|  |
| --- |
| **Batch : A-3 Roll No. : 16010122104**  **Experiment No.: 10** |

**Title: Deploying a Full Stack Web Application on Hostinger VPS (Virtual Private Server)**

**Objective:** Hosting website on VPS

**Expected Outcome of Experiment:**

|  |  |
| --- | --- |
| **CO** | **Outcome** |
| CO4 | To deploy and run a full-stack web application on a VPS with proper backend,  frontend, domain configuration, and secure HTTPS access. |

**Books/ Journals/ Websites referred:**

<https://www.hostinger.in/tutorials/getting-started-with-vps-hosting>

# Abstract

This project involved deploying a full-stack web application on a **Hostinger VPS** using Ubuntu. The assignment aimed to provide hands-on experience with cloud infrastructure, web server configuration, and application deployment.

Key tasks included:

* Setting up a VPS and accessing it via SSH
* Hosting the frontend (React) using **NGINX**
* Deploying the backend using **Node.js and Express**
* Connecting a database (MongoDB/PostgreSQL)
* Managing server processes with **PM2**
* Securing the site with a custom domain and **SSL certificates**

This exercise demonstrated the real-world process of hosting and maintaining web applications on a cloud-based server.

# Related Theory

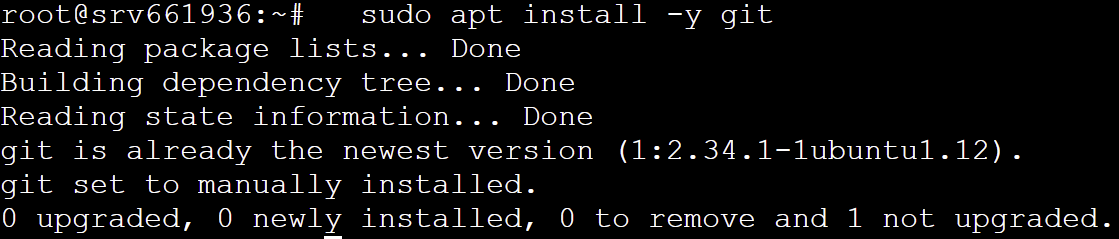
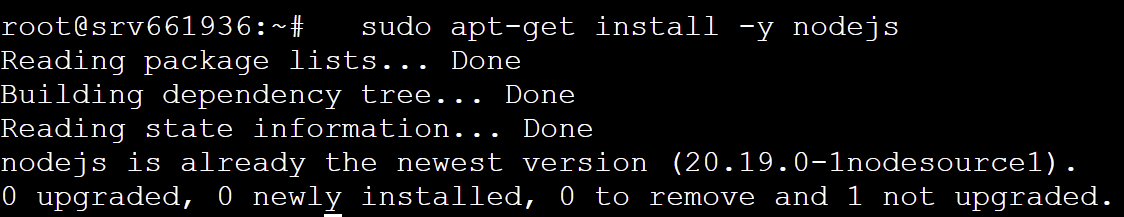
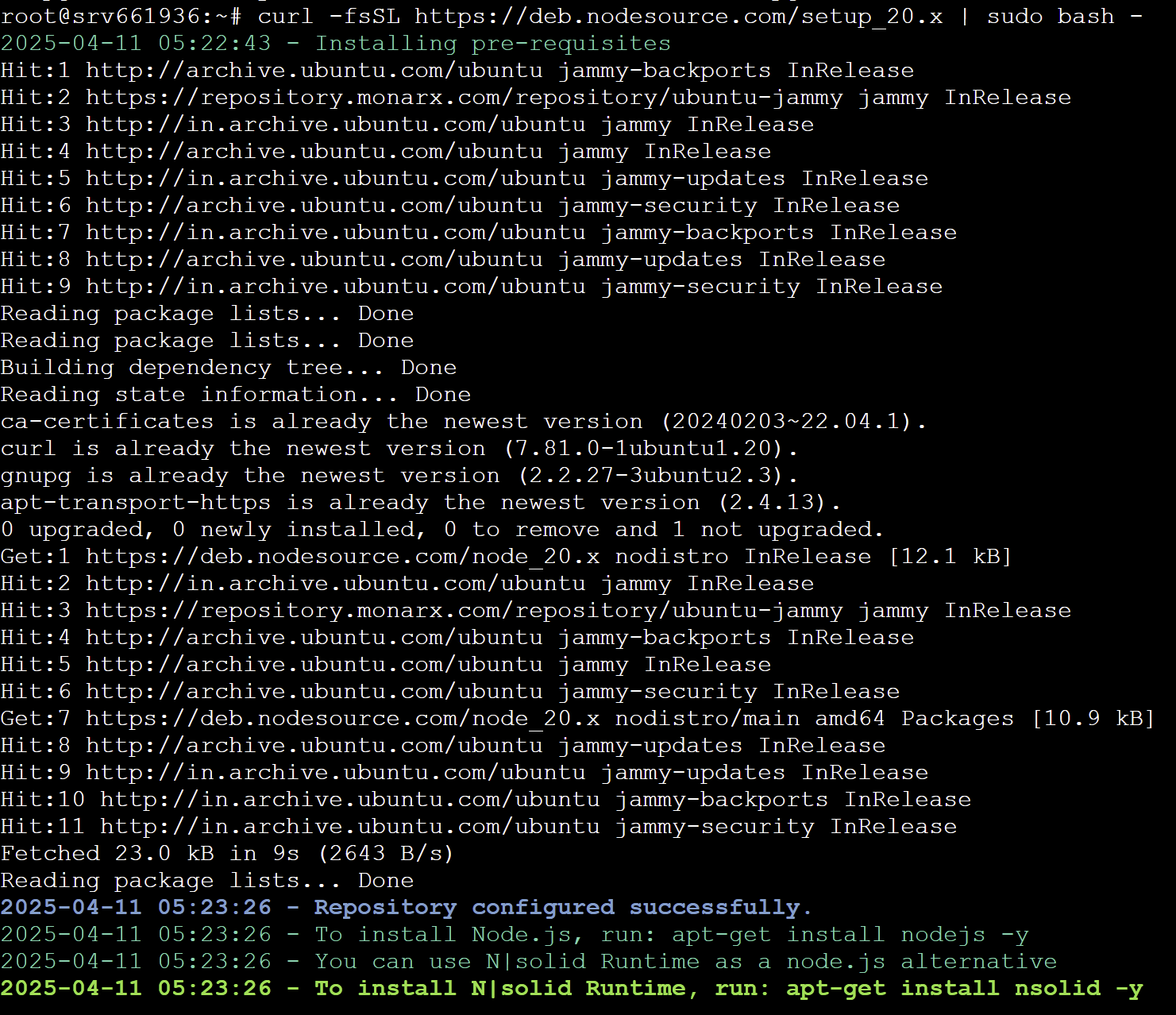
## Cloud Platform & Tools Used

