РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра теории вероятностей и кибербезопасности

Отчет лабораторной работы 4

Дисциплина: Администрирование сетевых подсистем

Студент: Астахова Марина

Группа: НПИбд-02-23

Тема: Базовая настройка HTTP-сервера Apache

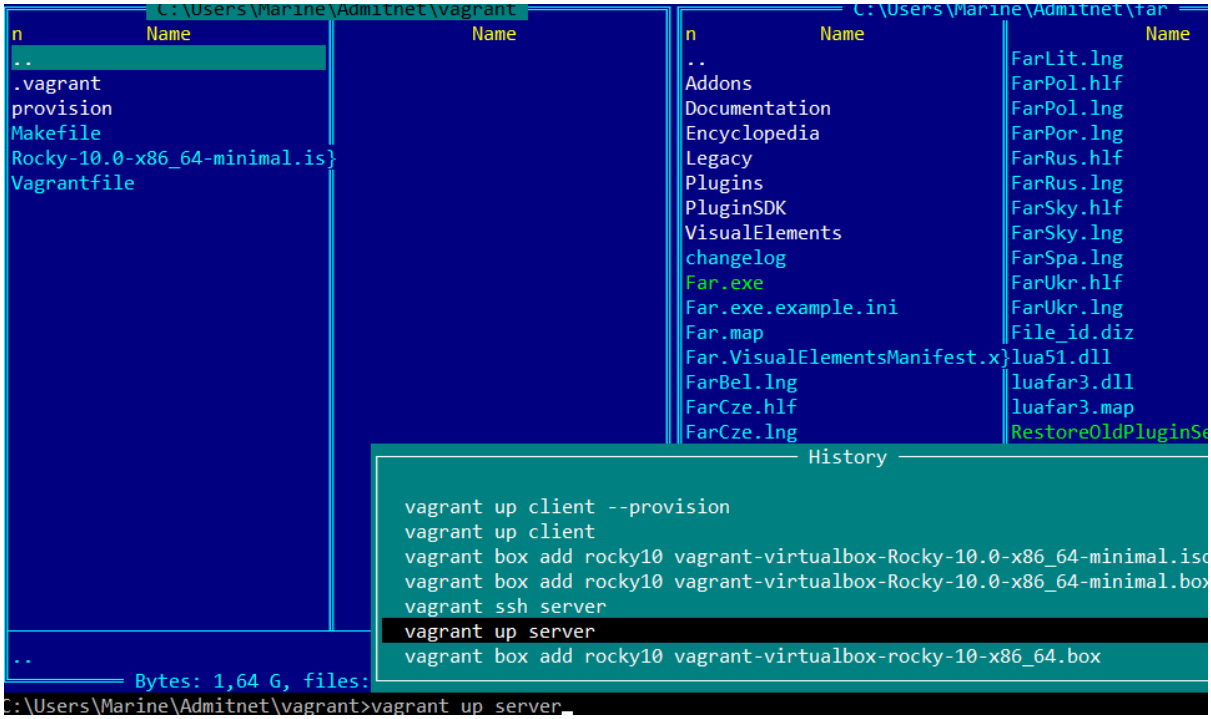
4.1. Цель работы.

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

4.2. Выполнение работы

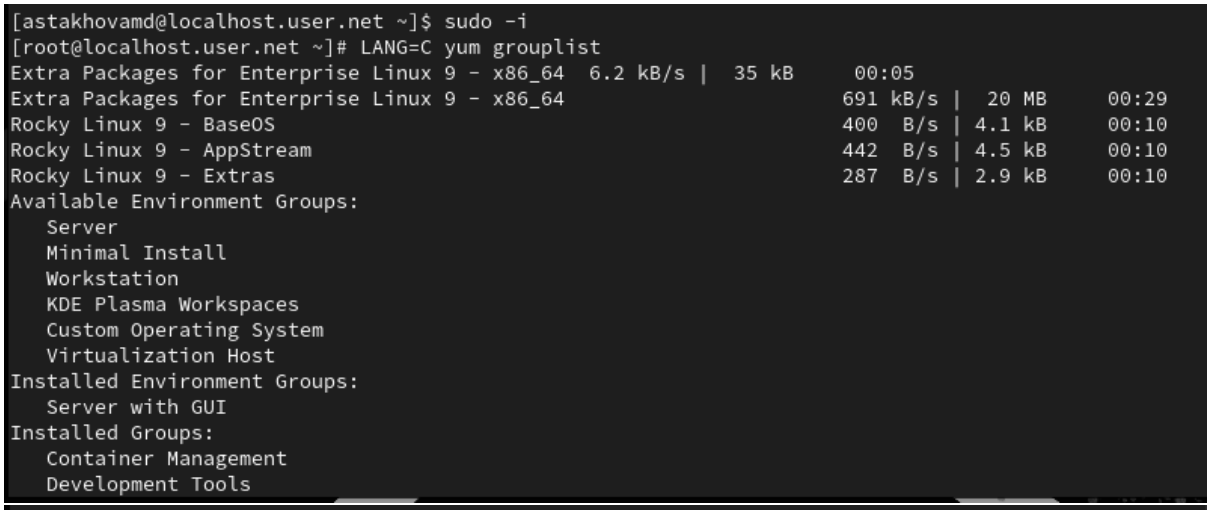
1. Установите необходимые для работы HTTP-сервера пакеты. Установка HTTP-сервера.

