Photography Portfolio

Global IP Address: 170.64.162.30

Domain Name: https://35162293.com

GitHub Repository: https://github.com/nayeemfaisal1999/ICT171.git

Video Explainer Drive link:

https://drive.google.com/drive/folders/19ZxspOGaVGuCwd0rT8cu-I0a14v5NgeJ?fbclid=IwAR2-lQEurH45HKtJj0hR-Nw85NL_lYoRrarRBUvtscuJjB-ah2vHIRMEok0

Project Overview

I built and documented a small web stack on DigitalOcean. Here's what I did and why:

- **Server** – Deployed an Ubuntu 22.04 droplet and grabbed the public IP 170.64.162.30.
- **Domain & DNS** – Pointed 35162293.com (and www) to the droplet with two A-records.
- **Web stack** – Installed Apache, wiped the default page, and copied my HTML / CSS / JS site into /var/www/html.
- **HTTPS** – Ran Certbot to generate a Let's Encrypt certificate so the site loads over TLS.
- **Automation** – Wrote backup-site.sh; it tars and gzips /var/www/html into /root/backups using a timestamped filename.
- **Evidence** – Collected console logs, command snippets, and screenshots at each stage so anyone in ICT171 can rebuild the same setup in about an hour.

The project gave me practice with Linux, IaaS, DNS, basic hardening, and a bit of Bash scripting. All skills I'll keep using in later units and side projects.

Server Setup Steps

Cloud Provider: DigitalOcean

Droplet Type: Basic Droplet

Operating System: Ubuntu 22.04 (LTS)

Public IP Address: 170.64.162.30

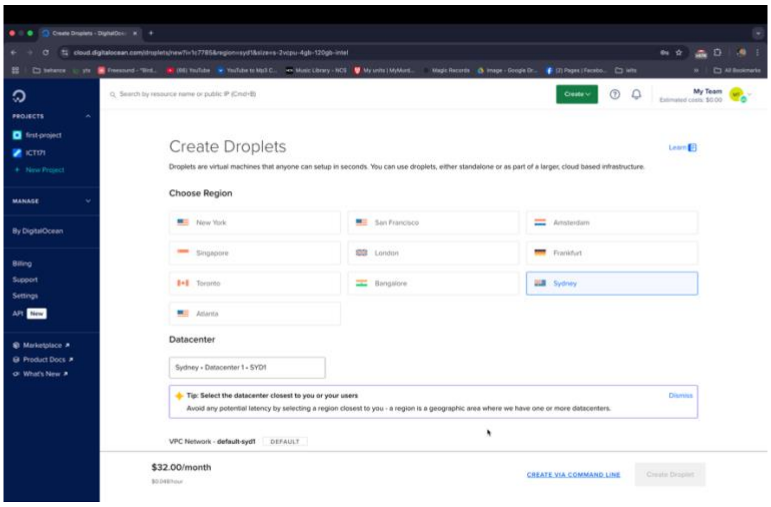
The droplet was created via DigitalOcean

Step 1: Create a Droplet on DigitalOcean

The server was deployed on DigitalOcean using the Ubuntu 22.04 (LTS) image.

Choose Ubuntu 22.04 LTS, allocate minimum 4GB RAM, and generate SSH

Keys for secure access.



cloud.digitalocean.com/droplets/new?i=1c7785&region=syd1&size=s-2vcpu-4gb-120gb-intel

Create Droplets

Droplets are virtual machines that anyone can setup in seconds. You can use droplets, either standalone or as part of a larger, cloud based infrastructure.

Choose Region: New York, San Francisco, Amsterdam, Singapore, London, Frankfurt, Toronto, Bangalore, [✓ Sydney], Atlanta

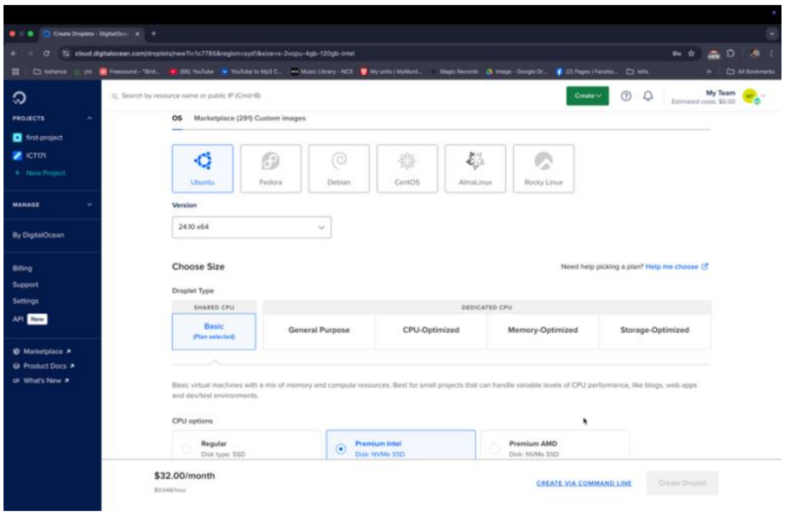
Datacenter: Sydney • Datacenter 1 • SYD1

Tip: Select the datacenter closest to you or your users
Avoid any potential latency by selecting a region closest to you - a region is a geographic area where we have one or more datacenters.

