

Практическая работа № 4. Знакомство с веб-сервером nginx. Установка, настройка.

Nginx [engine x] — это HTTP-сервер и обратный прокси-сервер, почтовый прокси-сервер, а также TCP/UDP прокси-сервер общего назначения, изначально написанный Игорем Сысоевым. Уже длительное время он обслуживает серверы многих высоконагруженных российских сайтов, таких как Яндекс, Mail.Ru, ВКонтакте и Рамблер. Согласно статистике Netcraft nginx обслуживал или проксировал 29.64% самых нагруженных сайтов в ноябре 2017 года.

Стандартные порты и пути к файлам Nginx

/usr/local/nginx/conf, /etc/nginx или /usr/local/etc/nginx — директория содержащая файлы настроек Nginx сервера, зависит от текущей версии nginx, путь остальных файлов тоже может быть другой, если директория НЕ /etc/nginx.

/etc/nginx/conf.d/ — директория содержащая файлы настроек SSL и виртуальных хостов (vhost);

/etc/nginx/sites-enabled/ — директория содержащая файлы настроек SSL и виртуальных хостов (vhost);

/etc/nginx/nginx.conf — основной файл настроек;

/var/log/nginx/error.log — логи ошибок;

/var/log/nginx/access.log — логи доступа;

/var/www/html/ — корневая директория сайта по умолчанию (document root), в последних версиях возможен другой путь, например /usr/local/nginx/html;

TCP 80 — HTTP порт по умолчанию для Nginx;

TCP 443 — HTTPS порт по умолчанию для Nginx.

Установка nginx

1. Есть несколько вариантов установки nginx: сбилдить из исходников (используется если нужна специфичная конфигурация, к примеру нужны/не нужны определенные модули), установить с помощью пакетного менеджера. Будем устанавливать с помощью пакетного менеджера (apt/yum, в зависимости от ОС). Обычно стандартный репозиторий yum/apt уже содержит репозиторий с nginx. Выполняем команду:

RHEL/CentOS: `yum install nginx` (sudo при необходимости)

Debian/Ubuntu: `apt-get install nginx` (sudo при необходимости)

(Инструкция по добавлению репозитория, если они не были найдены
http://nginx.org/ru/linux_packages.html)

2. Запускаем nginx командой:

```
service nginx start  
systemctl start nginx
```

3. Далее проверяем статус службы nginx:

```
service nginx status  
systemctl status nginx
```

Если установка прошла успешно, то вывод данной команды должен выглядеть так:

- `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 \$DATE; 46min ago

Структура конфигурационного файла

Nginx состоит из модулей, которые настраиваются директивами, указанными в конфигурационном файле. Директивы делятся на простые и блочные. Простая директива состоит из имени и параметров, разделенных пробелами, и оканчивается точкой с запятой (;). Блочная директива устроена так же, как и простая директива, но вместо точки с запятой после имени и параметров следует набор дополнительных инструкций, помещенных внутри фигурных скобок ({ и }). Если у блочной директивы внутри фигурных скобок можно задавать другие директивы, то она называется контекстом (примеры: `events`, `http`, `server` и `location`).

Директивы, помещенные в конфигурационном файле вне любого контекста, считаются находящимися в контексте `main`. Директивы `events` и `http` располагаются в контексте `main`, `server` — в `http`, а `location` — в `server`.

Часть строки после символа `#` считается комментарием.

Путь до стандартного конфига:

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

Настройка веб-сервера на отдачу статического контента

По-умолчанию, без специальных настроек `nginx` может отдавать только статический контент (`html/css`, картинки). То есть все `html`-документы лежат на сервере и просто отображаются пользователю по запросу. В случае динамического контента необходимо настраивать специальные модули/сервера приложений, т.к. отображаемая страница формируется “на лету” с помощью `backend’a` (`php/ruby/js/go/java` и т.д.).

Nginx содержит дефолтную страницу, которую можно просмотреть введя в браузере:
`http://localhost`

Если все хорошо, то вы увидите:

Welcome to nginx!

If you see this page, the `nginx` web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.

Commercial support is available at nginx.com.

Thank you for using nginx.

Можно изменить текст на странице отредактировав html-документ:

```
nano (vi / vim) /var/www/html/index.html
```

Изменяем заголовок страницы и заголовок страницы на webserver-<номер вашего варианта>

Сохраняем изменения, обновляем страницу, смотрим изменения.

Теперь создадим свою конфигурацию:

Создаем файл для виртуального хоста

```
/etc/nginx/conf.d/<номер_вашего_варианта>.conf
```

В соседнем окне терминала можно открыть стандартный конфиг nginx'а, чтобы было проще ориентироваться.