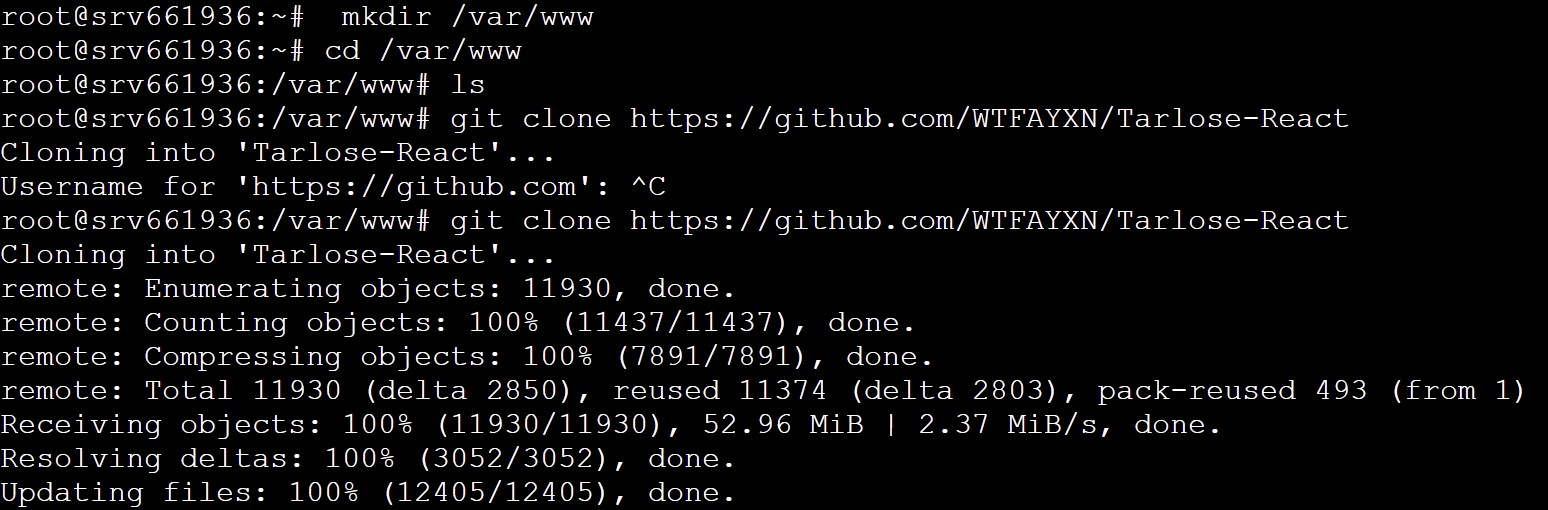
* **Hostinger VPS (Ubuntu 22.04):** Virtual Private Server used as the hosting platform.
* **SSH:** Secure Shell protocol used for remote server access.
* **Node.js & Express:** Backend technologies used to serve APIs and handle server logic.
* **MongoDB/PostgreSQL:** Databases used to manage application data.
* **React:** Frontend JavaScript library used for building the UI.
* **NGINX:** Web server and reverse proxy used to serve frontend files and route API requests.
* **PM2:** Process manager for keeping the backend running continuously.
* **UFW:** Uncomplicated Firewall used to manage and allow network traffic.
* **Certbot (Let’s Encrypt):** Tool used to obtain and configure free SSL certificates for HTTPS.
* **DNS & Domain Configuration:** Linking a domain to the VPS IP address and managing its routing.

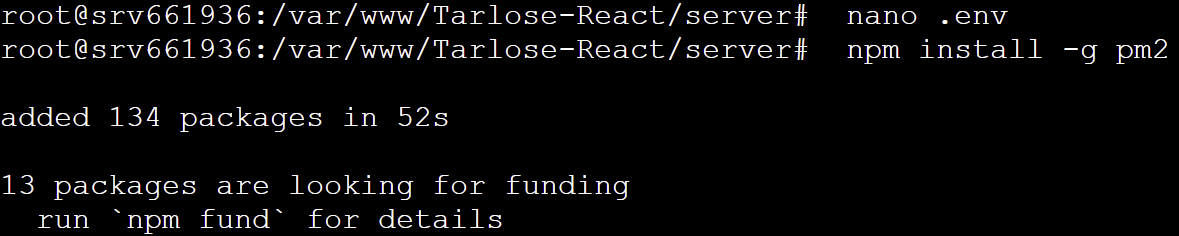
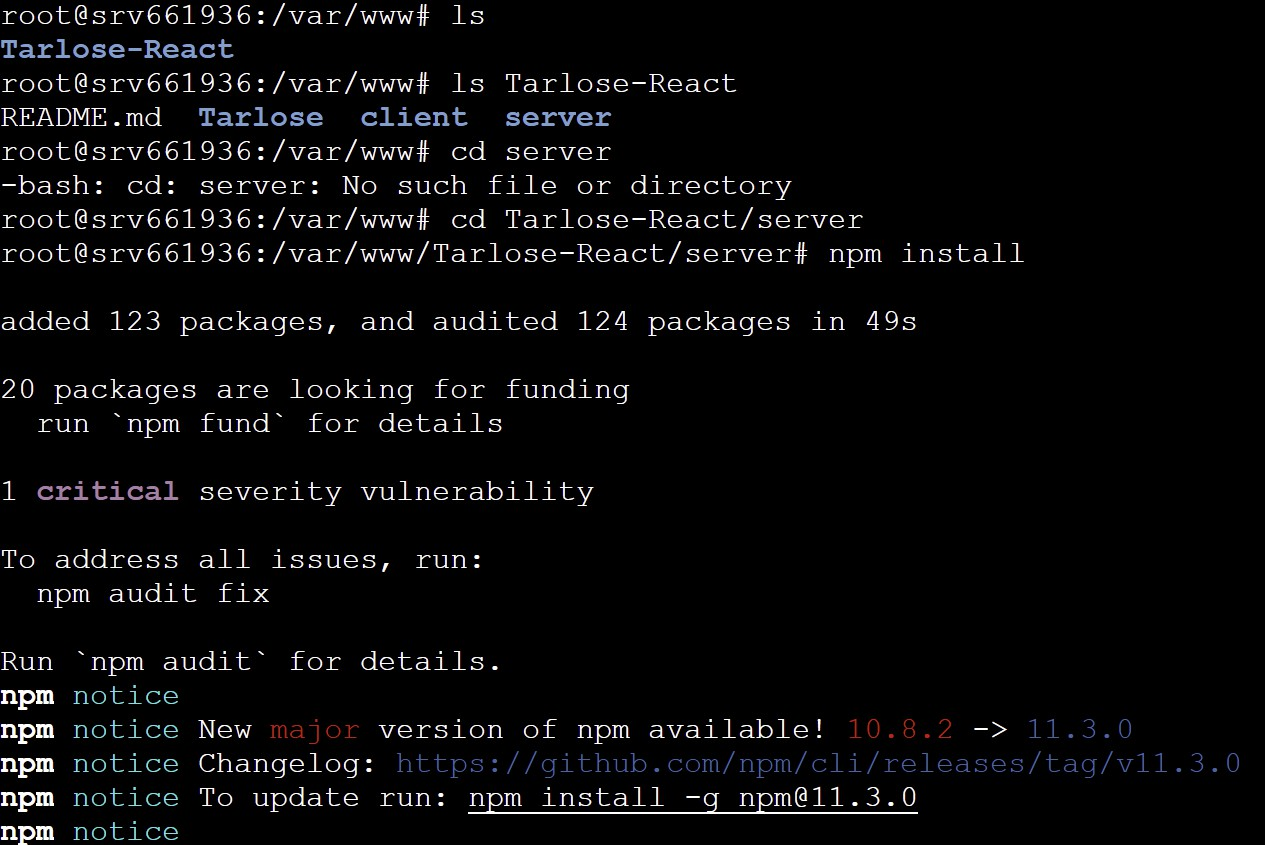
## Deployment Steps