VPC Network - default-syd1 DEFAULT

$32.00/month

Screenshot: DigitalOcean Droplet Setup interface



cloud.digitalocean.com/droplets/new?i=1c7785&region=syd1&size=s-2vcpu-4gb-120gb-intel

OS: [✓ Ubuntu], Fedora, Debian, CentOS, AlmaLinux, Rocky Linux

Version: 24.10 x64

Choose Size

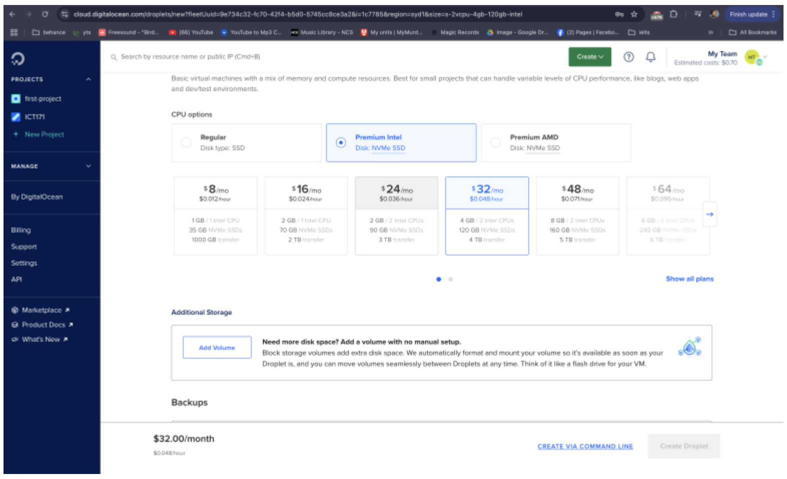
Droplet Type: SHARED CPU: [✓ Basic (Plan selected)]; DEDICATED CPU: General Purpose, CPU-Optimized, Memory-Optimized, Storage-Optimized

Basic virtual machines with a mix of memory and compute resources. Best for small projects that can handle variable levels of CPU performance, like blogs, web apps and dev/test environments.

CPU options: Regular (Disk type: SSD), [✓ Premium Intel (Disk: NVMe SSD)], Premium AMD (Disk: NVMe SSD)

$32.00/month

Screenshot: DigitalOcean Droplet Setup interface



cloud.digitalocean.com/droplets/new?fleetUuid=9e734c32-fc70-42f4-b5d0-5745cc8ce3a2&i=1c7785&region=syd1&size=s-2vcpu-4gb-120gb-intel
Search by resource name or public IP (Cmd+B)
Create
My Team
Estimated costs: $0.70
PROJECTS
first-project
ICT171
New Project
MANAGE
By DigitalOcean
Billing
Support
Settings
API
Marketplace
Product Docs
What's New

Basic virtual machines with a mix of memory and compute resources. Best for small projects that can handle variable levels of CPU performance, like blogs, web apps and dev/test environments.

CPU options

Regular — Disk type: SSD
[✓ Premium Intel — Disk: NVMe SSD]
Premium AMD — Disk: NVMe SSD

$8/mo	$16/mo	$24/mo	[✓ $32/mo]	$48/mo	$64/mo
$0.012/hour	$0.024/hour	$0.036/hour	$0.048/hour	$0.071/hour	$0.095/hour
1 GB / 1 Intel CPU	2 GB / 1 Intel CPU	2 GB / 2 Intel CPUs	4 GB / 2 Intel CPUs	8 GB / 2 Intel CPUs	8 GB / 4 Intel CPUs
35 GB NVMe SSDs	70 GB NVMe SSDs	90 GB NVMe SSDs	120 GB NVMe SSDs	160 GB NVMe SSDs	240 GB NVMe SSDs
1000 GB transfer	2 TB transfer	3 TB transfer	4 TB transfer	5 TB transfer	6 TB transfer