Сначала запускаем виртуальную машину с помощью команды vagrant up server. Так как операционная система Windows, то работаем через far.

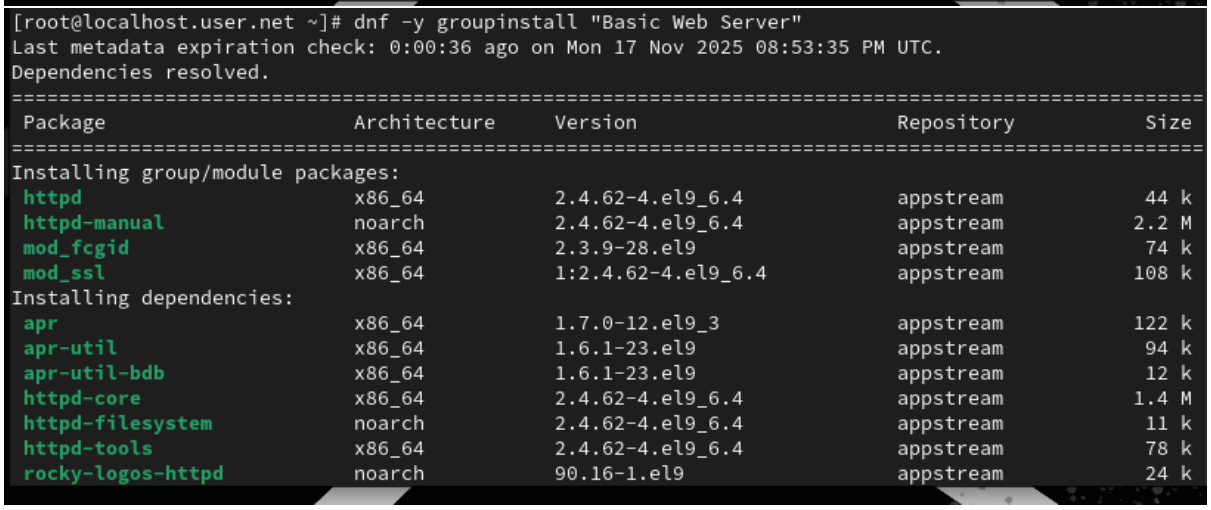


Команда vagrant up server в far

Установим из репозитория стандартный веб-сервер (HTTP-сервер и утилиты httpd, криптоутилиты и пр.):