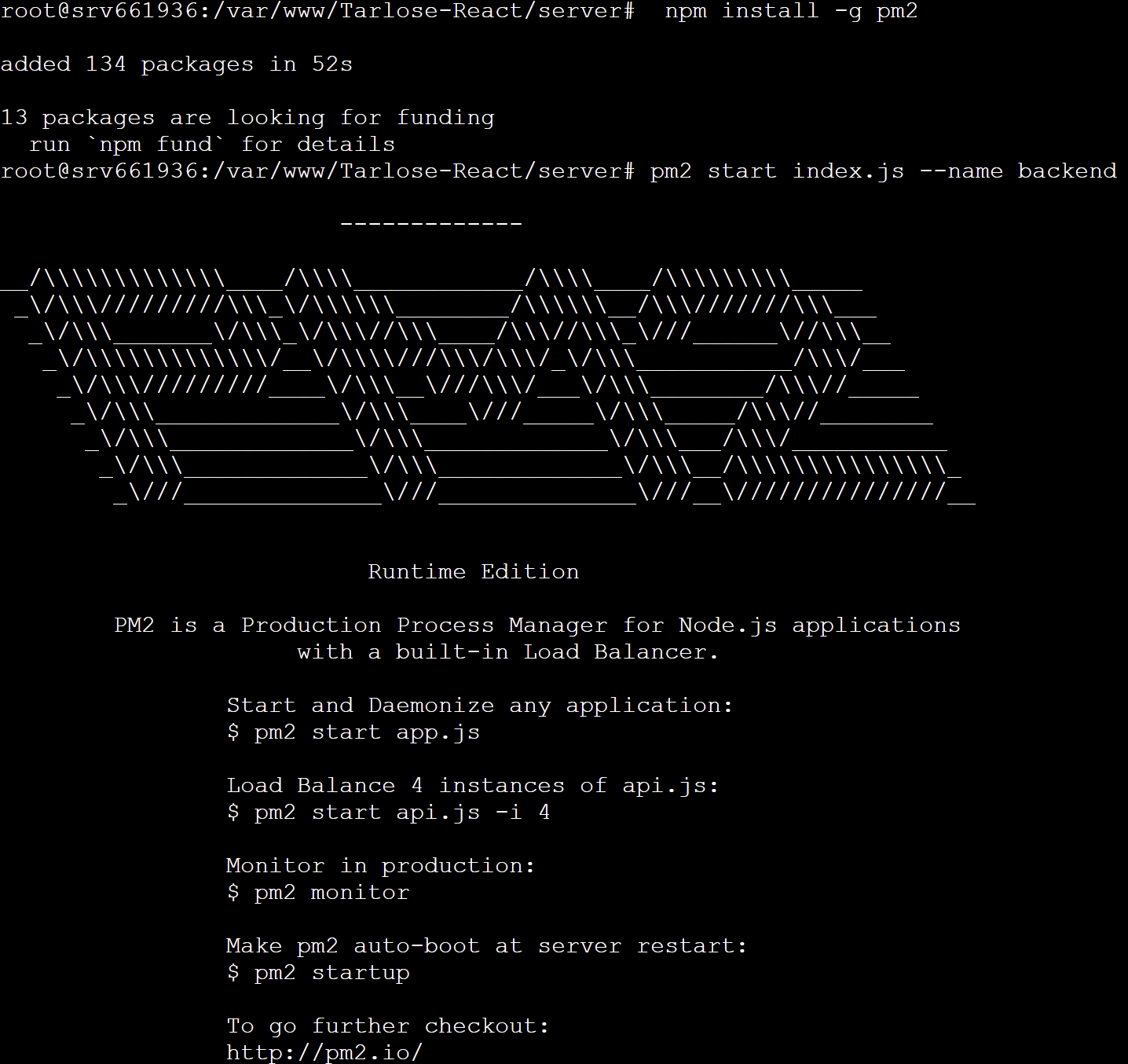
1. **VPS Setup:** Installed required software stack via SSH.
2. **Backend Deployment:** Cloned project, installed dependencies, configured environment, and launched with PM2.
3. **Frontend Hosting:** Built React app and served using NGINX.
4. **Database Connection:** Integrated MongoDB/PostgreSQL for data storage.
5. **Domain & SSL:** Linked domain to VPS, created NGINX server blocks, and added SSL via Certbot.
6. **Firewall Configuration:** Allowed essential ports like 80 (HTTP), 443 (HTTPS), and 4000 (API).