Show all plans

Additional Storage

Add Volume

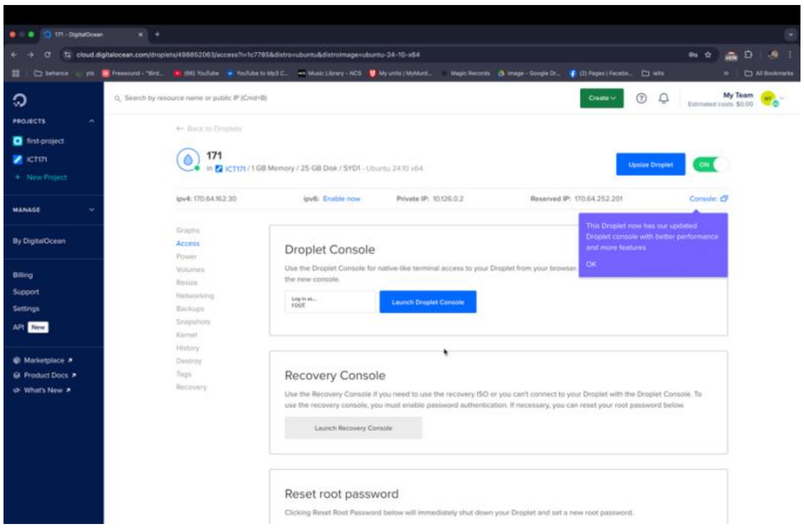
Need more disk space? Add a volume with no manual setup.
Block storage volumes add extra disk space. We automatically format and mount your volume so it's available as soon as your Droplet is, and you can move volumes seamlessly between Droplets at any time. Think of it like a flash drive for your VM.

Backups

$32.00/month
$0.048/hour

CREATE VIA COMMAND LINE
Create Droplet

Screenshot: DigitalOcean Droplet Setup interface



171
in ICT171 / 1 GB Memory / 25 GB Disk / SYD1 - Ubuntu 24.10 x64

ipv4: 170.64.162.30 ipv6: Enable now Private IP: 10.126.0.2 Reserved IP: 170.64.252.201 Console:

This Droplet now has our updated Droplet console with better performance and more features
OK

Droplet Console

Use the Droplet Console for native-like terminal access to your Droplet from your browser. the new console.

Log in as... root
Launch Droplet Console

Recovery Console

Use the Recovery Console if you need to use the recovery ISO or you can't connect to your Droplet with the Droplet Console. To use the recovery console, you must enable password authentication. If necessary, you can reset your root password below.

Launch Recovery Console

Reset root password

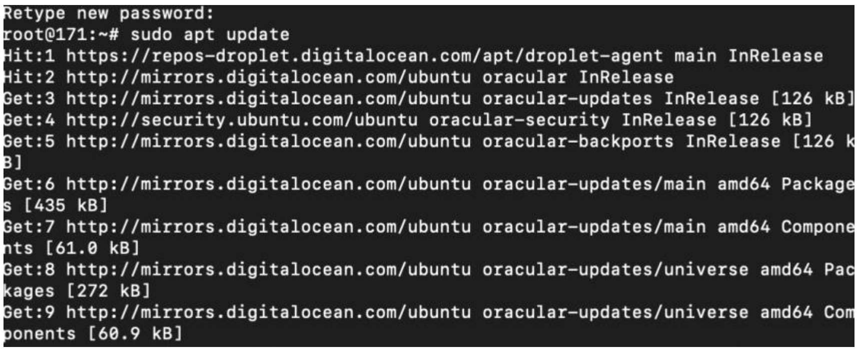
Clicking Reset Root Password below will immediately shut down your Droplet and set a new root password.

Screenshot: DigitalOcean Droplet Setup interface

No firewall rules or SSH keys were configured at this stage. Server access is currently done via root password, which will be changed later for security hardening.

