

Урок 6. Запуск стека для веб-приложения.

Домашнее задание:

1. Установить Nginx и настроить его на работу с PHP-FPM.
2. Установить Apache. Настроить обработку PHP. Добиться одновременной работы с Nginx.
3. Настроить схему обратного прокси для Nginx (динамика - на Apache).
4. Установить MySQL. Создать новую базу данных и таблицу в ней.
- 5.* Установить пакет phptuadmin и запустить его веб-интерфейс для управления MySQL.
- 6.* Настроить схему балансировки трафика между несколькими серверами Apache на стороне Nginx с помощью модуля ngx_http_upstream_module.

Решение:

1. Установить Nginx и настроить его на работу с PHP-FPM.

```
sudo apt install nginx
systemctl status nginx - проверка запуска nginx
curl localhost - проверка работоспособности nginx
sudo su
cd /etc/nginx
ll
cd sites-enabled/
ll - на данный момент у нас есть единый сайт default
nano default
меняем конфигурацию:
```

```
location / {
    proxy_pass http://localhost:8080;
    proxy_set_header Host $host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Real-IP $remote_addr;
}
location ~* \.(jpg | jpeg | gif | png | ico | css | zip | pdf | txt | tar | js)$ {
    root /var/www/html;
}
```

сохраняем конфигурацию

nginx -s reload - если не видим никаких ошибок, можем работать дальше

Установка PHP

```
apt install libapache2-mod-php8.1 php8.1
```

создадим файл

```
cd /var/www/html
```

```
nano info.php
```

```
<?php
phpinfo();
?>
```

Установка сервера PHP-FPM

```
apt install php8.1-fpm
```

```
cd /etc/php/8.1/fpm/
```

вносим следующую секцию в конфигурацию сайта в Nginx.

```
location ~ \.php$ {
    include snippets/fastcgi-php.conf;
    root /var/www/html;
    fastcgi_pass unix:/run/php/php8.1-fpm.sock;
}
```

Заходим на **http://localhost/info.php** чтобы найти упоминание сервиса PHP-FPM в выводе скрипта в браузере.

2. Установить Apache. Настроить обработку PHP. Добиться одновременной работы с Nginx.

```
apt install apache2
```

```
cd /etc/apache2
```

```
ll
```

nano ports.conf - меняем порт с 80 на 8080, сохраняем редактирование

```
systemctl restart apache2
```

```
systemctl status apache2
```

curl localhost:8080 - (если не работает команда curl, можно скачать снippet) порт настроен

```
ss -tnlp
```

3. Настроить схему обратного прокси для Nginx (динамика - на Apache).

```
nano /etc/apache2/sites-enabled/000-default.conf
```

```
cd /var/www/html
```

```
nano index.html
```

```
nano index.html
```

```
service apache2 restart
```

```
nano /etc/nginx/sites-enabled/default
```

```
cp ~/.nanorc /root
```

```
nano /etc/nginx/sites-enabled/default
```

```
service nginx restart
```

```
nano /etc/nginx/sites-enabled/default
```

```
service nginx restart
```

4. Установить MySQL. Создать новую базу данных и таблицу в ней.

```
apt install mysql-server-8.0
ps afx
sudo mysql
service mysql status
```

Команды SQL:

Посмотрим список баз данных:
show databases;

Зайдём в системную БД mysql и посмотрим список пользователей.
use mysql;

Создадим новую базу данных и таблицу в ней:

```
SELECT * FROM user\G
CREATE DATABASE gb_test;
CREATE TABLE test(i INT);
SHOW TABLES;
```

Создадим несколько записей в новой таблице и проверим, что они появились:

```
INSERT INTO test (i) VALUES (1),(2),(3),(4);
SELECT * FROM test;
exit
```

5.* Установить пакет phpmyadmin и запустить его веб-интерфейс для управления MySQL.

```
apt install phpmyadmin - установили пароль
```

```
cd /etc/nginx
```

```
cd sites-enabled/
```

```
nano default
```

```
location / {
    proxy_pass http://localhost:8080;
    proxy_set_header Host $host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Real-IP $remote_addr;
}
```

```
nginx -s reload
```

идем на наш сервер по ip адресу - 192.168.193.144/phpmyadmin

настроить логин и пароль для пользователя

6.* Настроить схему балансировки трафика между несколькими серверами Apache на стороне Nginx с помощью модуля ngx_http_upstream_module.

```
nano default
```

до параметров server внести группу серверов с разными портами:

```

upstream backend {
    server 127.0.0.1:8080;
    server 127.0.0.1:8081;
}
location / {
    proxy_pass http://backend;
    proxy_set_header Host $host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Real-IP $remote_addr;
}

```

сохраняем конфигурацию

nginx -s reload - если не видим никаких ошибок, можем работать дальше

- Настройки apache2

```
cd /etc/apache2
```

```
ll
```

```
cd sites-enabled/
```

```
ll
```

```
nano 000-default.conf
```

```

<VirtualHost *:8080>
DocumentRoot/var/www/html
копируем его
Listen 8081
<VirtualHost *:8081>
DocumentRoot/var/www/html1

```

```
apachectl -t
```

```
service apache2 reload
```

```
cd /var/www
```

```
cp -r html html1
```

```
cd html1
```

nano index.html - отредактируем файл index в заголовке

```
Apache2 Default Page :8081
```

идем на страницу и начинаем обновляться

```
service apache2 restart
```

```
nginx -s reload
```

```
service nginx restart
```

```
cd /etc/nginx/sites-enabled/
```

```
nano default
```

```
cd /etc/apache2/sites-enabled/
```

```
nano 000-default.conf
```

```
cd /var/www/html1
```

II

nano index.html

Apache2 Default Page ddd :8081

nginx -s reload

iptables -nvL

cd /etc/nginx/sites-enabled/

nano default

Проверяем балансировку через консоль

curl localhost | grep Default

curl localhost | grep Default

Через несколько запросов мы будем видеть разные заголовки.