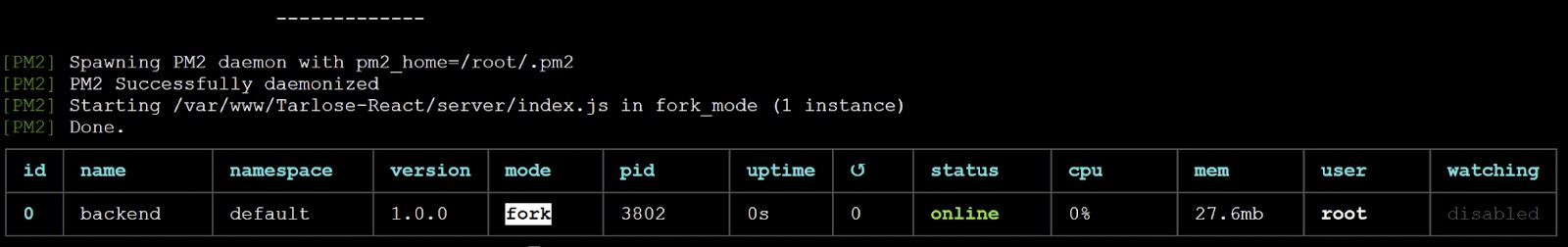
**Implementation:**

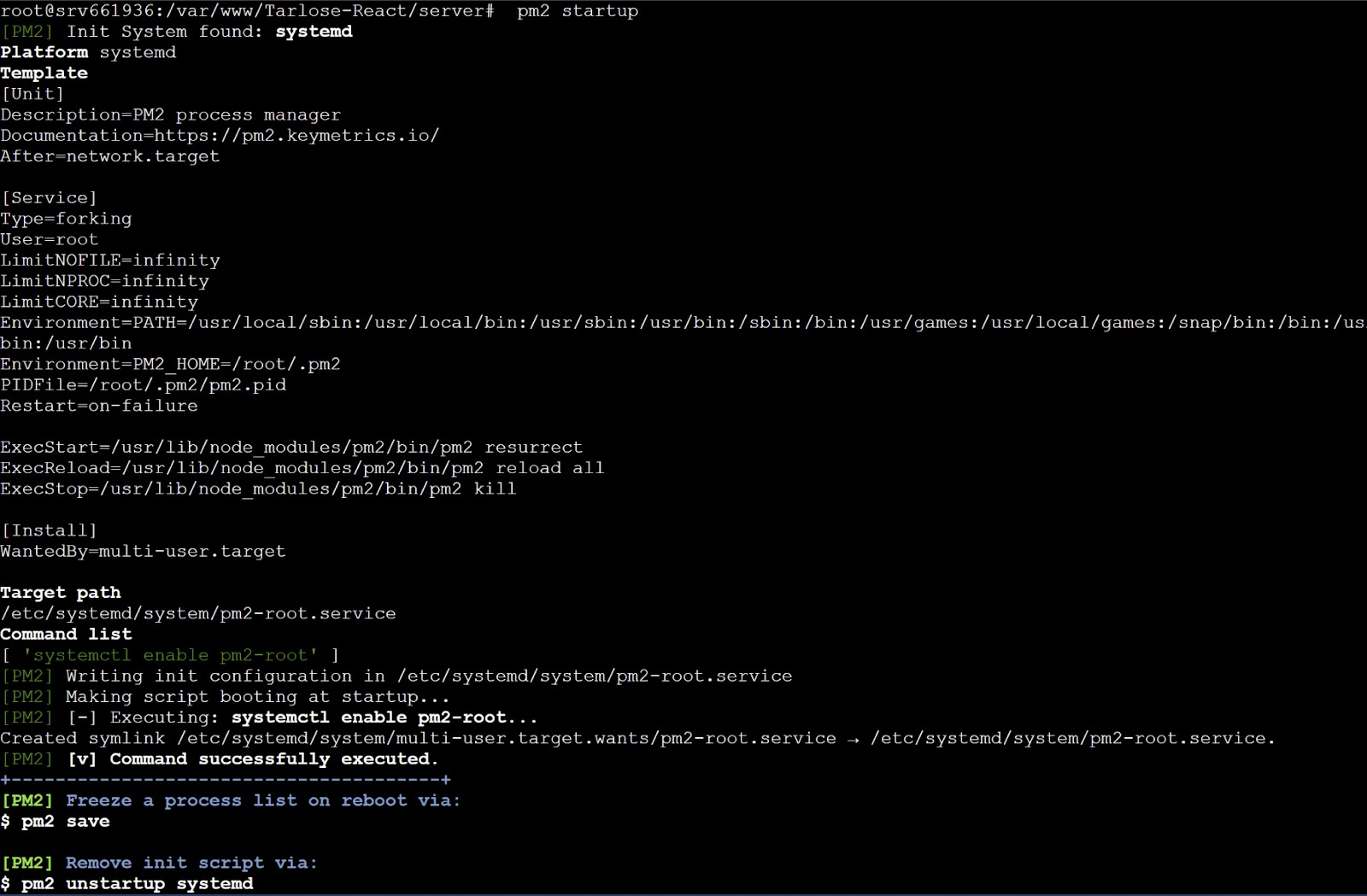


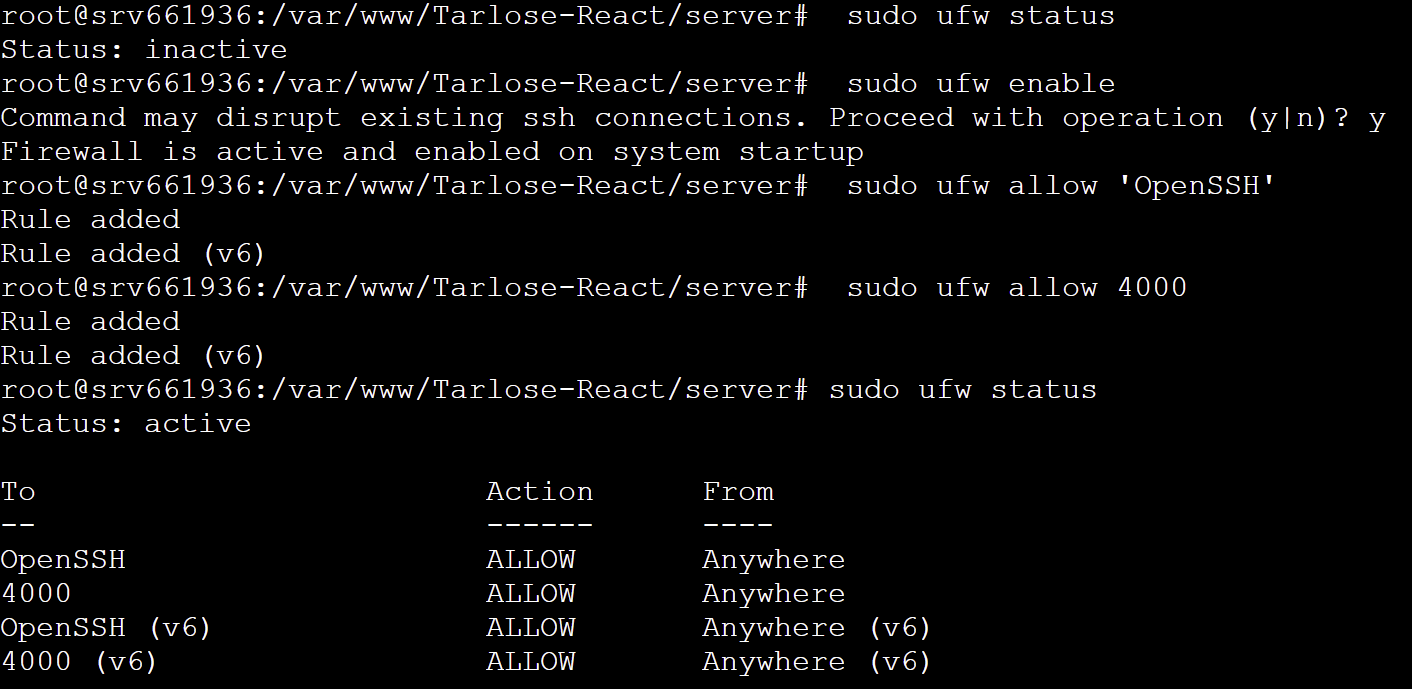
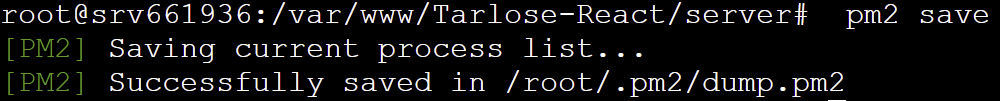