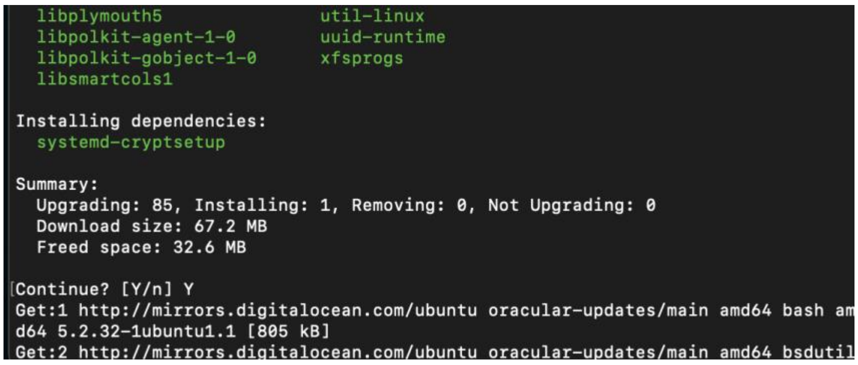
Step 2: Update the System

Sudo apt update



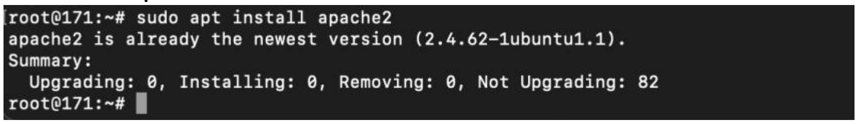
```
Retype new password:
root@171:~# sudo apt update
Hit:1 https://repos-droplet.digitalocean.com/apt/droplet-agent main InRelease
Hit:2 http://mirrors.digitalocean.com/ubuntu oracular InRelease
Get:3 http://mirrors.digitalocean.com/ubuntu oracular-updates InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu oracular-security InRelease [126 kB]
Get:5 http://mirrors.digitalocean.com/ubuntu oracular-backports InRelease [126 k
B]
Get:6 http://mirrors.digitalocean.com/ubuntu oracular-updates/main amd64 Package
s [435 kB]
Get:7 http://mirrors.digitalocean.com/ubuntu oracular-updates/main amd64 Compone
nts [61.0 kB]
Get:8 http://mirrors.digitalocean.com/ubuntu oracular-updates/universe amd64 Pac
kages [272 kB]
Get:9 http://mirrors.digitalocean.com/ubuntu oracular-updates/universe amd64 Com
ponents [60.9 kB]
```

Sudo apt upgrade – Y
Reason: Ensures the system has the latest security patches.

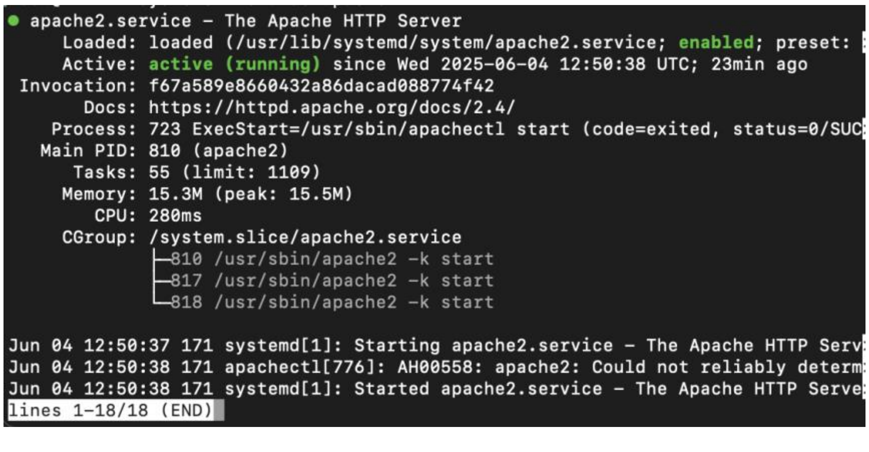


```
  libplymouth5              util-linux
  libpolkit-agent-1-0       uuid-runtime
  libpolkit-gobject-1-0     xfsprogs
  libsmartcols1

Installing dependencies:
  systemd-cryptsetup

Summary:
  Upgrading: 85, Installing: 1, Removing: 0, Not Upgrading: 0
  Download size: 67.2 MB
  Freed space: 32.6 MB

Continue? [Y/n] Y
Get:1 http://mirrors.digitalocean.com/ubuntu oracular-updates/main amd64 bash am
d64 5.2.32-1ubuntu1.1 [805 kB]
Get:2 http://mirrors.digitalocean.com/ubuntu oracular-updates/main amd64 bsdutil
```

Step 3: Install apache2
Sudo apt install apache2
Reason: Apache will serve the HTML website files.



```
root@171:~# sudo apt install apache2
apache2 is already the newest version (2.4.62-1ubuntu1.1).
Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 82
root@171:~# 
```

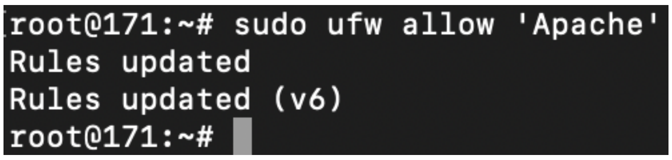


```
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: 
     Active: active (running) since Wed 2025-06-04 12:50:38 UTC; 23min ago
 Invocation: f67a589e8660432a86dacad088774f42
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 723 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUC
   Main PID: 810 (apache2)
      Tasks: 55 (limit: 1109)
     Memory: 15.3M (peak: 15.5M)
        CPU: 280ms
     CGroup: /system.slice/apache2.service
             ├─810 /usr/sbin/apache2 -k start
             ├─817 /usr/sbin/apache2 -k start
             └─818 /usr/sbin/apache2 -k start

Jun 04 12:50:37 171 systemd[1]: Starting apache2.service - The Apache HTTP Serv
Jun 04 12:50:38 171 apachectl[776]: AH00558: apache2: Could not reliably determ
Jun 04 12:50:38 171 systemd[1]: Started apache2.service - The Apache HTTP Serve
lines 1-18/18 (END)
```

