

Инструкция по развертыванию ПО “ПФП” на сервере

Содержание

Состав дистрибутива ПО “ПФП”	3
Описание параметров конфигурационных файлов.....	4
Предпочтительный порядок развертывания.....	5
Детальное описание развертывания ПО	6

Состав дистрибутива ПО “ПФП”

- Конфигурационные файлы системы
 - Config.yml – файл конфигурации GO
 - Pfp.netstl.ru
- Файл базы данных
- Файлы приложения
 - Go.go – файл основного функционала приложения
 - Favicon.ico – иконка приложения
 - Index.html – стартовая страница приложения
 - Index.css – файл стиля
 - Index.js – файл с реализацией интерфейса
 - Два файла шрифтов

Описание параметров конфигурационных файлов

В данном разделе описываются конфигурационные файлы дистрибутива и их параметры.

/var/www/FQDN сайта/go/config.yml

- Db
 - Host—Адрес базы данных
 - Dbname – Имя базы данных, к которой подключаемся
 - User - Пользователь, под которым проходит подключение к БД
 - Pw - Пароль для подключения к БД
 - Port—Порт, на котором находится база данных
 - Sslmode – Использовать ли SSL
- App
 - Host – Адрес приложения
 - Port – Порт, на котором находится приложение

/etc/nginx/sites-available/FQDN сайта

server_name - FQDN сайта

listen – порт сайта (80 для HTTP, 443 для HTTPS)

set \$root_path – каталог размещения html страниц сайта

Предпочтительный порядок развертывания

Предпочтительный порядок развёртывания ПО ПФП:

1. Установка сервера Ubuntu
2. Установка Nginx
3. Установка и настройка PostgreSQL
4. Копирование файлов дистрибутива и настройка конфигураций

Детальное описание развертывания ПО

Требования к серверу.

Установка может выполняться на физическую или виртуальную машину с минимальными требованиями:

Оперативная память	4 Гб
Процессор	2 ядра
Жесткий диск	20 ГБ
Сетевые интерфейсы	2

Установка сервера Ubuntu

1. Скачать дистрибутив сервера Ubuntu по ссылке:
<https://releases.ubuntu.com/22.04/ubuntu-22.04.2-live-server-amd64.iso>
2. Смонтировать скачанный iso файл для установки.
3. Запустить установку, выбрать пункт Try or install Ubuntu Server

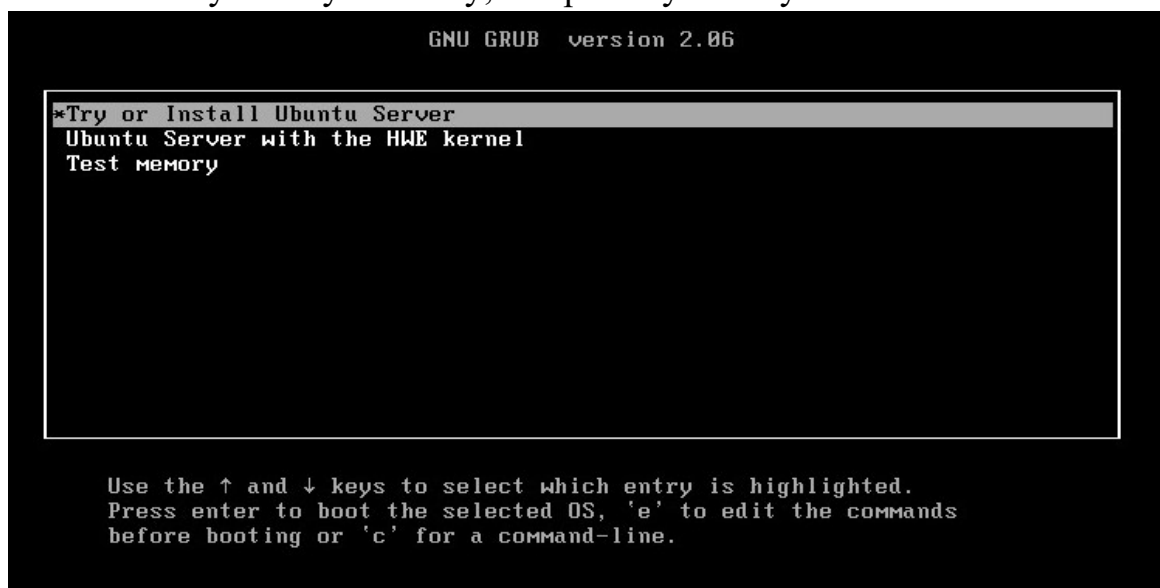


Рис. 1 Загрузочное меню Ubuntu

4. Далее по ходу установки выбираем язык, переключатель ввода.
5. В меню выбора типа установки выбираем стандартную