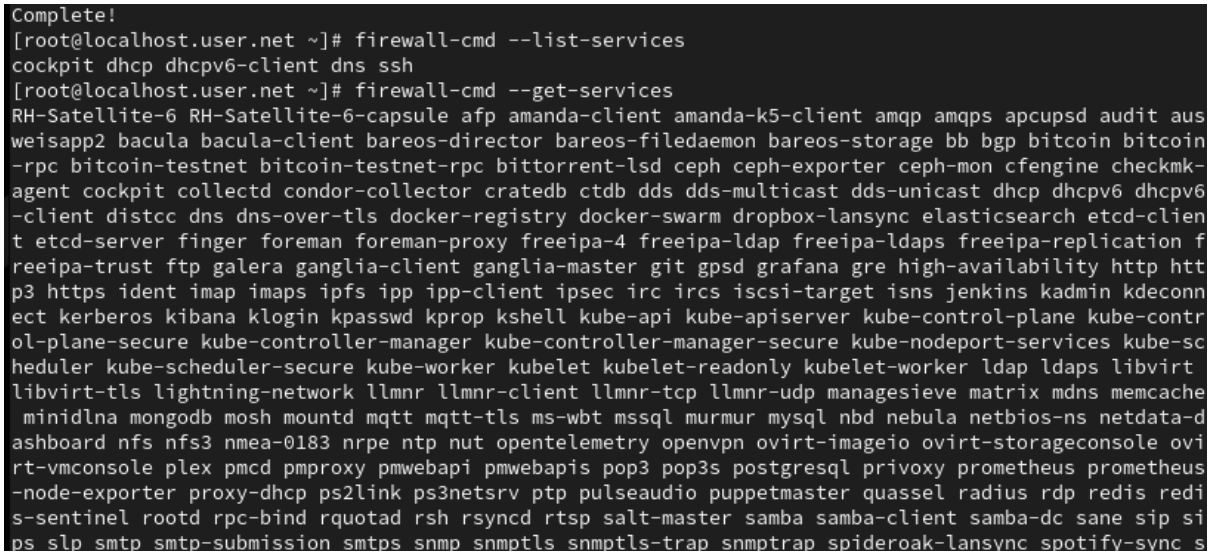
```
[astakhovamd@localhost.user.net ~]$ sudo -i
[root@localhost.user.net ~]# LANG=C yum grouplist
Extra Packages for Enterprise Linux 9 - x86_64  6.2 kB/s |  35 kB     00:05
Extra Packages for Enterprise Linux 9 - x86_64                691 kB/s |  20 MB     00:29
Rocky Linux 9 - BaseOS                                        400  B/s | 4.1 kB     00:10
Rocky Linux 9 - AppStream                                     442  B/s | 4.5 kB     00:10
Rocky Linux 9 - Extras                                        287  B/s | 2.9 kB     00:10
Available Environment Groups:
   Server
   Minimal Install
   Workstation
   KDE Plasma Workspaces
   Custom Operating System
   Virtualization Host
Installed Environment Groups:
   Server with GUI
Installed Groups:
   Container Management
   Development Tools
```



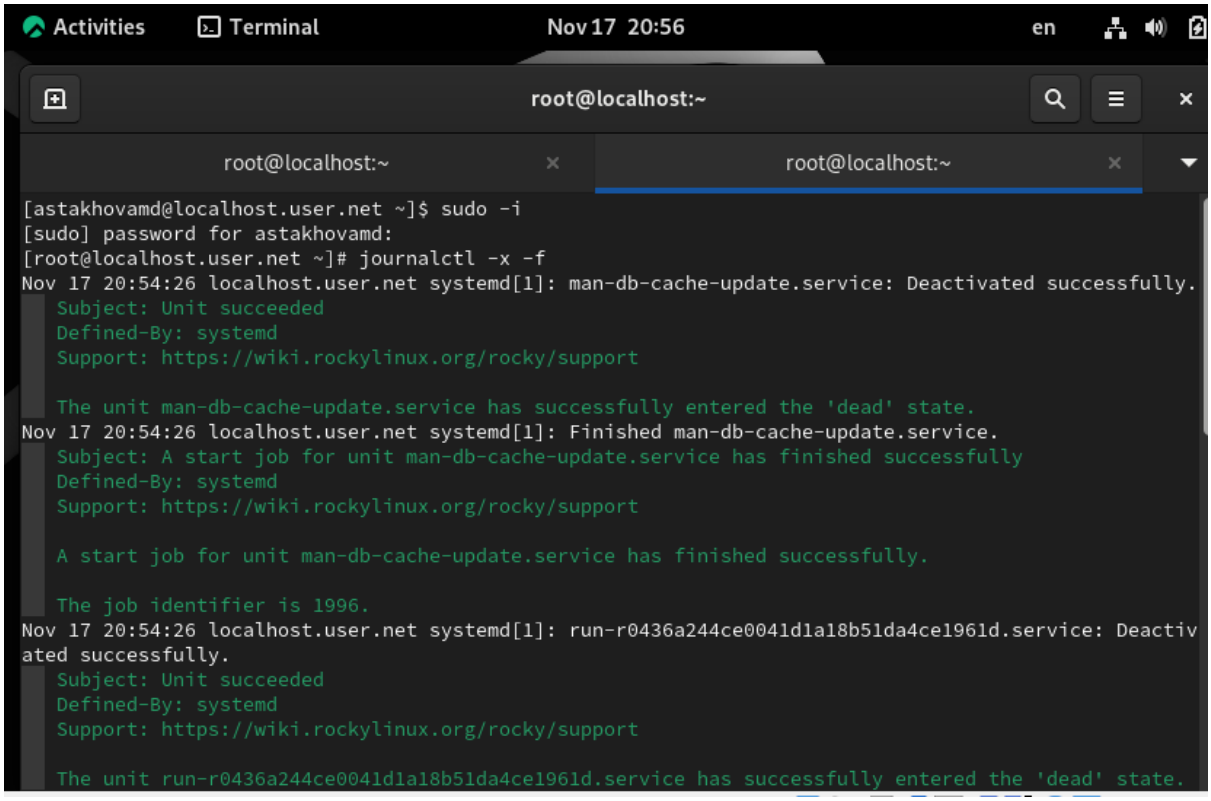
```
[root@localhost.user.net ~]# dnf -y groupinstall "Basic Web Server"
Last metadata expiration check: 0:00:36 ago on Mon 17 Nov 2025 08:53:35 PM UTC.
Dependencies resolved.
================================================================================
 Package                 Architecture     Version                  Repository          Size
================================================================================
Installing group/module packages:
 httpd                   x86_64           2.4.62-4.el9_6.4         appstream           44 k
 httpd-manual            noarch           2.4.62-4.el9_6.4         appstream          2.2 M
 mod_fcgid               x86_64           2.3.9-28.el9             appstream           74 k
 mod_ssl                 x86_64           1:2.4.62-4.el9_6.4       appstream          108 k
Installing dependencies:
 apr                     x86_64           1.7.0-12.el9_3           appstream          122 k
 apr-util                x86_64           1.6.1-23.el9             appstream           94 k
 apr-util-bdb            x86_64           1.6.1-23.el9             appstream           12 k
 httpd-core              x86_64           2.4.62-4.el9_6.4         appstream          1.4 M
 httpd-filesystem        noarch           2.4.62-4.el9_6.4         appstream           11 k
 httpd-tools             x86_64           2.4.62-4.el9_6.4         appstream           78 k
 rocky-logos-httpd       noarch           90.16-1.el9              appstream           24 k
```

