

Project Title:

Deployed Internee.pk website on AWS EC2 with Free Domain and HTTPS (SSL) using Nginx.

1. Introduction:

This project demonstrates deploying the **Internee.pk** clone website on a cloud-based server. The main goal was to make the website live, accessible through a free domain, and secure it using HTTPS. This deployment shows practical knowledge of server setup, domain configuration, web hosting, and website security.

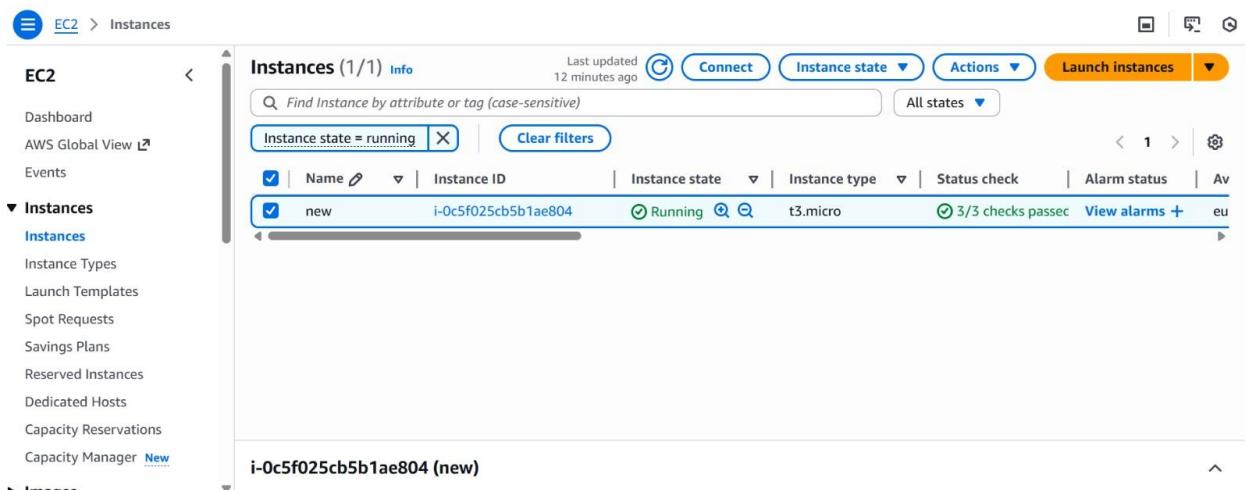
2. Tools Used:

- AWS EC2 Ubuntu Instance
- Nginx Web Server
- DuckDNS Free Domain
- Let's Encrypt SSL (via Certbot)
- SSH Client (MobaXterm)
- Web Browser for testing

3. Procedure:

Step 1: Cloud Server Setup

- Launched a virtual machine on AWS EC2.
- Configured network settings to allow web traffic (HTTP and HTTPS).
- Connected to the server using SSH client.



Step 2: Web Server Installation

- Installed Nginx to serve the website.
- Verified that the default Nginx welcome page was accessible.