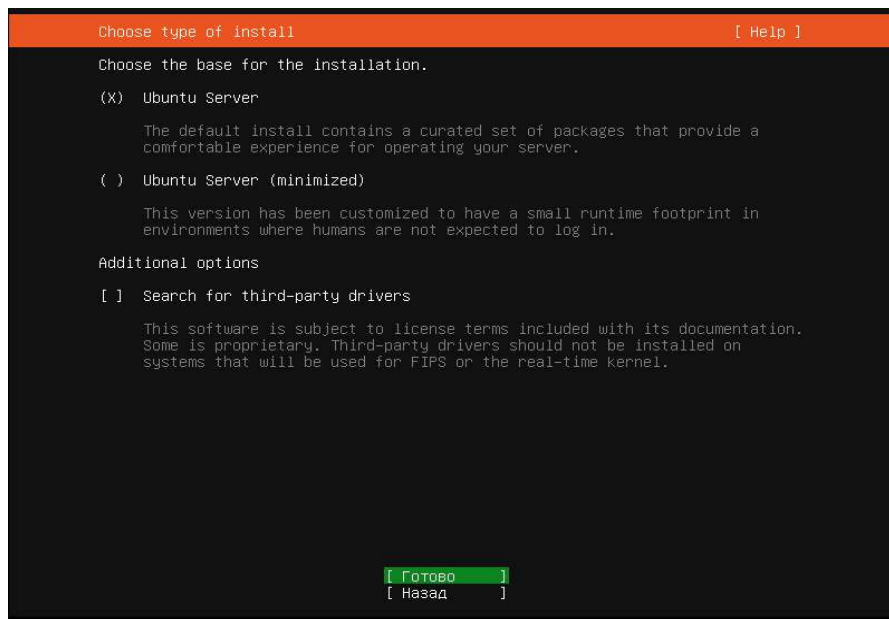


Рис. 2 Выбор типа установки

6. Создаем сетевые адреса, первый для доступа к серверу из сети Интернет

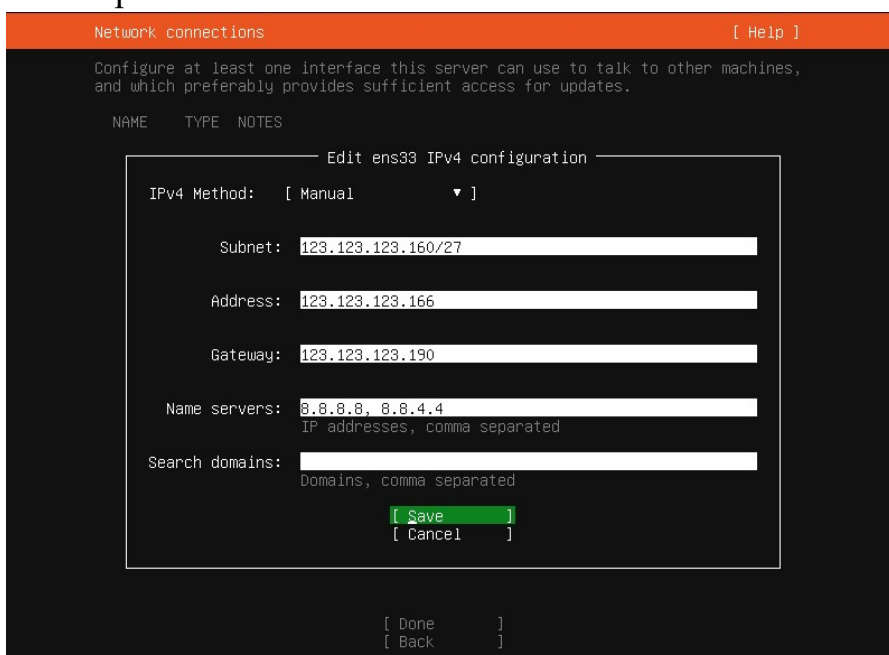


Рис. 3 Настройка адреса для выхода в интернет

