

🌐 Руководство по Nginx с примерами

Оглавление

- [🌐 Руководство по Nginx с примерами](#)
 -  [Оглавление](#)
 - [Что такое Nginx](#)
 - [Пример конфигурации сайта](#)
 - [а. Файл конфигурации: /etc/nginx/sites-enabled/site.ua](#)
 - [б. После изменения конфигурации:](#)
 - [Добавление нового сайта](#)
 - [Удаление старого сайта](#)
 - [Полезные команды](#)
- [Ссылки и ресурсы](#)

Что такое Nginx

Nginx — это **высокопроизводительный веб-сервер и обратный прокси-сервер**.

- *По человечески: в зависимости от того что принаимет будет отдавать то что назначите в его файле*

Он умеет:

- Обслуживать статические файлы (HTML, CSS, JS) 
 - Работать как обратный прокси для приложений ↲
 - Поддерживать SSL/TLS ⚡
 - Балансировать нагрузку между несколькими серверами ⚡
-

Пример конфигурации сайта

а. Файл конфигурации:

`/etc/nginx/sites-enabled/site.ua`

- *www.site.ua заменить на ваш домен соответственно сертификаты для https как получить [отдельно](#)*

```
# http://www.site.ua/  
  
server {  
    listen 80;  
    listen [::]:80;  
    server_name site.ua  
    www.site.ua;  
  
        # редирект всех HTTP  
        # запросов на HTTPS  
        return 301  
        https://$host$request_uri;
```

```
# выдача сайта
закомментируй верхнюю
строчку разакомментируй две
нижних

# root
/var/www/site.ua;
    # index index.html;
}

# https://www.site.ua/

server {
    listen 443 ssl;
    server_name uvs.site.ua
www.site.ua;

    ssl_certificate
/etc/letsencrypt/live/site.
ua/fullchain.pem;
    ssl_certificate_key
/etc/letsencrypt/live/site.
ua/privkey.pem;
```

```
root /var/www/site.ua;
index index.html
index.htm;
```

пример вставленной
локации то есть по адресу
https://www.site.ua/test/new_site/ будет индексная
страница тестового сайта
(осторожней с навигацией
внутри тестового сайта !!!)

```
location
/test/new_site/ {
    alias
/var/www/test/new_site/dist
/
    index index.html;
}
```

```
location / {
    try_files $uri
$uri/ =404;
}
```

```
}

# Тестовый сайт (на порту
# 3000)
http://www.site.ua:3000/

server {
    listen 3000;
    server_name
www.site.ua;

    root
/var/www/test/new_site/dist
/;
    index index.html;

    location / {
        try_files $uri
$uri/ =404;
    }
}
```

- *http://www.site.ua:3000/ навигация будет нормальная но скорее всего в интернете вы сайт не откроете железный занавес оставил мало портов для пользования и 3000 часто у провайдеров заблокирован но при условии если создать внутреннюю приватную сеть с сервером вы откроете так как вы свой провайдер, но если захотите выйти на чужой сайт с другим портом будете подпадать под провайдера сервера и его правила подробнее про **Виртуальную Приватную Сеть** читай [отдельно](#)*

б. После изменения конфигурации:

```
sudo nginx -t      # Проверка конфигурации
```

- важно проверять файл конфигурации пока не перезагрузил (все работает) в случае ошибки ляжет весь **nginx** а не один сайт

```
sudo systemctl reload nginx  
# Перезагрузка сервера
```

Проверка работы сайта :

```
curl -I  
https://site.ua/test/aleksa  
ndra/
```

Добавление нового сайта

Создаем новый конфиг-файл:

```
sudo nano /etc/nginx/sites-available/my_new_site.conf
```

Пример содержимого:

```
server {
    listen 443 ssl;
    server_name
my_new_site.com
www.my_new_site.com;

    ssl_certificate
/etc/letsencrypt/live/my_ne
w_site.com/fullchain.pem;
    ssl_certificate_key
/etc/letsencrypt/live/my_ne
w_site.com/privkey.pem;

    root
/var/www/my_new_site.com;
    index index.html;
```

```
location / {  
    try_files $uri  
    $uri/ =404;  
}  
}
```

Создаем символическую ссылку:

```
sudo ln -s  
/etc/nginx/sites-  
available/my_new_site.conf  
/etc/nginx/sites-enabled/
```

Проверяем и перезагружаем Nginx:

```
sudo nginx -t
```

```
sudo systemctl reload nginx
```

Удаление старого сайта

Удаляем символическую ссылку из sites-enabled:

```
sudo rm /etc/nginx/sites-enabled/old_site.conf
```

(Опционально) Удаляем файл конфигурации:

```
sudo rm /etc/nginx/sites-available/old_site.conf
```

Перезагружаем Nginx:

```
sudo nginx -t
```

```
sudo systemctl reload nginx
```

- ⚠ Важно: сначала удаляйте символическую ссылку, потом файл, иначе можно случайно сломать сайт.

Полезные команды

Команда

---> Описание

```
sudo nginx -t
```

---> Проверка конфигурации

```
sudo systemctl reload nginx
```

---> Перезагрузка без остановки

```
sudo systemctl restart  
nginx    ---> Полная  
перезагрузка  
sudo nginx -T  
---> Вывод всех  
конфигураций
```

```
curl -I https://example.com  
---> Проверка ответа  
сервера
```

Ссылки и ресурсы

//НЕ работают!!! Пока еще!

[Официальная документация Nginx](#) 

[Let's Encrypt SSL](#) 