Итак, в созданном файле объявим блок блок server:

```
server {  
}
```

Внутри блока server объявим директиву server_name и listen.

server_name - имя сервера, запросы на который будет обрабатывать nginx, в нашем случае это пока что localhost.

port - порт на котором nginx будет слушать/ожидать запросы, стандартный порт http - 80, можно использовать другой, будем использовать 81.

```
server {  
    server_name localhost;  
    listen 81;  
}
```

Теперь добавим блок location. Блок location определяет как будет обрабатываться тот или иной запрос. Например объявим:

```
server {  
    server_name localhost;  
    listen 81;  
    location / {  
        root /var/www/html/;  
    }  
}
```

Этот блок location задаёт “/” в качестве префикса, который сравнивается с URL из запроса. Для подходящих запросов добавлением URL к пути, указанному в директиве root, то есть, в данном случае, к /var/www/html/, получается путь к запрашиваемому файлу в локальной файловой системе (рис. 1). Если есть совпадение с несколькими

блоками location, nginx выбирает блок с самым длинным префиксом. В блоке location выше указан самый короткий префикс, длины один, и поэтому этот блок будет использован, только если не будет совпадения ни с одним из остальных блоков location.

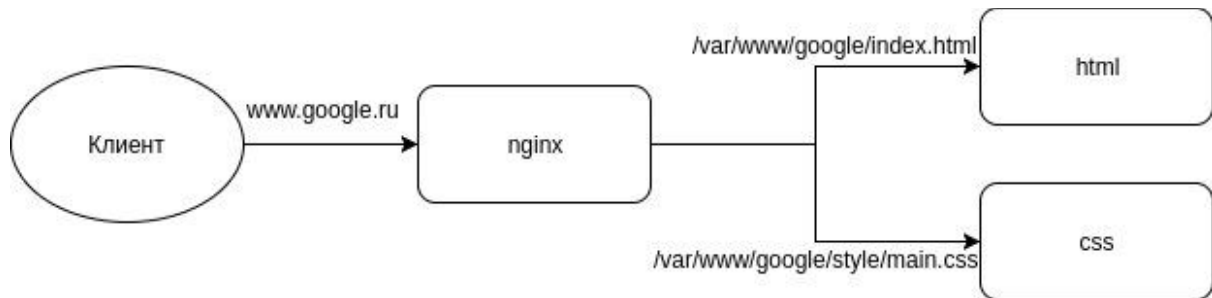


Рисунок 1. Схема взаимодействия веб-сервера с пользователем

По-умолчанию при отсутствии полного пути до статического файла (html-страницы, картинки) nginx будет пытаться найти "index.html" в папке, которая указана в root. Если ваш index имеет специфичное имя необходимо указать это добавив директиву index в блок location.

```
location / {  
    root /var/www/mysite/;  
    index index_1.html;  
}
```

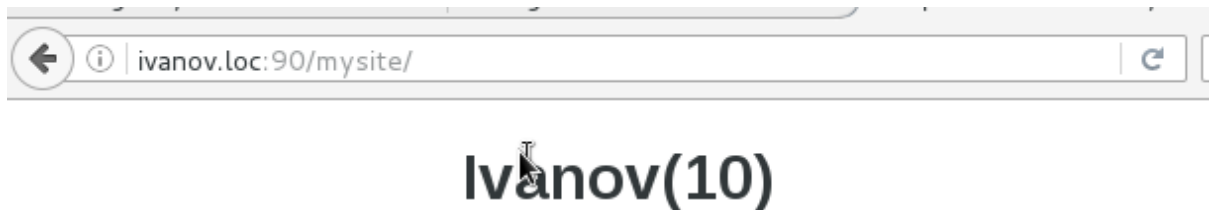
Теперь создадим в папке /var/www/mysite файл index.html, с любым содержимым. Перезагружаем nginx. Заходим в браузер, по адресу <http://localhost:81>. Если все хорошо, то вы увидите содержимое созданного вами index.html файла. Если в данной директории будут другие html-файлы, например, menu.html, page.html, то они будут доступны по адресам <http://localhost/menu.html> <http://localhost/page.html> соответственно. Аналогично с картинками - 1.jpg в папке /var/www/mysite/ = <http://localhost/1.jpg>.

Возможные проблемы и их решение (во всех лабораторных).

1. Если что-то не работает - смотрим логи `/var/log/nginx/error.log`, обычно информации в них достаточно, чтобы понять в чем дело. Удобно открыть соседнее окно терминала и просматривать “логи в реальном времени” командой `tail -f /var/log/nginx/error.log/`
2. На дистрибутиве CentOS возможно проблема с политиками SELinux, отключаем их до перезагрузки командой: `setenforce 0`.

Задание на лабораторную работу.

1. Найти информацию о том, за что отвечает файл `/etc/hosts`. С помощью данного файла и директивы `server_name` сделать так, чтобы при запросе в браузере: `http://<ваша_фамилия>.loc:<port=80+номер в журнале>/mysite/` отображалась html-страница содержащая ваш порядковым номер (как в журнале).
Например: Иванов (номер в журнале - 10).



2. По запросу `http://<ваша_фамилия>.loc:<port=80+номер в журнале>/images/pic.jpg`
Должна отобразиться картинка `/var/www/images/pic.jpg`

Контрольные вопросы:

1. Понятия веб-севера. Какие ещё веб-сервера существуют помимо nginx?
2. Объяснить значение директив(блоков) `server_name`, `location`, `root`, `server`, `listen` и на что они влияют.

Полезные ресурсы.

1. Nginx. Руководство для начинающих. https://nginx.ru/ru/docs/beginners_guide.html