7. Второй адрес для DMZ сети

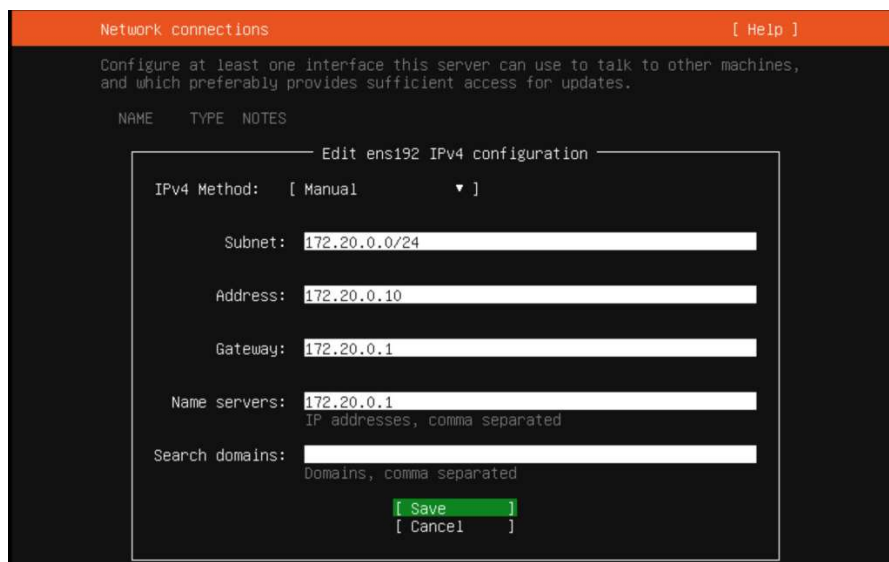


Рис. 4 Настройка адреса DMZ сети

8. Настройка прокси-сервера

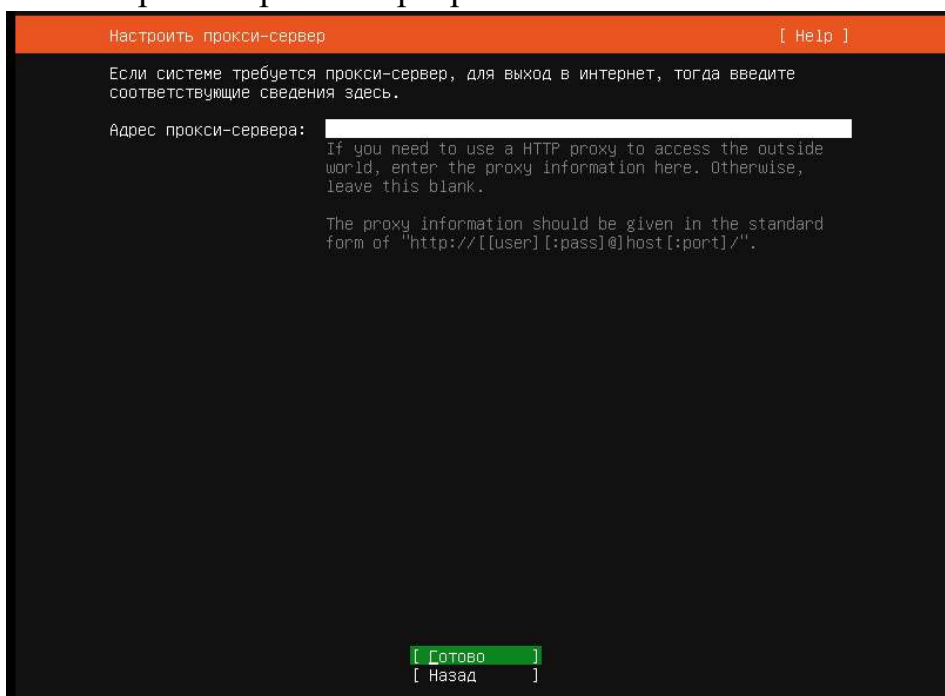


Рис. 5 Окно настройки прокси-сервера

