

## Лабораторная работа № 4. Базовая настройка HTTP-сервера Apache

### 4.1. Цель работы

Приобретение практических навыков по установке и базовому конфигурированию HTTP-сервера Apache.

### 4.2. Предварительные сведения

*Протокол передачи гипертекста (HyperText Transfer Protocol, HTTP)* — протокол передачи данных в компьютерных сетях, базирующийся на технологии взаимодействия «клиент–сервер».

*Гипертекст* — текст, отдельные фрагменты, записи или страницы которого представляяют собой ссылки на другие записи.

*Сервер HTTP (веб-сервер)* — приложение, прослушивающее соединение, принимающее HTTP-запросы на обслуживание и посылающее ответы.

*Клиент HTTP* — программа, устанавливающая соединение с целью отправки HTTP-запросов и получения на них ответов (например, браузер или веб-приложение).

В качестве веб-сервера может быть использовано следующее программное обеспечение: Apache HTTP Server, Lighttpd, nginx и др.

Для работы с HTTP в ОС Linux используется демон `httpd` (см. [2]).

Основным файлом конфигурации веб-сервера (в частности, Apache) является файл `httpd.conf`, содержащий директивы, управляющие работой сервера. Подробнее о файлах конфигурации Apache см. в [1].

### 4.3. Задание

1. Установите необходимые для работы HTTP-сервера пакеты (см. раздел 4.4.1).
2. Запустите HTTP-сервер с базовой конфигурацией и проанализируйте его работу (см. разделы 4.4.2 и 4.4.3).
3. Настройте виртуальный хостинг (см. раздел 4.4.4).
4. Напишите скрипт для Vagrant, фиксирующий действия по установке и настройке HTTP-сервера во внутреннем окружении виртуальной машины `server`. Соответствующим образом внесите изменения в `Vagrantfile` (см. раздел 4.4.5).

### 4.4. Последовательность выполнения работы

#### 4.4.1. Установка HTTP-сервера

1. Загрузите вашу операционную систему и перейдите в рабочий каталог с проектом:  

```
cd /var/tmp/user_name/vagrant
```

где `user_name` — идентифицирующее вас имя пользователя, обычно первые буквы инициалов и фамилия.
2. Запустите виртуальную машину `server`:  

```
make server
```

(или, если вы работаете под ОС Windows, то `vagrant up server`).
3. На виртуальной машине `server` войдите под вашим пользователем и откройте терминал. Перейдите в режим суперпользователя:

```
sudo -i
```

4. Установите из репозитория стандартный веб-сервер (HTTP-сервер и утилиты httpd, крипто-утилиты и пр.):

```
LANG=C yum grouplist
dnf -y groupinstall "Basic Web Server"
```

#### 4.4.2. Базовое конфигурирование HTTP-сервера

1. Просмотрите и прокомментируйте в отчёте содержание конфигурационных файлов в каталогах `/etc/httpd/conf` и `/etc/httpd/conf.d`.
2. Внесите изменения в настройки межсетевого экрана узла `server`, разрешив работу `http`:

```
firewall-cmd --list-services
firewall-cmd --get-services
firewall-cmd --add-service=http
firewall-cmd --add-service=http --permanent
```

3. В дополнительном терминале запустите в режиме реального времени расширенный лог системных сообщений, чтобы проверить корректность работы системы:

```
journalctl -x -f
```

4. В первом терминале активируйте и запустите HTTP-сервер:

```
systemctl enable httpd
systemctl start httpd
```

Просмотрев расширенный лог системных сообщений, убедитесь, что веб-сервер успешно запустился.

#### 4.4.3. Анализ работы HTTP-сервера

1. Запустите виртуальную машину `client`:

```
make client
```

(для работающих под ОС Windows: `vagrant up client`).

2. На виртуальной машине `server` просмотрите лог ошибок работы веб-сервера:

```
tail -f /var/log/httpd/error_log
```

3. На виртуальной машине `server` запустите мониторинг доступа к веб-серверу:

```
tail -f /var/log/httpd/access_log
```

На виртуальной машине `client` запустите браузер и в адресной строке введите `192.168.1.1`. Проанализируйте информацию, отразившуюся при мониторинге.

#### 4.4.4. Настройка виртуального хостинга для HTTP-сервера

Требуется настроить виртуальный хостинг по двум DNS-адресам: `server.user.net` и `www.user.net`.

1. Приостановите работу DNS-сервера для внесения изменений в файлы описания DNS-зон:

```
systemctl stop named
```

2. Добавьте запись для HTTP-сервера в конце файла прямой DNS-зоны `/var/named/master/fz/user.net`:

```
server    A      192.168.1.1
www       A      192.168.1.1
```

и в конце файла обратной зоны `/var/named/master/rz/192.168.1`:

```
1         PTR    server.user.net.
1         PTR    www.user.net.
```

(вместо user укажите свой логин).

При этом не забудьте в обоих файлах изменить серийный номер файла зоны, указав текущую дату в нотации ГГТММДДВВ. Также из соответствующих каталогов следует удалить файлы журналов DNS: user.net.jnl и 192.168.1.jnl.

3. Перезапустите DNS-сервер:

```
systemctl start named
```

4. В каталоге /etc/httpd/conf.d создайте файлы server.user.net.conf и www.user.net.conf (вместо user укажите свой логин):

```
cd /etc/httpd/conf.d
touch server.user.net.conf
touch www.user.net.conf
```

5. Откройте на редактирование файл server.user.net.conf и внесите следующее содержание:

```
<VirtualHost *:80>
    ServerAdmin webmaster@user.net
    DocumentRoot /var/www/html/server.user.net
    ServerName server.user.net
    ErrorLog logs/server.user.net-error_log
    CustomLog logs/server.user.net-access_log common
</VirtualHost>
```

(вместо user укажите свой логин).