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

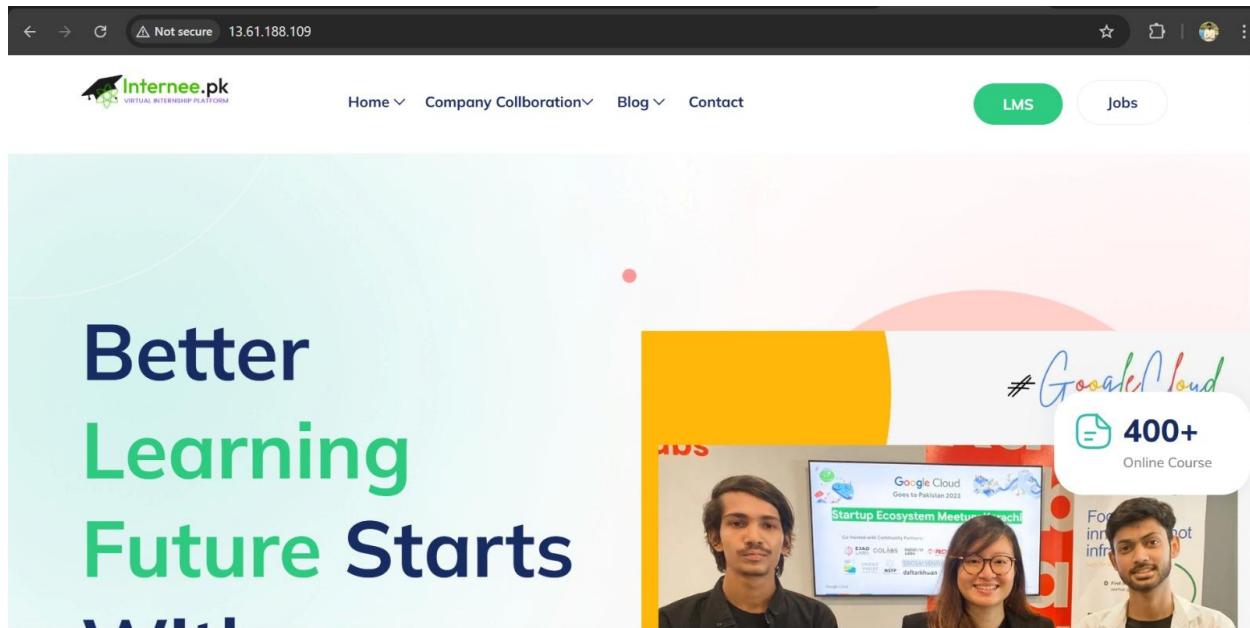
Step 3: Project Deployment

- Cloned the Internee.pk repository from GitHub.
- Placed project files in the web server's root directory.
- Ensured proper access permissions for files and folders.

```
root@ip-172-31-20-246:~# sudo apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.34.1-1ubuntu1.15).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ip-172-31-20-246:~# git clone https://github.com/zulqarnainahmed07/TSK-000-36-Internee-Clone.git
Cloning into 'TSK-000-36-Internee-Clone'...
remote: Enumerating objects: 204, done.
remote: Counting objects: 100% (204/204), done.
remote: Compressing objects: 100% (190/190), done.
remote: Total 204 (delta 23), reused 176 (delta 11), pack-reused 0 (from 0)
Receiving objects: 100% (204/204), 22.15 MiB | 33.06 MiB/s, done.
Resolving deltas: 100% (23/23), done.
root@ip-172-31-20-246:~# ll
total 40
drwx----- 5 root root 4096 Dec 28 09:46 .
drwxr-xr-x 19 root root 4096 Dec 28 08:36 ../
-rw-r--r-- 1 root root 3106 Oct 15 2021 .bashrc
-rw-r--r-- 1 root root 68 Dec 28 09:27 .gitconfig
-rw----- 1 root root 20 Dec 28 08:40 .lesshst
-rw-r--r-- 1 root root 161 Jul 9 2019 .profile
drwx----- 2 root root 4096 Dec 28 08:36 .ssh/
-rw-r--r-- 1 root root 0 Dec 28 08:38 .sudo_as_admin_successful
-rw----- 1 root root 776 Dec 28 09:25 .viminfo
drwxr-xr-x 4 root root 4096 Dec 28 09:46 TSK-000-36-Internee-Clone/
drwx----- 4 root root 4096 Dec 28 08:36 snap/
root@ip-172-31-20-246:~#
```

Step 4: Nginx Configuration for Domain

- Created a new Nginx configuration to link the domain with the project.
- Enabled the site and restarted the web server to apply changes.
- Verified that the project was accessible through the server's IP.