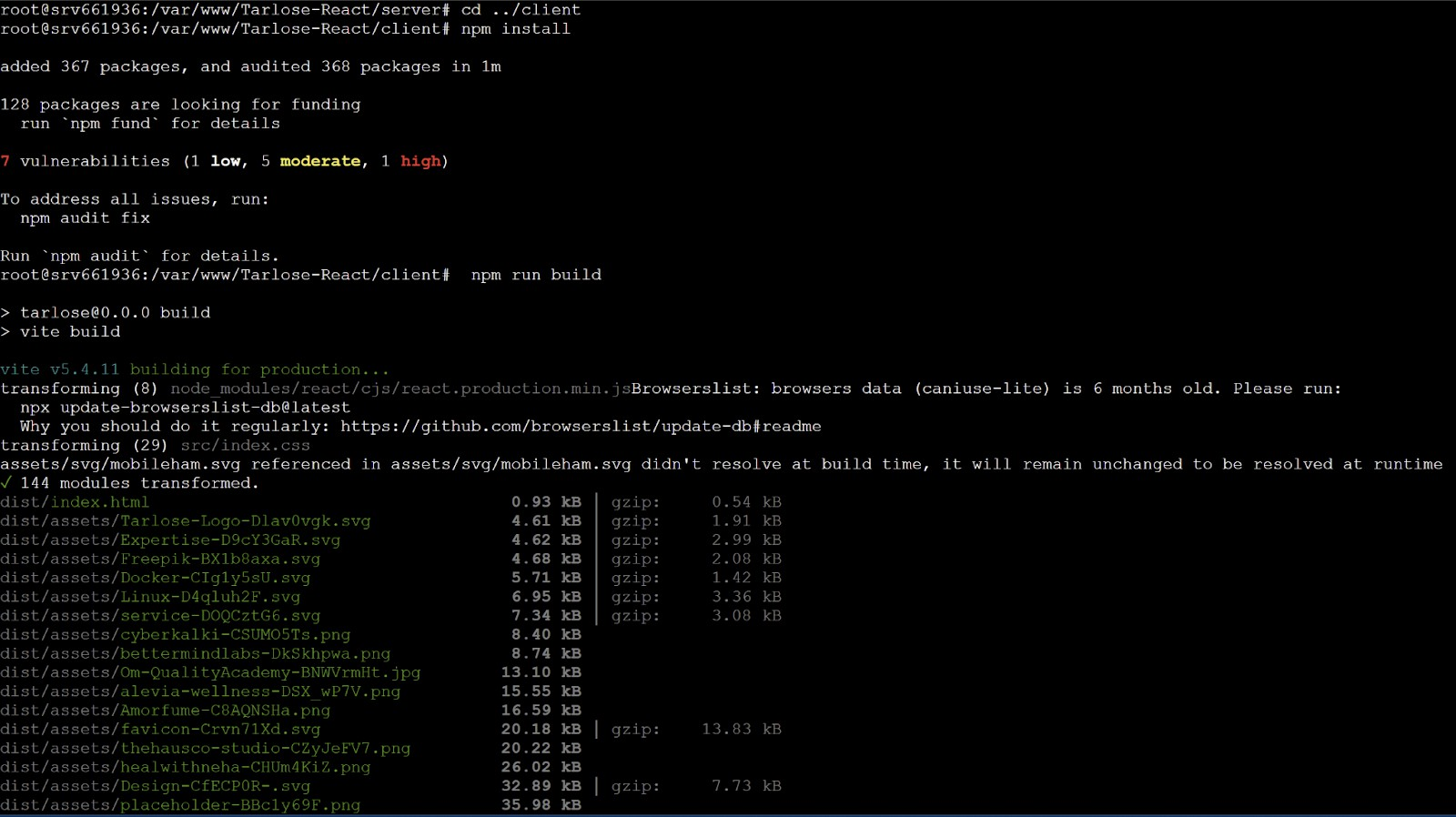


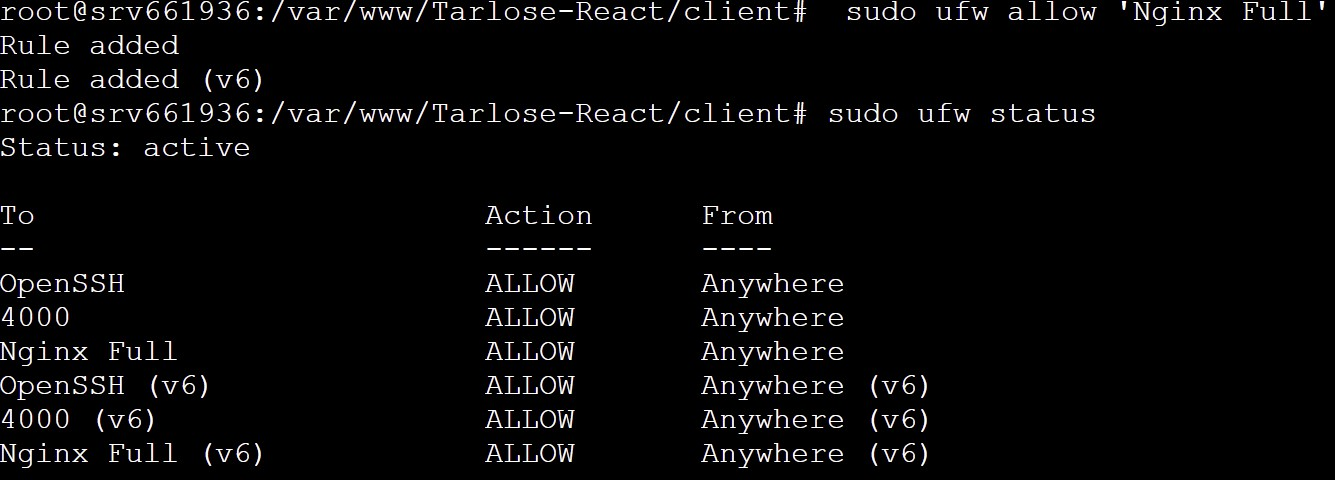
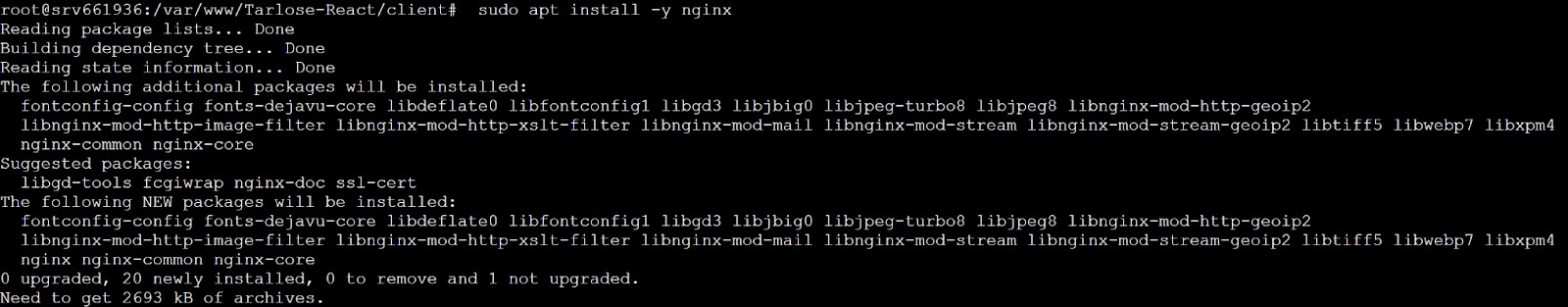