2. Запустите HTTP-сервер с базовой конфигурацией и проанализируйте его работу. Базовое конфигурирование HTTP-сервера.

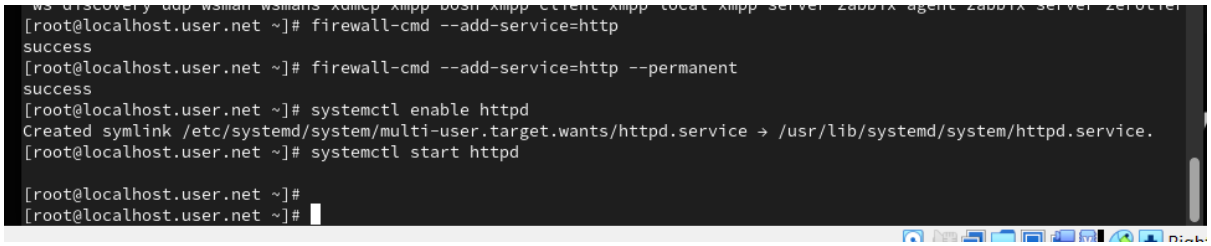
Внесем изменения в настройки межсетевого экрана узла server, разрешив работу с http:



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

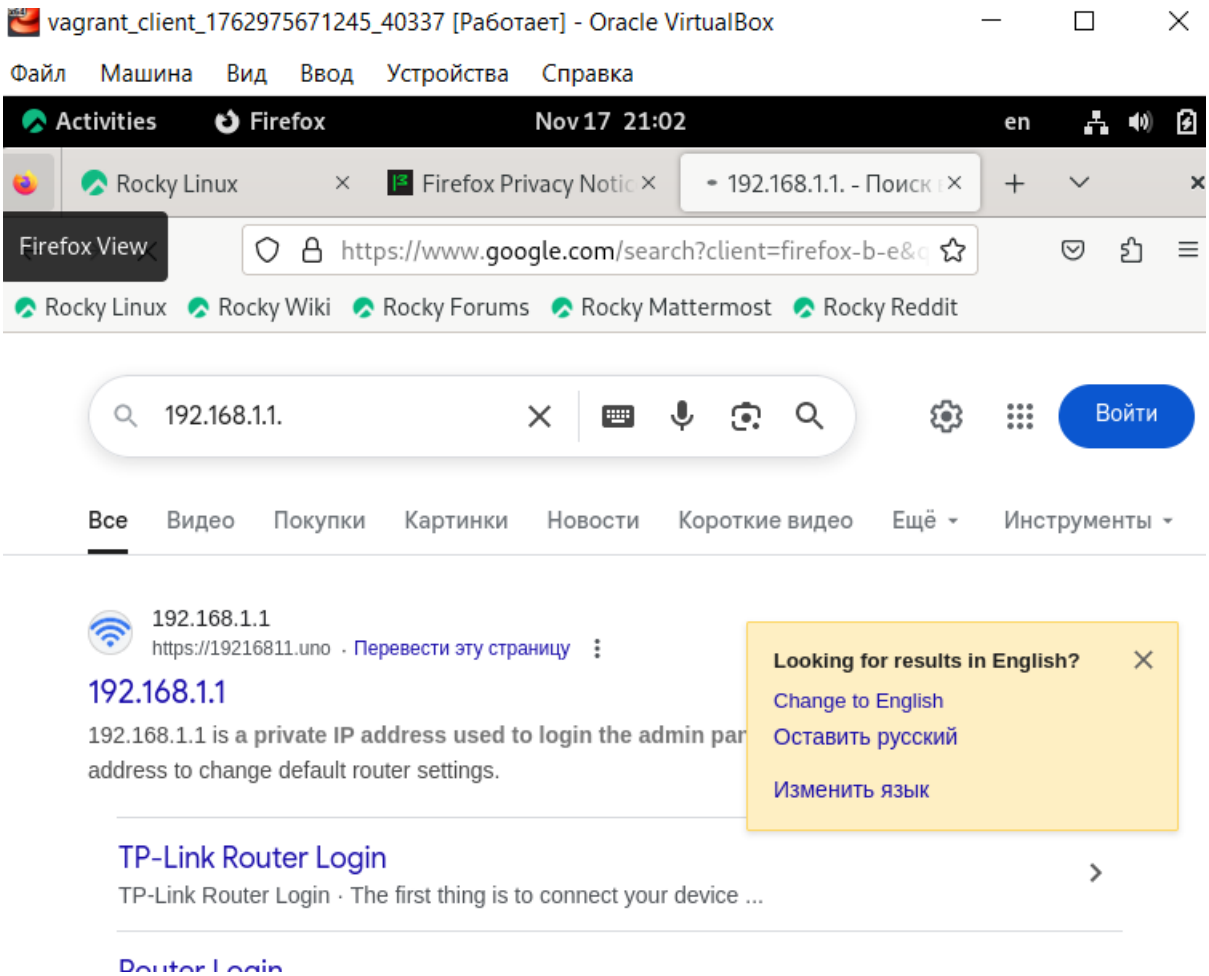


В первом терминале активировали и запустили HTTP-сервер:



3. Анализ работы HTTP-сервера

Результат запроса в client

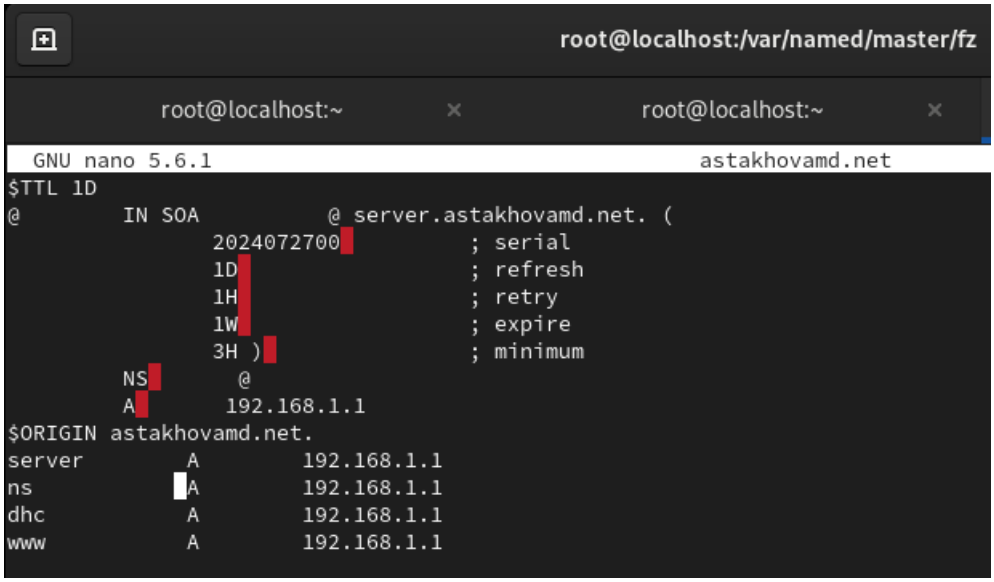


4. Настройте виртуальный хостинг. Настройка виртуального хостинга для HTTP-сервера.