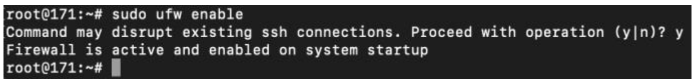
Step 4: Enable Apache and allow firewall

Sudo ufw allow ‘Apache’
Sudo enable apache
Sudo ufw status tart apache

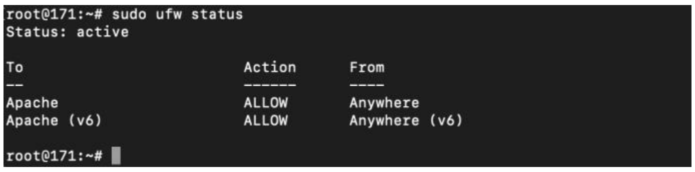
Enable Firewall (UFW)



```
root@171:~# sudo ufw allow 'Apache'
Rules updated
Rules updated (v6)
root@171:~#
```



```
root@171:~# sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
root@171:~#
```

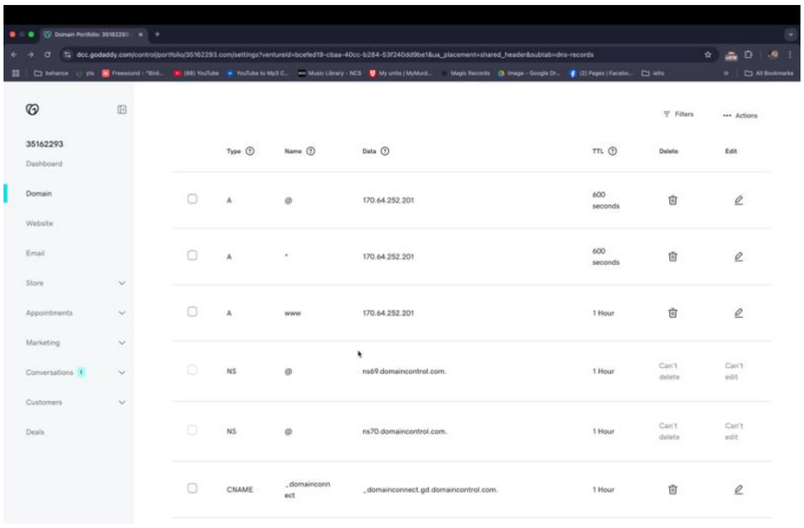


```
root@171:~# sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
Apache                     ALLOW       Anywhere
Apache (v6)                ALLOW       Anywhere (v6)

root@171:~#
```

DNS Configuration
Step 5: Point Domain to Server IP

A custom domain, 35162293.com, was registered previously.
Domain Name: www.35162293.com
DNS Provider: DigitalOcean DNS
DNS Records:
@ → 170.64.162.30
www → 170.64.162.30

The DNS setup ensures that both 35162293.com and www.35162293.com resolve to the server's IP. DNS propagation was confirmed via dig and ping.


Domain Portfolio: 35162293
dcc.godaddy.com/control/portfolio/35162293.com/settings?ventureId=bcefed19-cbaa-40cc-b284-53f240dd9be1&ua_placement=shared_header&subtab=dns-records
35162293
Dashboard
Domain
Website
Email
Store
Appointments
Marketing
Conversations
Customers
Deals
Filters
Actions

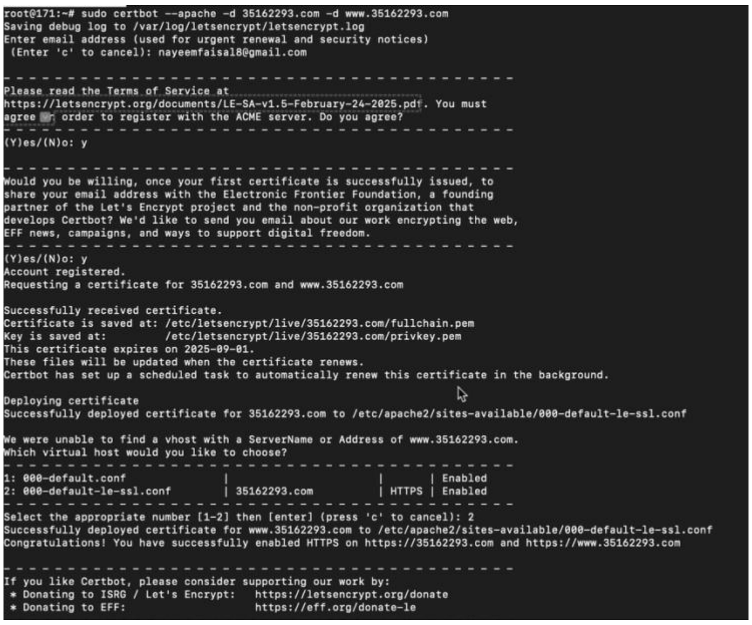
	Type	Name	Data	TTL	Delete	Edit
☐	A	@	170.64.252.201	600 seconds	🗑	✎
☐	A	*	170.64.252.201	600 seconds	🗑	✎
☐	A	www	170.64.252.201	1 Hour	🗑	✎
☐	NS	@	ns69.domaincontrol.com.	1 Hour	Can't delete	Can't edit
☐	NS	@	ns70.domaincontrol.com.	1 Hour	Can't delete	Can't edit
☐	CNAME	_domainconnect	_domainconnect.gd.domaincontrol.com.	1 Hour	🗑	✎

