**Deploying Python web app с Nginx & Gunicorn на Debian:**

**Подготовка сервера:**

Заходим на сервер:

***ssh root@111.222.333.444***

Обновление & установка обновлений модулей:

***apt-get update***

***apt-get upgrade***

Установка модулей:

***apt-get install -y vim ufw htop git curl wget build-essential make***

Создание нового администратора:

***adduser cunao***

***adduser cunao sudo***

Покидаем сервер:

***exit***

***exit***

Заходим на сервер под новым администратором:

***ssh sunao@111.222.333.444***

Покидаем сервер:

***exit***

***exit***

Создание пар ключей на локальной машине & передача их на сервер:

***ssh-keygen***

***ssh-copy-id*** [***cunao@111.222.333.444***](mailto:cunao@111.222.333.444)

Заходим на сервер:

***ssh sunao@111.222.333.444***

Изменение прав root:

***sudo vim /etc/ssh/sshd\_config***

***\_\_\_***

AllowUsers cunao

PermitRootLogin no

PasswordAuthentication no

***\_\_\_***

Установка CPython:

***wget https://www.python.org/ftp/python3.9.1/Python-3.9.1.tgz***

***tar xvf Python-3.9.1***

***cd Python-3.9.1***

***mkdir ~/.python***

***./configure --enable-optimizations --prefix=/home/cunao/.python***

***make -j8***

***sudo make altinstall***

Сохранение пути к последнему CPython:

***sudo vim ~/ .zshrc***

***\_\_\_***

export PATH=$PATH:/home/cunao/.python/bin

***\_\_\_***

**Развёртывание файлов приложения:**

***cd /home/cunao***

***sudo git clone*** [***https://github.com/cunao-nako/a\_beta***](https://github.com/cunao-nako/a_beta)

***cd a\_beta***

***python3 -m venv a\_betaenv***

***source a\_betaenv/bin/activate***

***pip3 install -r requirements.txt***

***python3 a\_beta.py***

***pip3 install gunicorn***

***gunicorn –bind 0.0.0.0:5000 wsgi:app***

***deactivate***

***nano wsgi.py***

***\_\_\_***

from a\_beta import app

if \_\_name\_\_ == ‘\_\_main\_\_’:

app.run()

***\_\_\_***

**Конфигурация Gunicorn:**

***sudo nano /etc/systemd/system/a\_beta.service***

***\_\_\_***

[Unit]

Description=Gunicorn instance to serve a\_beta

After=network.target

[Service]

User=cunao

Group=www-data

WorkingDirectory=/home/cunao/a\_beta/

Environment="PATH=/home/cunao/a\_beta/a\_betaenv/bin"

ExecStart=/home/cunao/a\_beta/a\_betaenv/bin/gunicorn --workers 3 --bind unix:a\_beta.sock wsgi:app

[Install]

WantedBy=multi-user.target

***\_\_\_***

Разрешаем запуск (# exac: …  - запускает wsgi.py):

***sudo chmod +x /etc/systemd/system/a\_beta.service***

Запускаем Gunicorn & проверям работоспособность:

***sudo systemctl start a\_beta.service***

***sudo systemctl enable a\_beta.service***

***sudo systemctl status a\_beta.service***

**Конфигурация Nginx:**

***sudo nano /etc/nginx/sites-available/default***

***\_\_\_***

server {

listen 80 default\_server;

listen [::]:80 default\_server;

root /var/www/html;

index index.html index.htm index.nginx-debian.html;

server\_name www.xn----7sbbobf2cfkt.xn--p1ai;

location / {

include proxy\_params;

proxy\_pass http://unix:/home/cunao/a\_beta/a\_beta.sock;

proxy\_set\_header X-Forwarded-Host $server\_name;

add\_header Access-Control-Allow-Origin \*;

}

}

***\_\_\_***

Проверям работоспособность & запускаем Nginx:

***sudo nginx -t***

***sudo service nginx restart***

***sudo service nginx status***

Разрешаем подключения через ssh к нашему серверу:

***sudo ufw allow 'OpenSSH'***

Разрешаем подключения Nginx получать запросы из сети:

***sudo ufw allow 'Nginx Full'***

Запускаем & проверяем работу фаервола ufw:

***sudo ufw enable***

***sudo ufw status***

**Безопасность:**

***sudo add-apt-repository ppa:certbot/certbot***

***sudo apt install python-certbot-nginx***

***sudo certbot --nginx -d your\_domain -d www.your\_domain***

***sudo service nginx restart***

**Отладка:**

проверяет журналы ошибок Nginx:

***sudo less /var/log/nginx/error.log***

проверяет журналы доступа Nginx:

***sudo less /var/log/nginx/access.log***

проверяет журналы процессов Nginx:

***sudo journalctl -u nginx***

проверяет журналы Gunicorn:

***sudo journalctl -u myproject***

**Изменение в файлах приложения:**

\*изменение в файлах\*

***sudo rm /etc/systemd/system/a\_beta.service***

***systemctl stop a\_beta.service***

***systemctl daemon-reload***

***sudo nano /etc/systemd/system/a\_beta.service***

***\_\_\_***

[Unit]

Description=Gunicorn instance to serve a\_beta

After=network.target

[Service]

User=cunao

Group=www-data

WorkingDirectory=/home/cunao/a\_beta/

Environment="PATH=/home/cunao/a\_beta/a\_betaenv/bin"

ExecStart=/home/cunao/a\_beta/a\_betaenv/bin/gunicorn --workers 3 --bind unix:a\_beta.sock wsgi:app

[Install]

WantedBy=multi-user.target

***\_\_\_***

***sudo chmod +x /etc/systemd/system/a\_beta.service***

***systemctl daemon-reload***

***sudo systemctl start a\_beta.service***

***sudo systemctl enable a\_beta.service***

***sudo systemctl status a\_beta.service***

***sudo service nginx restart***

**Источники:**

Обслуживание flask web app с Gunicorn и Nginx на Ubuntu 18.04:

<https://www.digitalocean.com/community/tutorials/how-to-serve-flask-applications-with-gunicorn-and-nginx-on-ubuntu-18-04-ru>

ssh ключи:

<https://www.youtube.com/watch?v=IVHv3eVQa14>

поднятие & подготовка Debian сервера для Python web app:

<https://youtu.be/FLiKTJqyyvs>

листинг команд из видео про поднятие & подготовка Debian сервера для Python web app:

https://github.com/alexey-goloburdin/debian-set-up-for-django/tree/4375730953b96b7900b9d0ea93683dd0cc9a163a