9. Настройка сервера обновлений

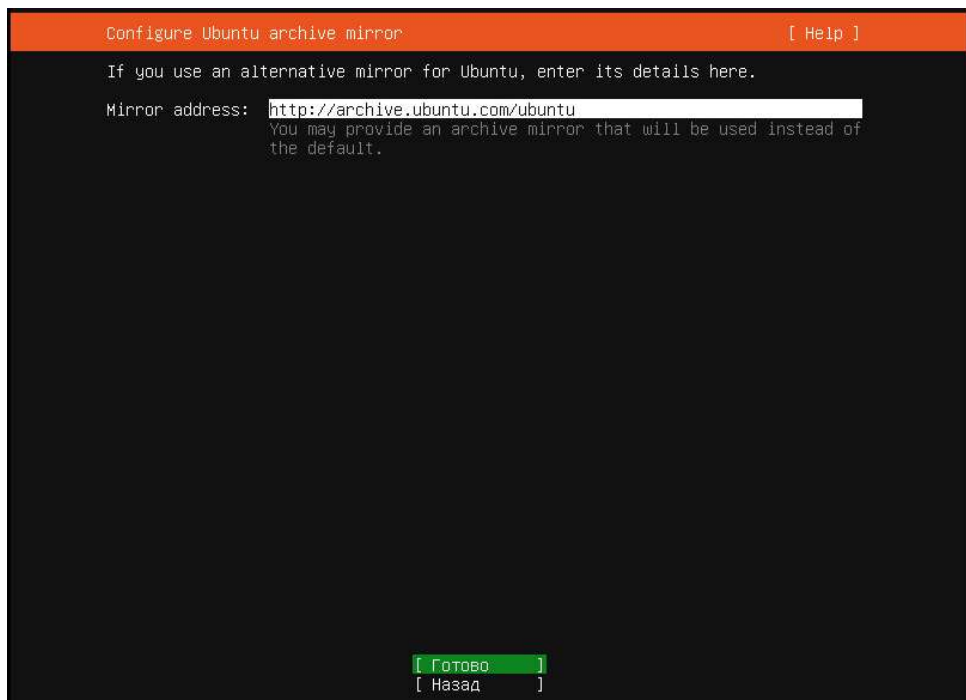


Рис. 6 Настройка сервера обновлений

10. Разметка диска. В появившемся окне выбираем стандартные значения и нажимаем готово.