SSL/TLS with Certbot

Step6: Install Certbot and SSL

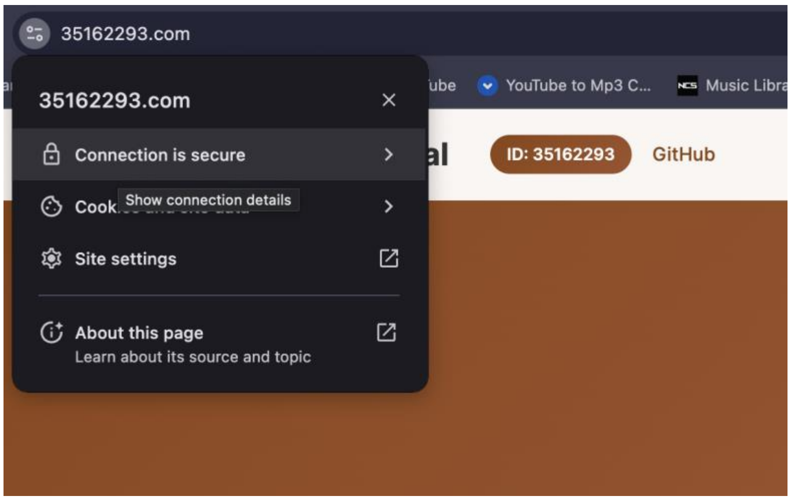
Sudo certbot –apache -d 35162293.com -d www. 35612293.com

Reason: Automatically secures your site with HTTPS using let's Encrypt.



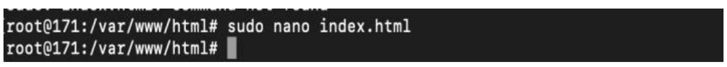
```
root@171:~# sudo certbot --apache -d 35162293.com -d www.35162293.com
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Enter email address (used for urgent renewal and security notices)
 (Enter 'c' to cancel): nayeemfaisal8@gmail.com

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
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

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
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.
Requesting a certificate for 35162293.com and www.35162293.com

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/35162293.com/fullchain.pem
Key is saved at:         /etc/letsencrypt/live/35162293.com/privkey.pem
This certificate expires on 2025-09-01.
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 35162293.com to /etc/apache2/sites-available/000-default-le-ssl.conf

We were unable to find a vhost with a ServerName or Address of www.35162293.com.
Which virtual host would you like to choose?
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
1: 000-default.conf            |                        |       | Enabled
2: 000-default-le-ssl.conf     | 35162293.com           | HTTPS | Enabled
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Select the appropriate number [1-2] then [enter] (press 'c' to cancel): 2
Successfully deployed certificate for www.35162293.com to /etc/apache2/sites-available/000-default-le-ssl.conf
Congratulations! You have successfully enabled HTTPS on https://35162293.com and https://www.35162293.com

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
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
```

Screenshot: Certbot SSL certificate installation

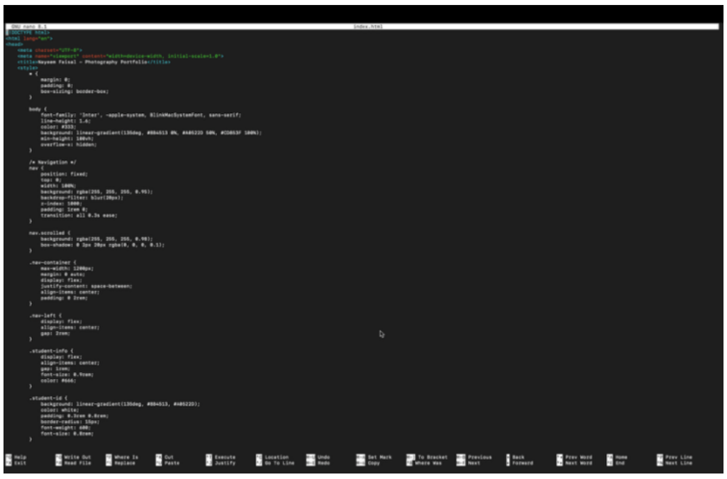


Deploy the website
Step 7: Upload Files to Sever
Root@170.64.162.30:/var/www/html