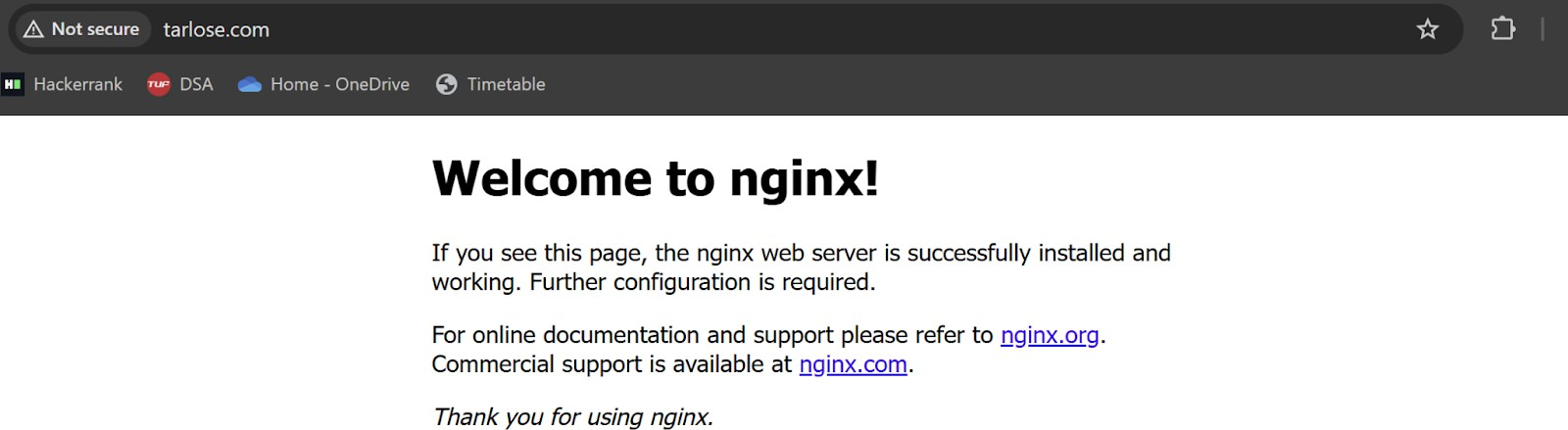


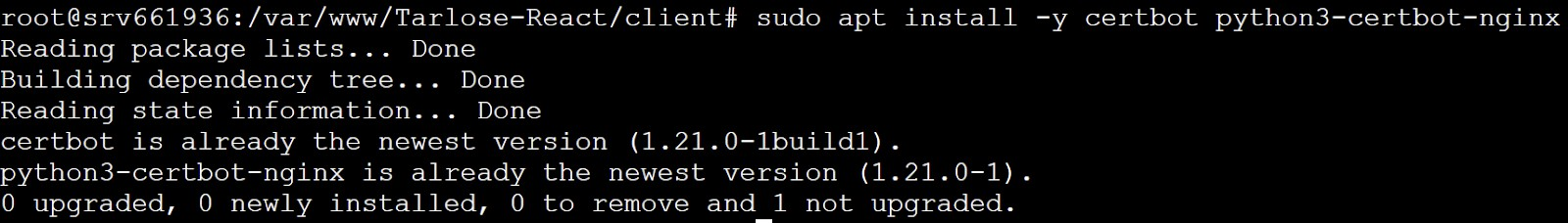
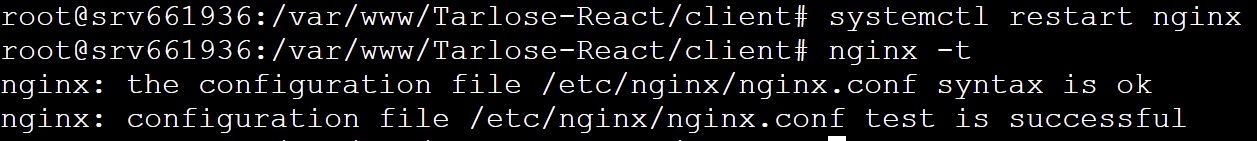
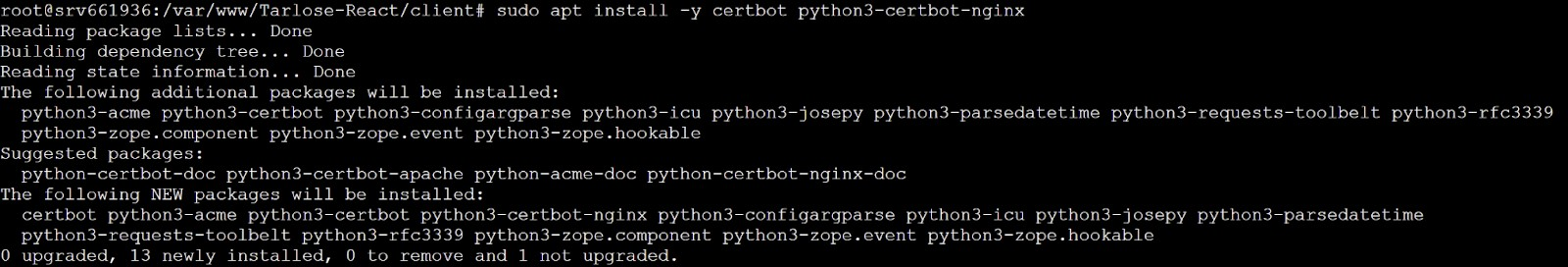