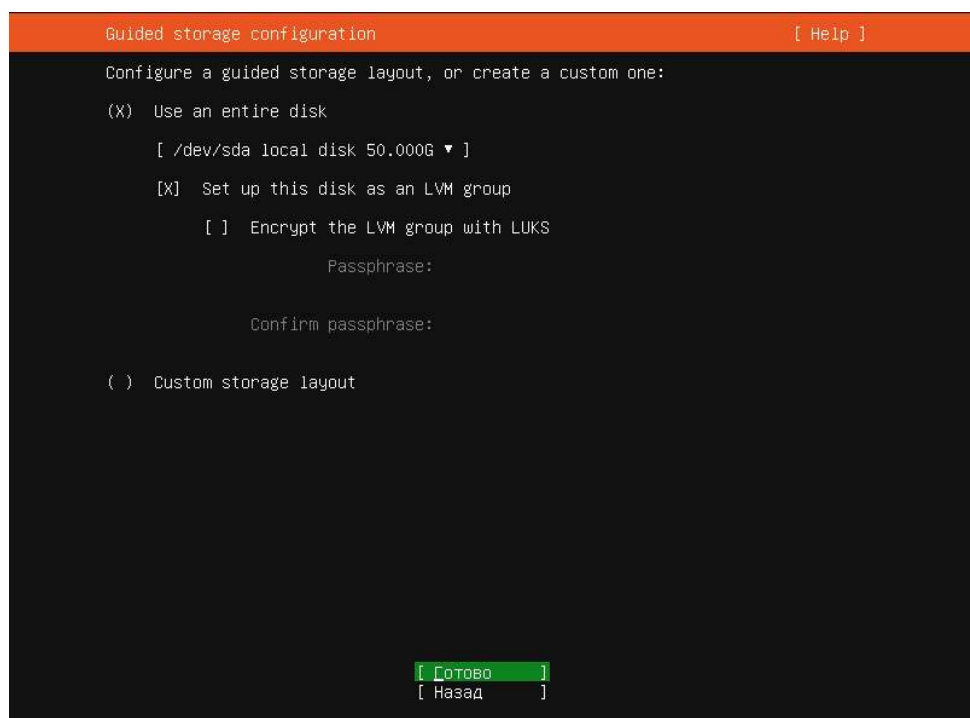


Рис. 7 Окно разметки диска

11. Настройка учетной записи. Далее необходимо добавить хотя бы одну учетную запись для взаимодействия с веб-сервером

Profile setup [Help]

Enter the username and password you will use to log in to the system. You can configure SSH access on the next screen but a password is still needed for sudo.

Your name: adminname

Your server's name: your-servername
The name it uses when it talks to other computers.

Pick a username: adminname

Choose a password: *****

Confirm your password: *****

[Done]

Рис. 8 Окно создания учетной записи

12.SSH доступ. Для возможности использования SSH в появившемся окне выбираем пункт install

SSH Setup [Help]

You can choose to install the OpenSSH server package to enable secure remote access to your server.

[X] Install OpenSSH server

Import SSH identity: [Нет ▼]
You can import your SSH keys from GitHub or Launchpad.

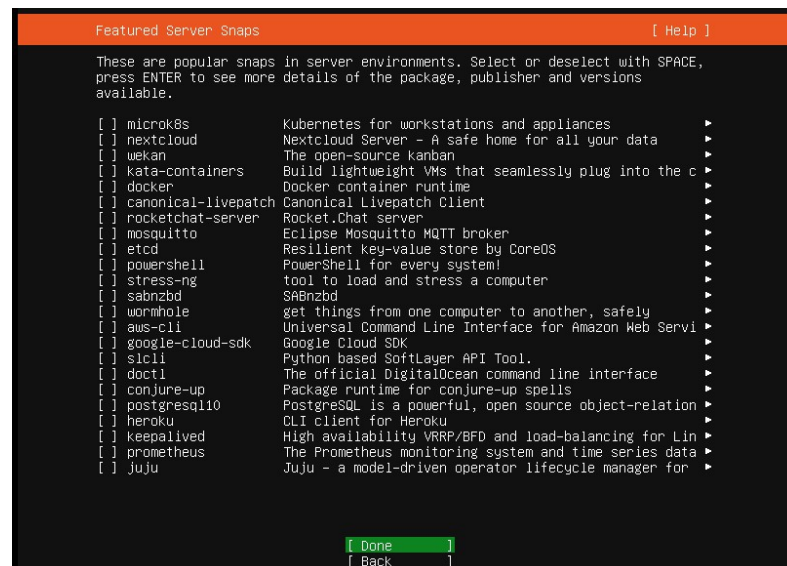
Import Username:

[X] Allow password authentication over SSH

[Готово]
[Назад]

Рис. 9 Окно установкиssh

Установку дополнительных пакетов пока можно пропустить



13. По окончании установки размонтируем установочный iso файл и выберем пункт Reboot Now