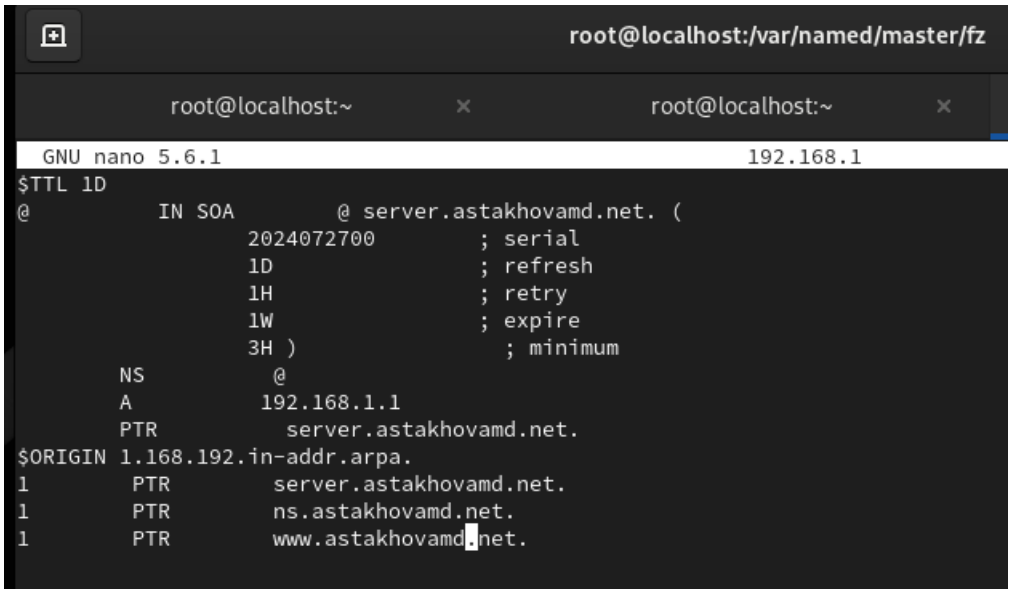
Остановили работу DNS-сервера для внесения изменений в файлы описания DNS зон:

```
[astakhovamd@localhost.user.net ~]$ sudo -i
[sudo] password for astakhovamd:
[root@localhost.user.net ~]# systemctl stop named
[root@localhost.user.net ~]# cd /var/named/master/fz
[root@localhost.user.net fz]#
```

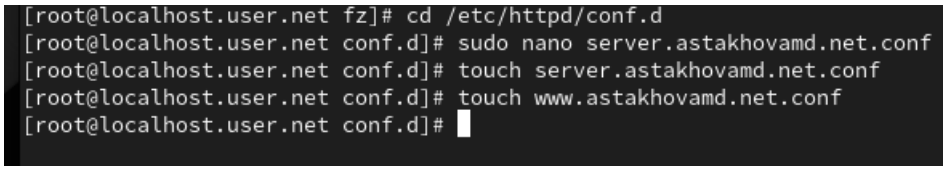
Добавили запись для HTTP-сервера в конце файла прямой DNS-зоны /var/named/master/fz/astakhovamd.net:



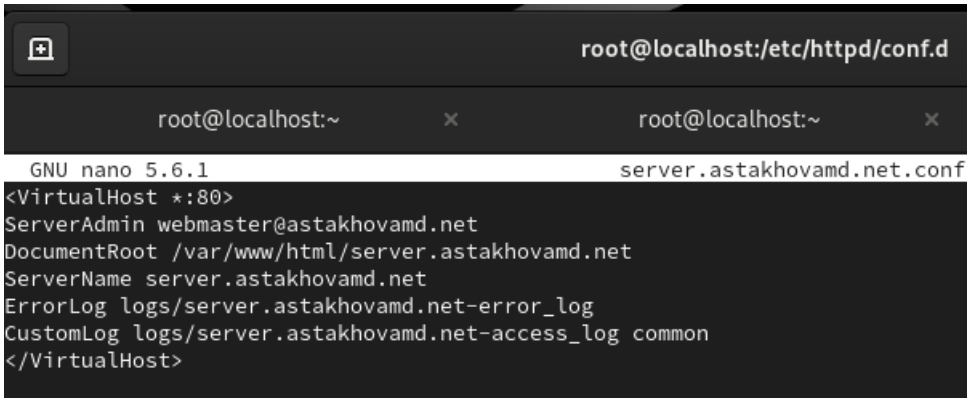
И в файле 192.168.1



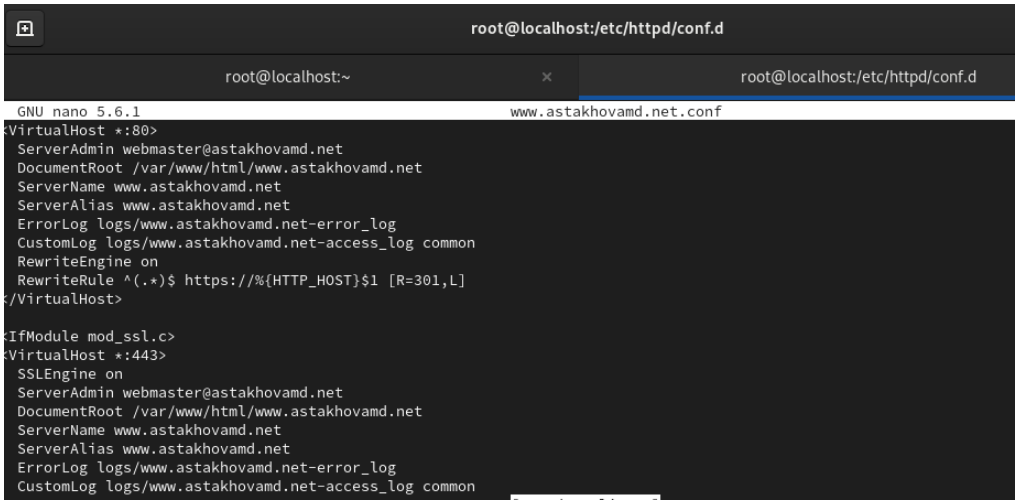
В каталоге /etc/httpd/conf.d создали файлы server.astakhovamd.net.conf и www.astakhovamd.net.conf:



Открыли на редактирование файл server.astakhovamd.net.conf и внесли следующее содержание:

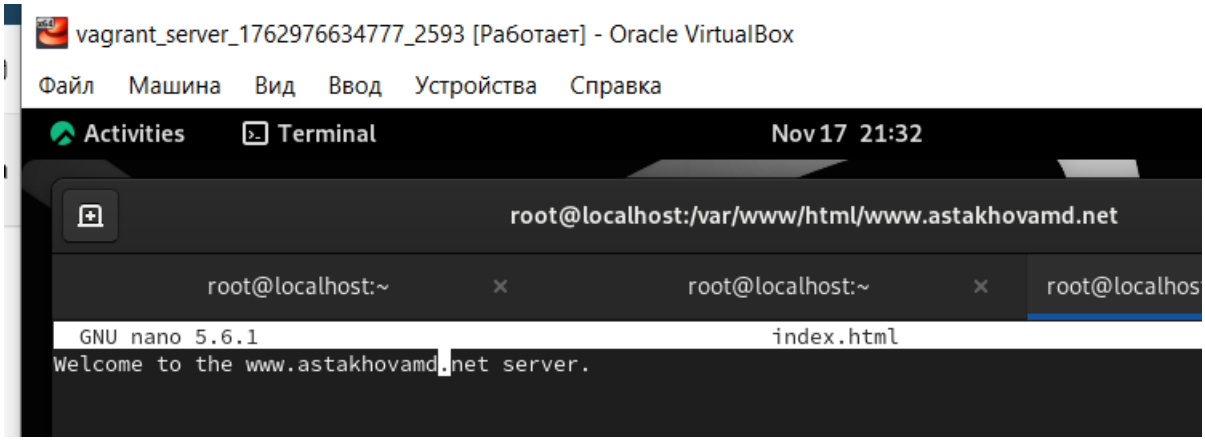


Открыли на редактирование файл www.astakhovamd.net.conf и внесли следующее содержание:



Перейдем в каталог /var/www/html, в котором должны находиться файлы с содержимым (контентом) веб-серверов, и создадим тестовые страницы для виртуальных веб-серверов server.astakhovamd.net и www.astakhovamd.net. Для виртуального веб-сервера server.astakhovamd.net:

Открыли на редактирование файл index.html и внесли следующее содержание: Welcome to the server.astakhovamd.net.server.



Скорректировали права доступа в каталог с веб-контентом и восстановили контекст безопасности, и перезапустили сервер:

```
[root@localhost.user.net www.astakhovamd.net]# sudo nano index.html
[root@localhost.user.net www.astakhovamd.net]# chown -R apache:apache /var/www
[root@localhost.user.net www.astakhovamd.net]# restorecon -vR /etc
[root@localhost.user.net www.astakhovamd.net]# restorecon -vR /var/named
[root@localhost.user.net www.astakhovamd.net]# restorecon -vR /var/www
[root@localhost.user.net www.astakhovamd.net]# systemctl restart httpd
```

5. Напишите скрипт для Vagrant, фиксирующий действия по установке и настройке HTTP-сервера во внутреннем окружении виртуальной машины server. Соответствующим образом внесите изменения в Vagrantfile. Внесение изменений в настройки внутреннего окружения виртуальной машины.