Reason: Copies your HTML/CSS/JS files to server's web directory

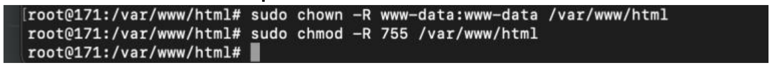


```
root@171:/var/www/html# sudo nano index.html
root@171:/var/www/html# 
```



```
  GNU nano 8.1                                   index.html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Nayeem Faisal - Photography Portfolio</title>
    <style>
        * {
            margin: 0;
            padding: 0;
            box-sizing: border-box;
        }

        body {
            font-family: 'Inter', -apple-system, BlinkMacSystemFont, sans-serif;
            line-height: 1.6;
            color: #333;
            background: linear-gradient(135deg, #8B4513 0%, #A0522D 50%, #CD853F 100%);
            min-height: 100vh;
            overflow-x: hidden;
        }

        /* Navigation */
        nav {
            position: fixed;
            top: 0;
            width: 100%;
            background: rgba(255, 255, 255, 0.95);
            backdrop-filter: blur(20px);
            z-index: 1000;
            padding: 1rem 0;
            transition: all 0.3s ease;
        }

        nav.scrolled {
            background: rgba(255, 255, 255, 0.98);
            box-shadow: 0 2px 20px rgba(0, 0, 0, 0.1);
        }

        .nav-container {
            max-width: 1200px;
            margin: 0 auto;
            display: flex;
            justify-content: space-between;
            align-items: center;
            padding: 0 2rem;
        }

        .nav-left {
            display: flex;
            align-items: center;
            gap: 2rem;
        }

        .student-info {
            display: flex;
            align-items: center;
            gap: 1rem;
            font-size: 0.9rem;
            color: #666;
        }

        .student-id {
            background: linear-gradient(135deg, #8B4513, #A0522D);
            color: white;
            padding: 0.3rem 0.8rem;
            border-radius: 15px;
            font-weight: 600;
            font-size: 0.8rem;
        }

^G Help      ^O Write Out  ^F Where Is   ^K Cut       ^T Execute   ^C Location   M-U Undo     M-A Set Mark
^X Exit      ^R Read File  ^\ Replace    ^U Paste     ^J Justify   ^/ Go To Line M-E Redo     M-6 Copy
```

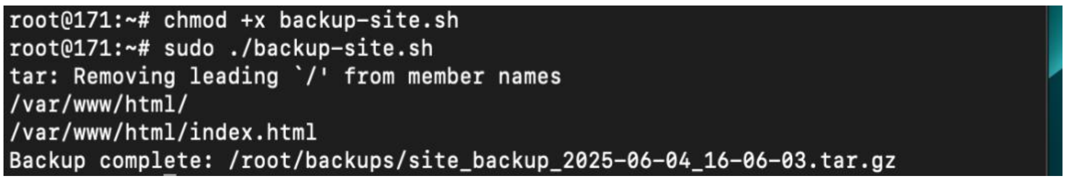
Step 8: Set Permissions
Sudo chwon -R www-data:www-data/var/www/html
Reason: Ensures Apache can read the website files.



```
root@171:/var/www/html# sudo chown -R www-data:www-data /var/www/html
root@171:/var/www/html# sudo chmod -R 755 /var/www/html
root@171:/var/www/html# 
```

z

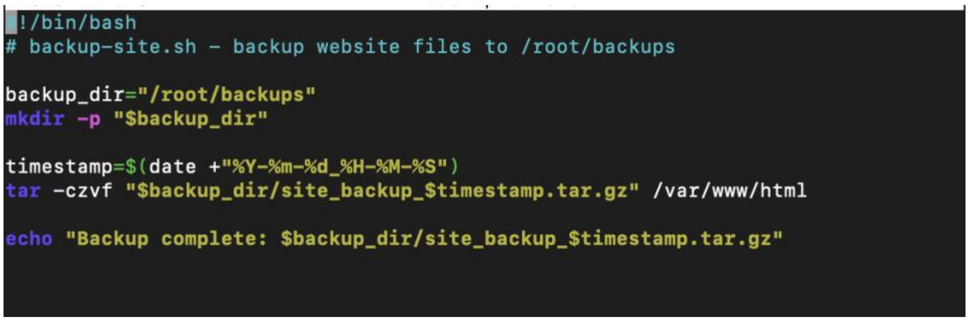
Bash Scipt: Server Monitor
Script Purpose: Checks if Apache is running and logs uptime
Step 9: Bash



```
root@171:~# chmod +x backup-site.sh
root@171:~# sudo ./backup-site.sh
tar: Removing leading `/' from member names
/var/www/html/
/var/www/html/index.html
Backup complete: /root/backups/site_backup_2025-06-04_16-06-03.tar.gz
```

Bash Script: Website Backup Utility

As part of this project, a Bash script was created to automate backups of the web server's content. The script compresses the contents of /var/www/html into a timestamped .tar.gz archive and stores it in /root/backups.
This script allows quick manual backups and can easily be scheduled via cron in future improvements.