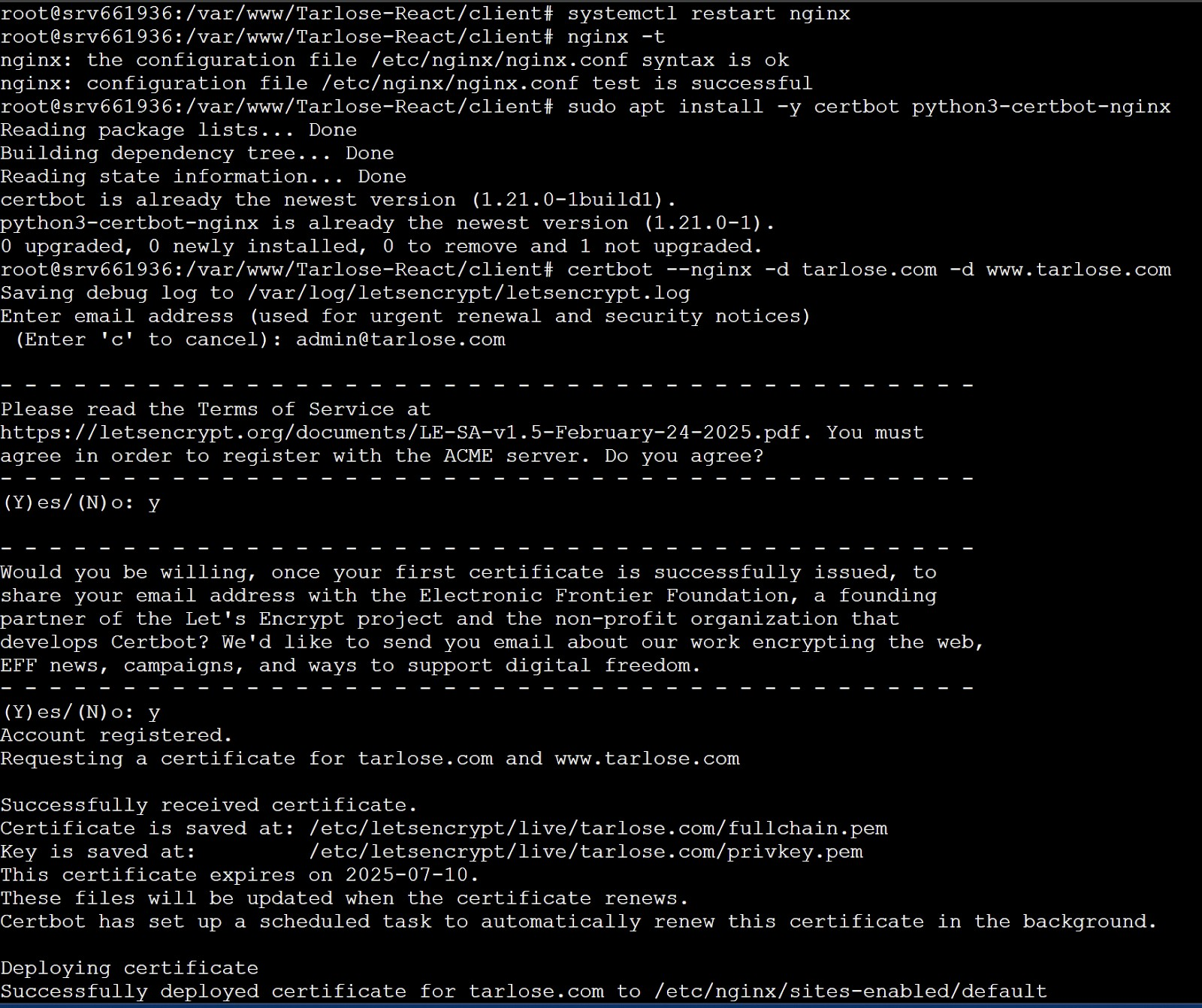


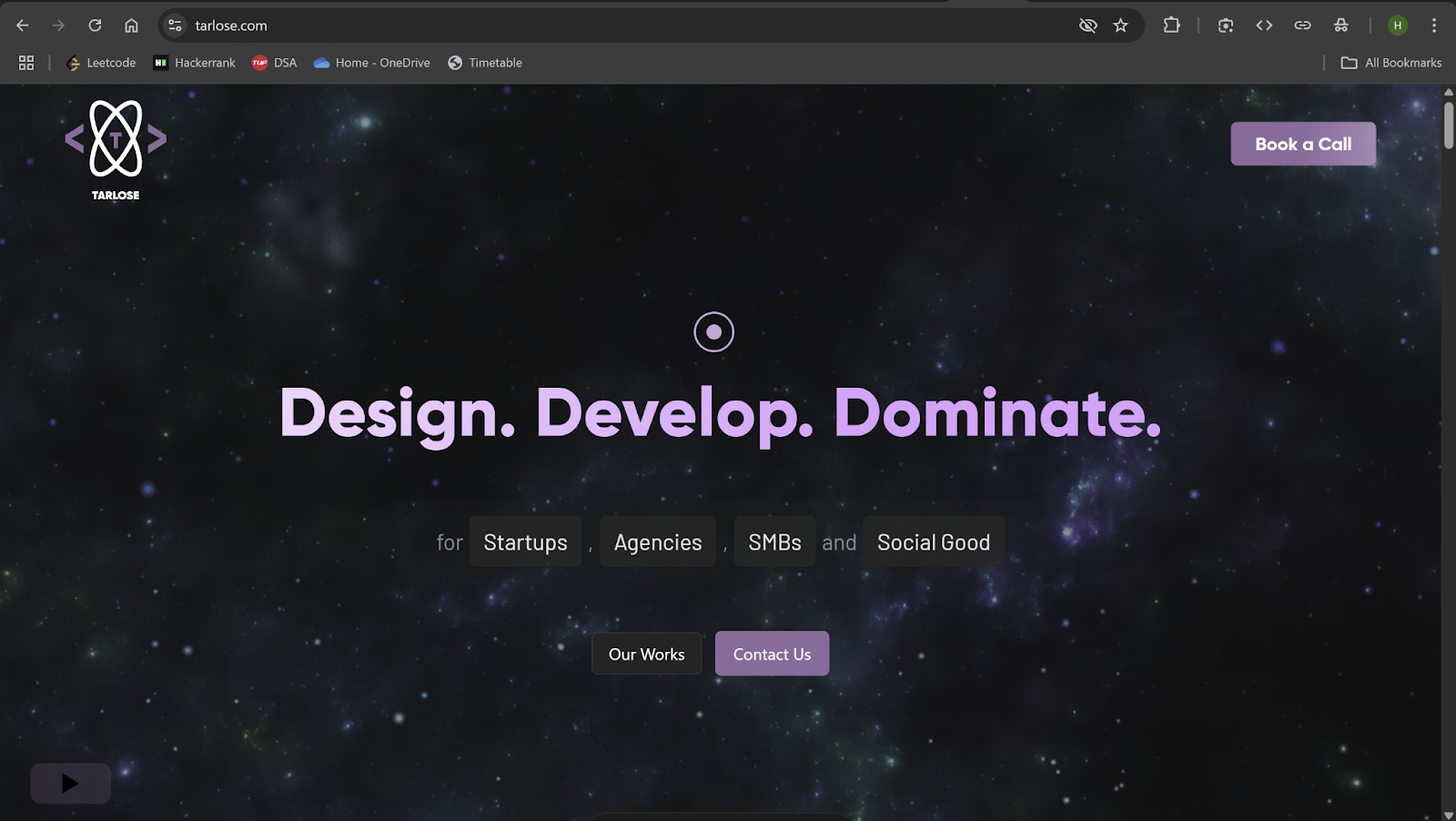


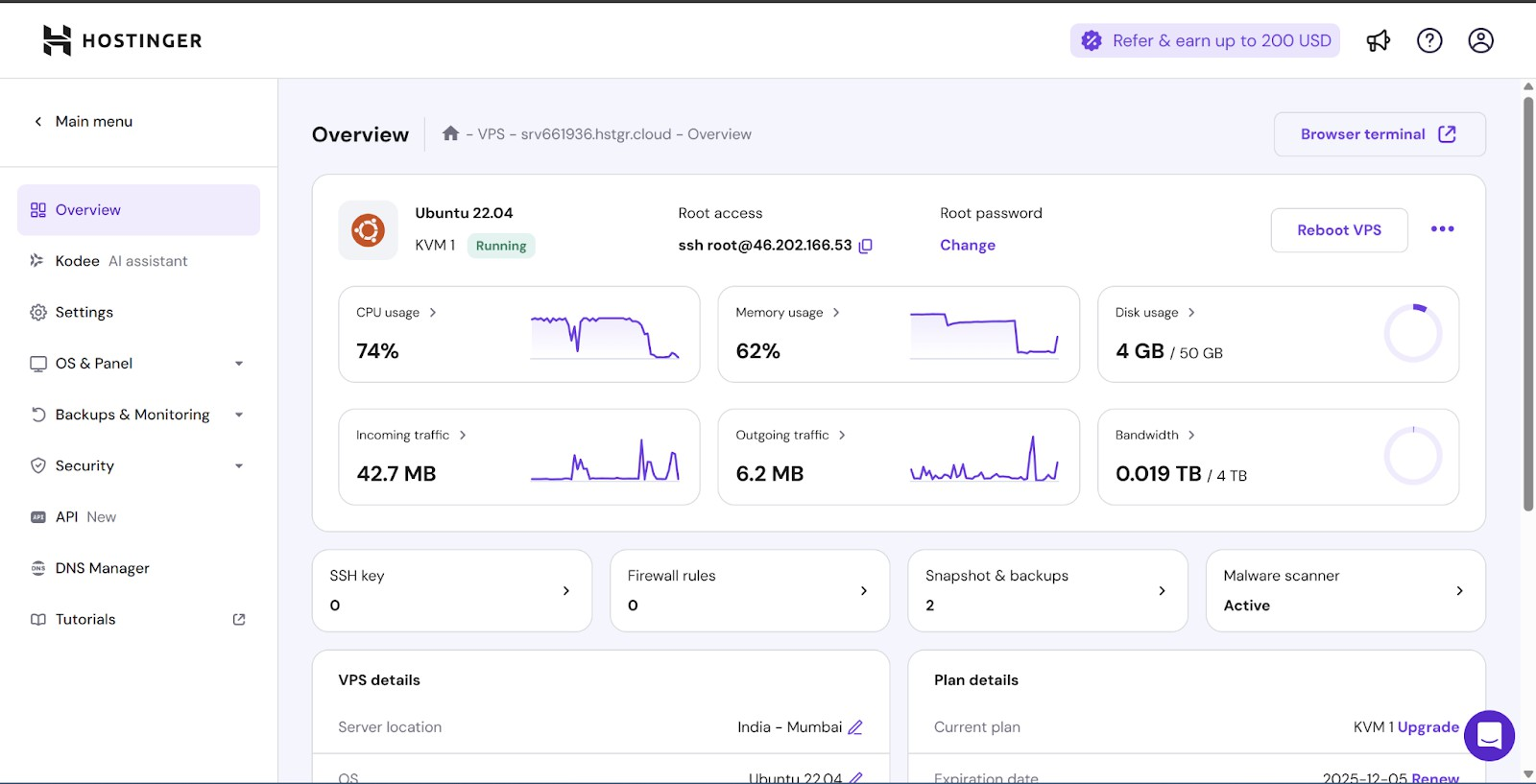












**Conclusion:** Successfully hosted a full-stack web application on Hostinger VPS with custom domain and SSL, demonstrating end-to-end deployment.