6. Откройте на редактирование файл www.user.net.conf и внесите следующее содержание:

```
<VirtualHost *:80>
    ServerAdmin webmaster@user.net
    DocumentRoot /var/www/html/www.user.net
    ServerName www.user.net
    ErrorLog logs/www.user.net-error_log
    CustomLog logs/www.user.net-access_log common
</VirtualHost>
```

(вместо user укажите свой логин).

7. Перейдите в каталог /var/www/html, в котором должны находиться файлы с содержанием (контентом) веб-серверов, и создайте тестовые страницы для виртуальных веб-серверов server.user.net и www.user.net.

Для виртуального веб-сервера server.user.net (вместо user укажите свой логин):

```
cd /var/www/html
mkdir server.user.net
cd /var/www/html/server.user.net
touch index.html
```

Откройте на редактирование файл index.html и внесите следующее содержание:

```
Welcome to the server.user.net server.
```

(вместо user укажите свой логин).

Для виртуального веб-сервера www.user.net (вместо user укажите свой логин):

```
cd /var/www/html
mkdir www.user.net
cd /var/www/html/www.user.net
touch index.html
```

Откройте на редактирование файл index.html и внесите следующее содержание:

```
Welcome to the www.user.net server.
```

(вместо user укажите свой логин).

8. Скорректируйте права доступа в каталог с веб-контентом:

```
chown -R apache:apache /var/www
```

9. Восстановите контекст безопасности в SELinux:

```
restorecon -vR /etc
restorecon -vR /var/named
restorecon -vR /var/www
```

10. Перезапустите HTTP-сервер:
 

```
systemctl restart httpd
```
11. На виртуальной машине client убедитесь в корректном доступе к веб-серверу по адресам `server.user.net` и `www.user.net` (вместо `user` укажите свой логин) в адресной строке веб-браузера.

#### 4.4.5. Внесение изменений в настройки внутреннего окружения виртуальной машины

1. На виртуальной машине `server` перейдите в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/`, создайте в нём каталог `http`, в который поместите в соответствующие подкаталоги конфигурационные файлы HTTP-сервера:

```
cd /vagrant/provision/server
mkdir -p /vagrant/provision/server/http/etc/httpd/conf.d
mkdir -p /vagrant/provision/server/http/var/www/html
cp -R /etc/httpd/conf.d/*
↪ /vagrant/provision/server/http/etc/httpd/conf.d/
cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
```

2. Замените конфигурационные файлы DNS-сервера:

```
cd /vagrant/provision/server/dns/
cp -R /var/named/* /vagrant/provision/server/dns/var/named/
```

3. В каталоге `/vagrant/provision/server` создайте исполняемый файл `http.sh`:

```
cd /vagrant/provision/server
touch http.sh
chmod +x http.sh
```

Открыв его на редактирование, пропишите в нём следующий скрипт:

```
#!/bin/bash
```

```
echo "Provisioning script $0"
```

```
echo "Install needed packages"
```

```
dnf -y groupinstall "Basic Web Server"
```

```
echo "Copy configuration files"
```

```
cp -R /vagrant/provision/server/http/etc/httpd/* /etc/httpd
```

```
cp -R /vagrant/provision/server/http/var/www/* /var/www
```

```
chown -R apache:apache /var/www
```

```
restorecon -vR /etc
```

```
restorecon -vR /var/www
```

```
echo "Configure firewall"
```

```
firewall-cmd --add-service=http
```

```
firewall-cmd --add-service=http --permanent
```

```
echo "Start http service"
```

```
systemctl enable httpd
systemctl start httpd
```

Этот скрипт, по сути, повторяет произведённые вами действия по установке и настройке HTTP-сервера.

4. Для отработки созданного скрипта во время загрузки виртуальных машин в конфигурационном файле Vagrantfile необходимо добавить в конфигурации сервера следующую запись:

```
server.vm.provision "server http",
    type: "shell",
    preserve_order: true,
    path: "provision/server/http.sh"
```

#### 4.5. Содержание отчёта

1. Титульный лист с указанием номера лабораторной работы и ФИО студента.
2. Формулировка задания работы.
3. Описание результатов выполнения задания:
  - скриншоты (снимки экрана), фиксирующие выполнение работы;
  - подробное описание настроек служб в соответствии с заданием;
  - полные тексты конфигурационных файлов настраиваемых в работе служб;
  - результаты проверки корректности настроек служб в соответствии с заданием (подтверждённые скриншотами).
4. Выводы, согласованные с заданием работы.
5. Ответы на контрольные вопросы.

#### 4.6. Контрольные вопросы

1. Через какой порт по умолчанию работает Apache?
2. Под каким пользователем запускается Apache и к какой группе относится этот пользователь?
3. Где располагаются лог-файлы веб-сервера? Что можно по ним отслеживать?
4. Где по умолчанию содержится контент веб-серверов?
5. Каким образом реализуется виртуальный хостинг? Что он даёт?

#### Список литературы

1. Apache HTTP Server Version 2.4 Documentation. — URL: <http://httpd.apache.org/docs/current/>.
2. Httpd — Apache Hypertext Transfer Protocol Server. — URL: <https://httpd.apache.org/docs/2.4/programs/httpd.html>.