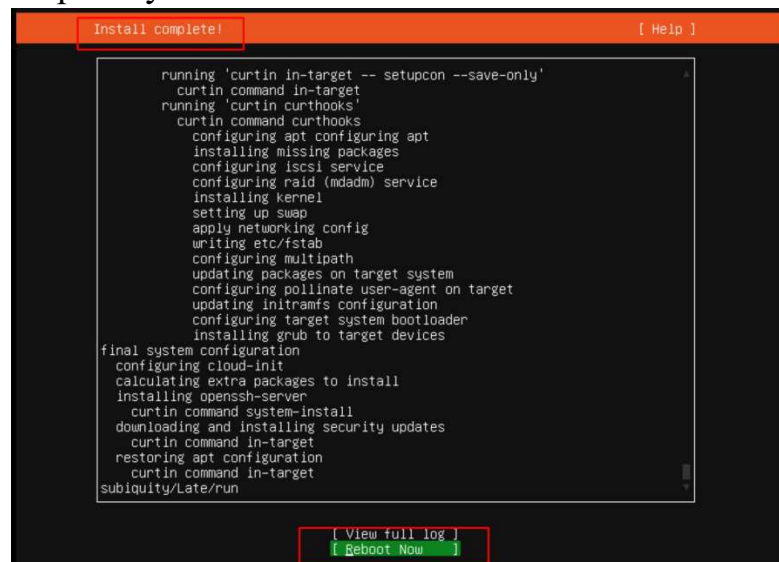


Рис. 10 Окно с сообщением об успешной установке

14. После перезагрузки разрешить сетевой трафик между сетевыми интерфейсами:

```
$ sudo sysctl -w net.ipv4.ip_forward=1
```

```
user1@host16:~$ sudo sysctl -w net.ipv4.ip_forward=1
```

Обновление пакетов и установка NGINX

1. В командной строке выполнить обновление пакетов командой
\$ sudo apt update

```
user1@host16:~$ sudo apt update
Hit:1 http://ru.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://ru.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://ru.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://ru.archive.ubuntu.com/ubuntu jammy-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
18 packages can be upgraded. Run 'apt list --upgradable' to see them.
user1@host16:~$
```

Рис. 11 Результат выполнения команды

2. Выполнить установку NGINX командой:
\$ sudo apt install nginx

```
[ 329.039886] blk_update_request: I/O error, dev fd0, sector 0 op 0x0:(READ) flags 0x0 phys_seg 1 p
rio class 0
[ 329.042141] Buffer I/O error on dev fd0, logical block 0, async page read
Setting up libjbig0:amd64 (2.1-3.1ubuntu0.22.04.1) ...
Setting up libnginx-mod-http-xslt-filter (1.18.0-6ubuntu14.3) ...
Setting up fonts-dejavu-core (2.37-2build1) ...
Setting up libjpeg-turbo8:amd64 (2.1.2-0ubuntu1) ...
Setting up libwebp7:amd64 (1.2.2-2) ...
Setting up libnginx-mod-http-geoip2 (1.18.0-6ubuntu14.3) ...
Setting up libjpeg8:amd64 (8c-2ubuntu10) ...
Setting up libnginx-mod-mail (1.18.0-6ubuntu14.3) ...
Setting up fontconfig-config (2.13.1-4.2ubuntu5) ...
Setting up libnginx-mod-stream (1.18.0-6ubuntu14.3) ...
Setting up libtiff5:amd64 (4.3.0-6ubuntu0.3) ...
Setting up libfontconfig1:amd64 (2.13.1-4.2ubuntu5) ...
Setting up libnginx-mod-stream-geoip2 (1.18.0-6ubuntu14.3) ...
Setting up libgd3:amd64 (2.3.0-2ubuntu2) ...
Setting up libnginx-mod-http-image-filter (1.18.0-6ubuntu14.3) ...
Setting up nginx-core (1.18.0-6ubuntu14.3) ...
* Upgrading binary nginx
Setting up nginx (1.18.0-6ubuntu14.3) ... [ OK ]
Processing triggers for ufw (0.36.1-4build1) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
igor@nets1-dev:~$
```

Рис. 12 Результат выполнения команды

3. Необходимо также провести настройку брандмауэра, выполнив команды:
\$ sudo ufw allow 'Nginx Full'
\$ sudo ufw allow 'OpenSSH'
\$ sudo ufw enable

```
igor@nets1-dev:~$ sudo ufw allow 'Nginx Full'
Rules updated
Rules updated (v6)
igor@nets1-dev:~$
igor@nets1-dev:~$ sudo ufw allow OpenSSH
Rules updated
Rules updated (v6)
igor@nets1-dev:~$
```

Рис. 13 Настройка брандмауэра

4. Выполнить проверку установки Nginx командой:

\$ systemctl status nginx

```
• nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-03-01 15:12:47 UTC; 7min ago
     Docs: man:nginx(8)
  Process: 3926 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=ex
 Process: 3927 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, sta
 Main PID: 4021 (nginx)
    Tasks: 3 (limit: 4573)
   Memory: 4.6M
      CPU: 45ms
   CGroup: /system.slice/nginx.service
           └─4021 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─4024 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
               └─4025 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""

Mar 01 15:12:47 nets1-dev systemd[1]: Starting A high performance web server and a reverse pro
Mar 01 15:12:47 nets1-dev systemd[1]: Started A high performance web server and a reverse prox
~
igor@nets1-dev:~$
```

