--- English ---

Complete Deployment Guide for the Orchestrator on a VPS (Ubuntu 22.04)

This guide provides a comprehensive, step-by-step process for installing and configuring your orchestrator server on a public VPS.

Step 0: Prerequisites

A VPS Server: A VPS running Ubuntu 22.04. You will need its public IP address and root access (or a user with sudo privileges).

A Domain Name (Recommended): While you can use the IP, a domain is more professional. Point your domain (e.g., your-domain.com) or a subdomain (e.g., api.your-domain.com) to your VPS's public IP address via your domain provider's DNS records.

The orchestrator.py file: Have the latest version of the orchestrator code ready on your local computer.

Step 1: Connect and Update Your Server

Connect to your VPS via SSH, replacing your\_user and your\_public\_ip.

ssh your\_user@your\_public\_ip

Once logged in, update the system's package list:

sudo apt update && sudo apt upgrade -y

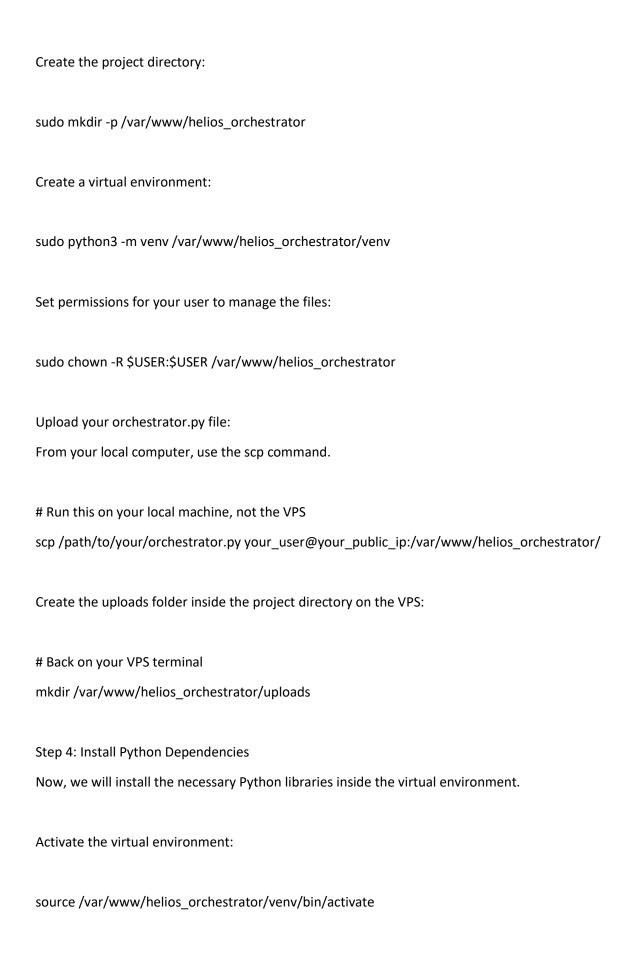
Step 2: Install System Dependencies

Install Python, its package manager pip, the virtual environment tool, and the NGINX web server.

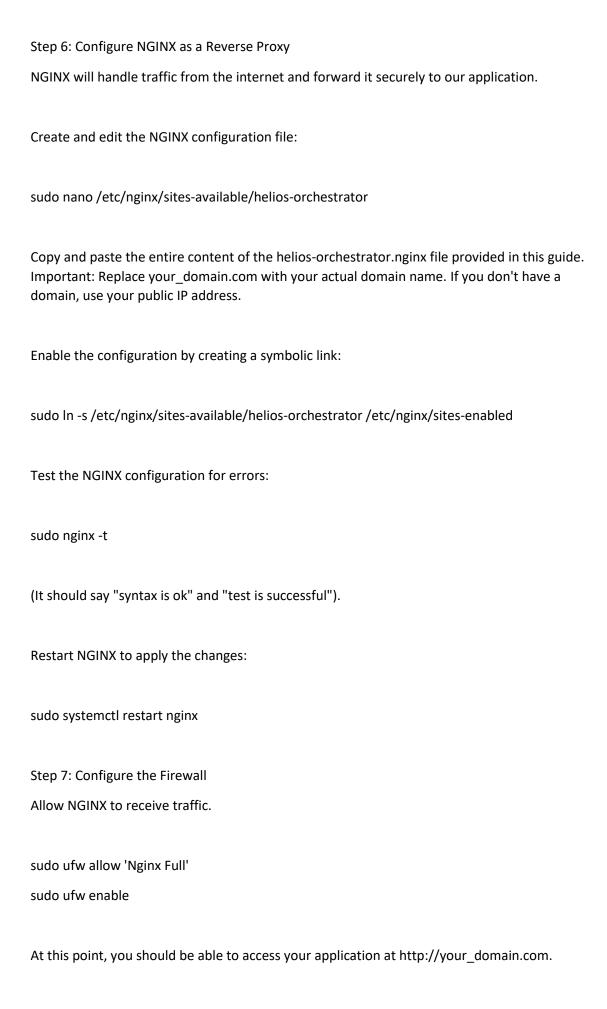
sudo apt install python3 python3-pip python3-venv nginx -y

Step 3: Prepare the Application Environment

We will create a dedicated folder for the project and a Python virtual environment to keep dependencies isolated.



(You will see (venv) at the beginning of your terminal prompt).
Install the libraries:
pip install "fastapi[all]" gunicorn
Deactivate the environment for now:
deactivate
Step 5: Create the systemd Service
This will run our orchestrator as a system service, ensuring it starts on boot and restarts if it fails.
Create and edit the service file:
sudo nano /etc/systemd/system/helios-orchestrator.service
Copy and paste the entire content of the helios-orchestrator.service file provided in this guide. Save (Ctrl+O, Enter) and exit (Ctrl+X).
Start and enable the service:
sudo systemctl start helios-orchestrator
sudo systemctl enable helios-orchestrator
Check its status:
sudo systemctl status helios-orchestrator
(You should see a green "active (running)" message).



Step 8: (Highly Recommended) Secure with HTTPS

This final step encrypts all communication to and from your server.

Install Certbot, the Let's Encrypt client:

sudo apt install certbot python3-certbot-nginx -y

Obtain and install the SSL certificate. Replace your\_domain.com with your domain.

sudo certbot --nginx -d your\_domain.com

Certbot will ask for your email and for you to agree to the terms. It will also ask if you want to redirect HTTP traffic to HTTPS. Choose option 2 (Redirect).

Congratulations! Your orchestrator is now securely deployed and running 24/7 at https://your\_domain.com. Remember to update the ORCHESTRATOR\_PUBLIC\_URL in your worker.py file before distributing it.