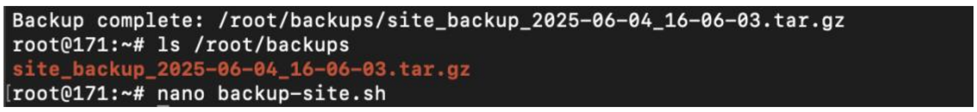


```
#!/bin/bash
# backup-site.sh - backup website files to /root/backups

backup_dir="/root/backups"
mkdir -p "$backup_dir"

timestamp=$(date +"%Y-%m-%d_%H-%M-%S")
tar -czvf "$backup_dir/site_backup_$timestamp.tar.gz" /var/www/html

echo "Backup complete: $backup_dir/site_backup_$timestamp.tar.gz"
```

What it does:

- **Line 1**: Specifies Bash as the interpreter.
- **Line 3**: Sets the target backup folder path.
- **Line 4**: Creates the folder if it doesn't exist.
- **Line 6**: Generates a timestamp to make each backup unique.
- **Line 7**: Uses tar to compress the website files into a .tar.gz archive.
- **Line 9**: Outputs a confirmation message

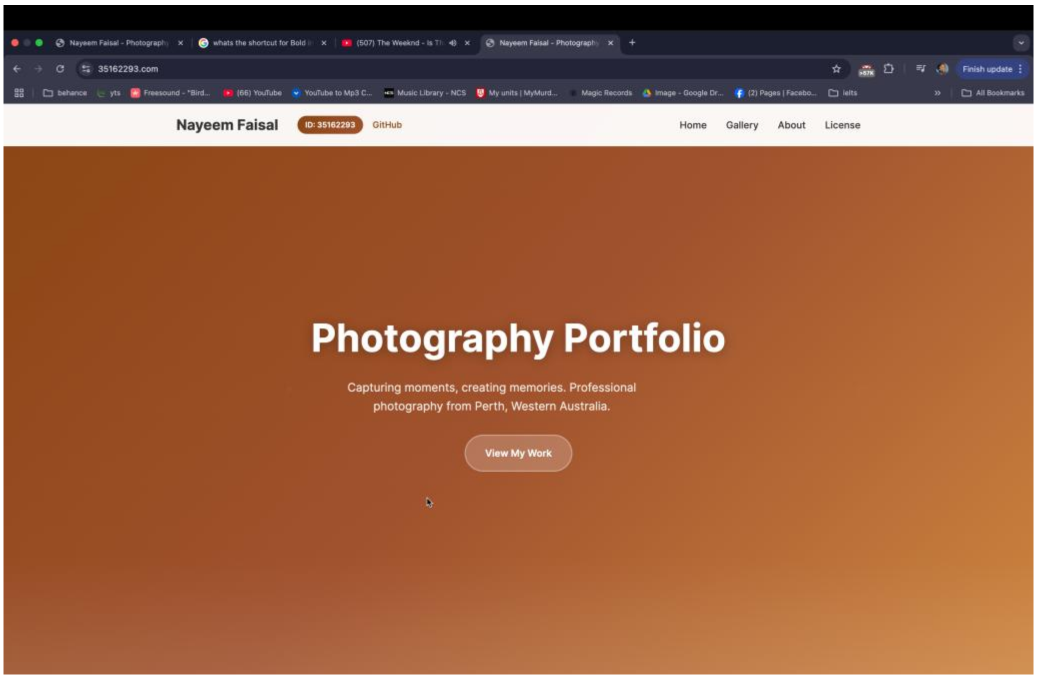
Backup



```
Backup complete: /root/backups/site_backup_2025-06-04_16-06-03.tar.gz
root@171:~# ls /root/backups
site_backup_2025-06-04_16-06-03.tar.gz
root@171:~# nano backup-site.sh
```

Testing on Browser

https://www.35162293.com



To conclude, this project involved building a secure, publicly accessible web server using infrastructure as a Iaas service on DigitalOcean. The server hosts a personal website created with HTML, CSS, and Javascript, and is accessible via a custom domain name.

What was built

Ubuntu 22.04 droplet on DigitalOcean (170.64.162.30).
Apache 2.4 serving my HTML/CSS/JS site from /var/www/html.
Domain 35162293.com (+ www) mapped with two A-records.
TLS via Let's Encrypt & Certbot; auto-renew in place.

Backup script backup-site.sh creates time-stamped .tar.gz archives under /root/backups.
Steps, commands, and screenshots are documented so a classmate could rebuild the server in under an hour.

References

DigitalOcean docs – “Install Apache on Ubuntu 22.04”.
Let’s Encrypt / Certbot – Apache guide