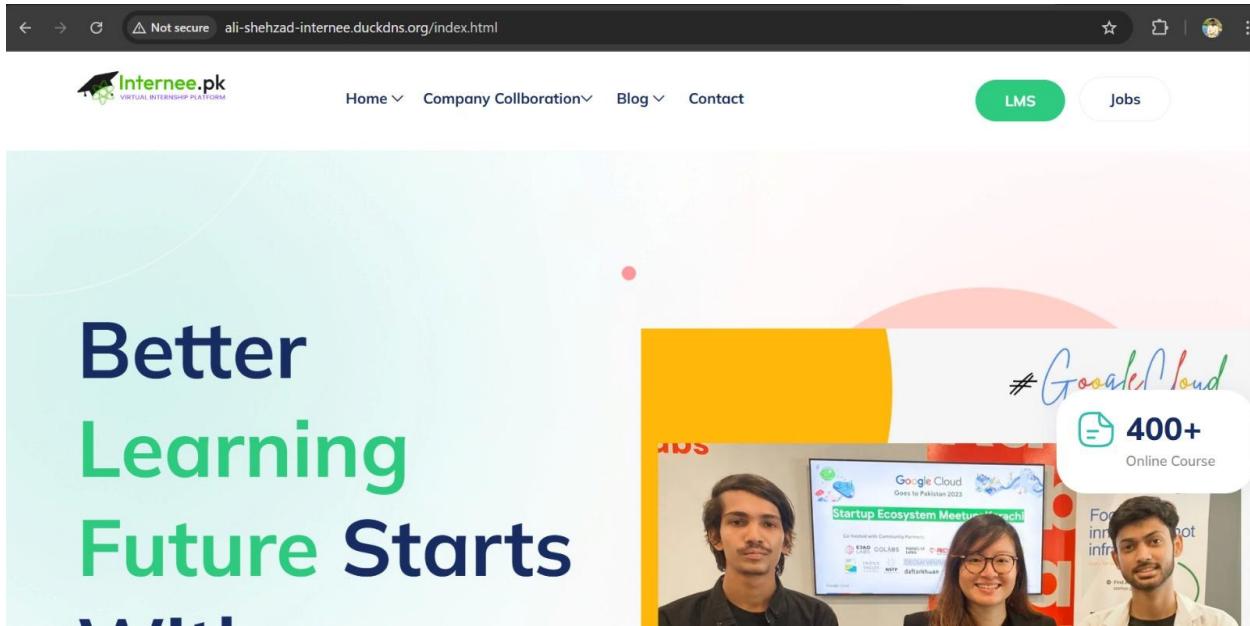
Step 5: Free Domain Configuration

- Registered a free subdomain `ali-shehzad-internee.duckdns.org` on [DuckDNS](#).
- Updated the domain's IP to point to the cloud server.
- Verified the website was accessible via the domain in a browser.

A screenshot of the Duck DNS website. The URL in the address bar is www.duckdns.org/domains. The page displays account information: account (princealishazad943@gmail.com), type (free), token (31d97643-82c4-43a3-b2b9-6c574e4c1f84), token generated (4 minutes ago), and created date (28 Dec 2025, 08:58:12). A large yellow rubber duck icon is on the left. Below the account info, a green success message box says "success: ip address for ali-shehzad-internee.duckdns.org updated to 13.61.188.109". At the bottom, there is a search bar for "domains" and a button to "add domain".

Step 6: Enabling HTTPS

- Installed SSL certificate from Let's Encrypt.
- Configured the website to automatically redirect all traffic to HTTPS.
- Verified the SSL certificate was active and browser connection was secure.
- Ensured SSL certificate will renew automatically in the background.



Terminal Output:

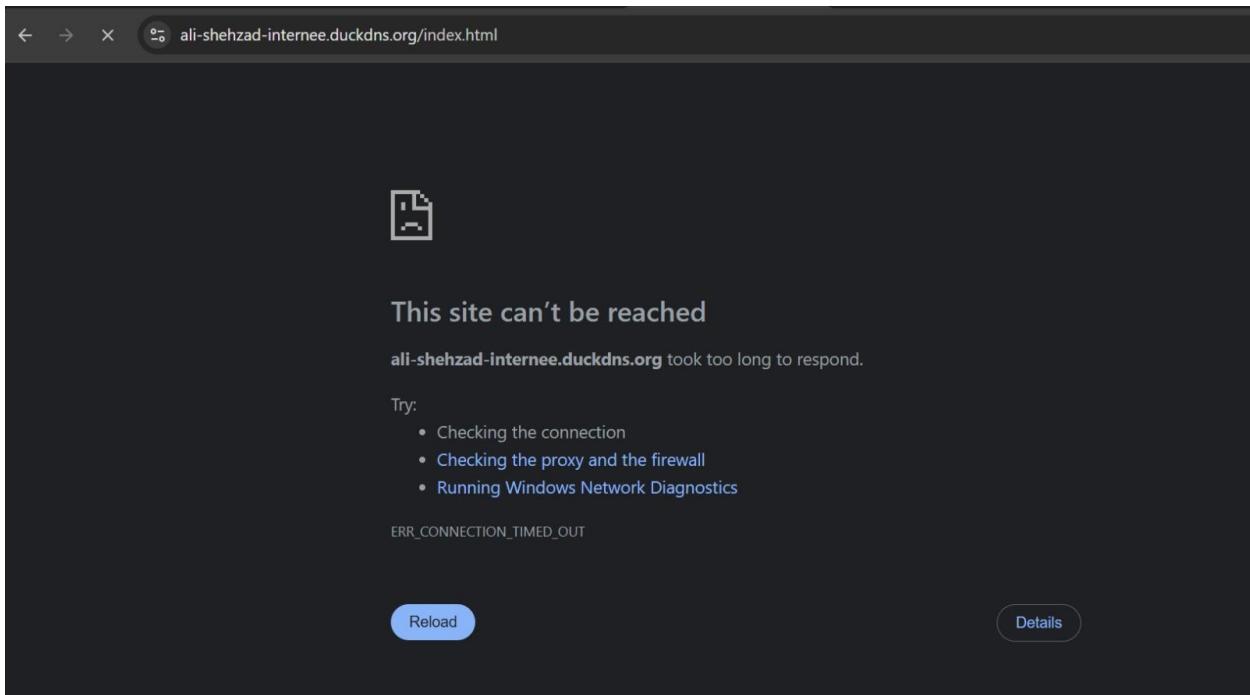
```
Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/ali-shehzad-internee.duckdns.org/fullchain.pem
Key is saved at:          /etc/letsencrypt/live/ali-shehzad-internee.duckdns.org/privkey.pem
This certificate expires on 2026-03-28.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

Deploying certificate
Successfully deployed certificate for ali-shehzad-internee.duckdns.org to /etc/nginx/sites-enabled/default
Congratulations! You have successfully enabled HTTPS on https://ali-shehzad-internee.duckdns.org

-----
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

root@ip-172-31-20-246:/home/ubuntu# sudo tail -f /var/log/nginx/access.log
sudo tail -f /var/log/nginx/error.log
209.97.180.8 - - [28/Dec/2025:09:13:03 +0000] "GET /config.json HTTP/1.1" 301 178 "-" "Mozilla/5.0 (l9scan/2.0.930313e2838313e21363e23313; +https://leakix.net)"
146.190.63.248 - - [28/Dec/2025:09:13:04 +0000] "GET /.git/config HTTP/1.1" 404 134 "-" "Mozilla/5.0 (l9scan/2.0.930313e2838313e21363e23313; +https://leakix.net)"
209.97.180.8 - - [28/Dec/2025:09:13:04 +0000] "GET /telescope/requests HTTP/1.1" 301 178 "-" "Mozilla/5.0 (l9scan/2.0.930313e2838313e21363e23313; +https://leakix.net)"
146.190.63.248 - - [28/Dec/2025:09:13:05 +0000] "GET /s/930313e2838313e21363e23313/_;/META-INF/maven/com.atlassian.jira/jira-webapp-dist/pom.properties HTTP/1.1" 404 134 "-" "Mozilla/5.0 (l9scan/2.0.930313e2838313e21363e23313; +https://leakix.net)"
209.97.180.8 - - [28/Dec/2025:09:13:05 +0000] "GET /info.php HTTP/1.1" 301 178 "-" "Mozilla/5.0 (l9scan/2.0.930313e2838313e21363e23313; +https://leakix.net)"
```

After OFF Server:



4. Conclusion:

- The Internee.pk clone was successfully deployed on AWS EC2.
- Configured a free DuckDNS domain to make the website accessible.
- Enabled HTTPS using Let's Encrypt SSL to secure the website.
- The project demonstrates cloud deployment, web server management, and domain setup skills.
- The website is live, secure, and fully functional.

GitHub History Code: [Link](#)

Project Vedio: [Link](#)