Рис. 14 Проверка Nginx

Установка и подключение PostgreSQL

1. В командной строке Ubuntu необходимо выполнить команду:
\$ sudo apt install postgresql postgresql-contrib

```
user1@host16:~$ sudo apt install postgresql postgresql-contrib
```

Рис.15 Установка PostgreSQL

2. Далее необходимо получить доступ к командной строке PostgreSQL выполнив команду:

```
$ sudo -u postgres psql
```

```
josepes@ubuntu-pf:~$ sudo apt install postgresql postgresql-contrib
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
Уже установлен пакет postgresql самой новой версии (12+214ubuntu0.1).
Уже установлен пакет postgresql-contrib самой новой версии (12+214ubuntu0.1).
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 26 пак
бновлено.
josepes@ubuntu-pf:~$ sudo -u postgres psql
psql (12.14 (Ubuntu 12.14-0ubuntu0.20.04.1))
Type "help" for help.

postgres=#
```

Рис. 16 Получение доступа к командной строке PostgreSQL

3. Попад в командную строку PostgreSQL создадим новую базу данных воспользовавшись командой:

```
# create database желаемое имя создаваемой базы данных
```

```
postgres=# create database pfp;
CREATE DATABASE
postgres=#
```

Рис. 17 Создание базы данных

4. Для добавления нового пользователя нужно воспользоваться командой:
create user 'желаемое имя создаваемого пользователя' with encrypted password 'желаемый пароль'

```
postgres=# create user pfp with encrypted password '12345678';  
CREATE ROLE  
postgres=#
```

Рис. 18 Добавление нового пользователя

5. Далее необходимо добавить созданному пользователю права на работу с созданной базой данных. Для этого выполним команду:

grant all privileges on database 'имя созданной базы' to 'имя созданного пользователя'

#grant all privileges on database pfp to pfp;

```
postgres=# grant all privileges on database "pfp" to pfp;  
GRANT  
postgres=#
```

Копирование файлов дистрибутива и настройка конфигураций

1. Скопировать папки html и go в каталог /var/www/FQDN сайта/.
2. В каталог /etc/nginx/sites-available/ скопировать файл конфигурации из дистрибутива
3. Для активации веб-сервера необходимо создать символическую ссылку на файл конфигурации, используя команды:
\$ cd /etc/nginx/sites-enabled
\$ sudo ln -s /etc/nginx/sites-available/FQDN сайта /etc/nginx/sites-enabled/ FQDN сайта
4. Установить bundle сертификатов и private key сайта и добавить их по пути указанному в /etc/nginx/sites-available/ FQDN сайта в переменных ssl_certificate, ssl_certificate_key. При отсутствии публичного сертификата необходимо закомментировать в файле /etc/nginx/sites-available/FQDN сайта раздел server {} относящийся к 443 порту, но в этом случае сайт не будет работать по протоколу HTTPS, а только по протоколу HTTP.
5. Обновить файл конфигурации config.yml (который ранее был добавлен в каталог /var/www/FQDN сайта/go), а именно - заменить параметры dbname, user и pw, соответственно на имя созданной базы данных, имя созданного пользователя и пароль созданного пользователя.
6. Добавить разрешение на запуск файла go (который ранее был добавлен в каталог /var/www/FQDN сайта/go) командой chmod +x имя файла
7. Скопировать файл службы pfp.go.service в каталог /etc/systemd/system.
В этом файле при необходимости нужно заменить параметры execstart (указать путь к go файлу, скопированному ранее) и workingdirectory (указать путь в папку с файлом конфигурации go). Активировать службу командой:
\$ systemctl enable pfp.go

```
root@host16:/etc/systemd/system#  
root@host16:/etc/systemd/system# systemctl enable pfp.go  
Created symlink /etc/systemd/system/multi-user.target.wants/pfp.go.service → /etc/systemd/system/pfp.go.service.
```

Рис. 19 Установка сервиса pfp.go

Приложение работает и доступно по адресу домена!!!