На виртуальной машине server перешли в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создали в нём каталог http, в который поместили в соответствующие подкаталоги конфигурационные файлы HTTP-сервера:

```
[root@localhost.user.net www.astakhovamd.net]# cd /vagrant/provision/server
[root@localhost.user.net server]# mkdir -p /vagrant/provision/server/http/etc/httpd/conf.d
[root@localhost.user.net server]# mkdir -p /vagrant/provision/server/http/var/www/html
[root@localhost.user.net server]# cp -R /etc/httpd/conf.d/*
↪ /vagrant/provision/server/http/etc/httpd/conf.d/
cp: target '/etc/httpd/conf.d/www.astakhovamd.net.conf' is not a directory
bash: ↪: command not found...
[root@localhost.user.net server]# cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/httpd/conf.d/
[root@localhost.user.net server]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
[root@localhost.user.net server]#
```

Заменили конфигурационные файлы DNS-сервера:

```
[root@localhost.user.net server]# cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/httpd/conf.d/
[root@localhost.user.net server]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
[root@localhost.user.net server]# cd /vagrant/provision/server/dns/
[root@localhost.user.net dns]# cp -R /var/named/* /vagrant/provision/server/dns/var/named/
cp: overwrite '/vagrant/provision/server/dns/var/named/data/named.run'? cp -R /var/named/* /vagrant/provision/server/
```

```
[astakhovamd@localhost.user.net ~]$ sudo -i
[sudo] password for astakhovamd:
[root@localhost.user.net ~]# cp -R /var/named/* /vagrant/provision/server/dns/var/named
cp: overwrite '/vagrant/provision/server/dns/var/named/data/named.run'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind.jnl'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/astakhovamd.net'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/192.168.1'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/user.net'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/192.168.1'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.astakhovamd.net.zone'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.ca'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.empty'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.localhost'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.loopback'? Y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.user.net.zone'? Y
[root@localhost.user.net ~]#
```

```
[root@localhost.user.net www.astakhovamd.net]# cd /vagrant/provision/server
[root@localhost.user.net server]# mkdir -p /vagrant/provision/server/http/etc/httpd/conf.d
[root@localhost.user.net server]# mkdir -p /vagrant/provision/server/http/var/www/html
[root@localhost.user.net server]# cp -R /etc/httpd/conf.d/*
↪ /vagrant/provision/server/http/etc/httpd/conf.d/
cp: target '/etc/httpd/conf.d/www.astakhovamd.net.conf' is not a directory
bash: ↪: command not found...
[root@localhost.user.net server]# cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/httpd/conf.d/
[root@localhost.user.net server]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
[root@localhost.user.net server]#
```

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

```
[root@localhost.user.net ~]# cd /vagrant/provision/server
[root@localhost.user.net server]# touch http.sh
[root@localhost.user.net server]# chmod +x http.sh
[root@localhost.user.net server]#
```

```
root@localhost:/vagrant/provision/server
root@localhost:~ × root@localhost:~ × roo
GNU nano 5.6.1 http.sh
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
```

Для отработки созданного скрипта во время загрузки виртуальных машин в конфигурационном файле Vagrantfile добавляем в конфигурации сервера следующую запись:

```
  end

  server.vm.provision "server dhcp",
                      type: "shell",
                      preserve_order: true,
                      path: "provision/server/dhcp.sh"
  end

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

4.3. Итог работы

Приобрели практические навыки по установке и базовому конфигурированию HTTP-сервера Apache.

4.4. Контрольные вопросы

Через какой порт по умолчанию работает Apache?

A. ***Порт 80 (HTTP):*** Это стандартный порт для незашифрованного веб-трафика (Hypertext Transfer Protocol). Apache слушает этот порт по умолчанию для обработки обычных HTTP-запросов.
B. ***Порт 443 (HTTPS):*** Если Apache настроен для работы с SSL/TLS (то есть для защищенного соединения HTTPS - Hypertext Transfer Protocol Secure), то он будет слушать порт 443.

Под каким пользователем запускается Apache и к какой группе относится этот пользователь?

Apache разработан для работы с минимальными привилегиями из соображений безопасности. Он запускается как root только для того, чтобы привязаться к портам ниже 1024 (например, 80 и 443), а затем переключается на непривилегированного пользователя для обработки запросов.

Имя пользователя и группы зависят от дистрибутива Linux:

1. На Debian/Ubuntu:
2. Пользователь: www-data
3. Группа: www-data
4. На RHEL/CentOS/Fedora:
5. Пользователь: apache
6. Группа: apache

Зачем это нужно?

Если злоумышленник скомпрометирует веб-сервер, работающий под непривилегированным пользователем, его возможности будут сильно ограничены. Он не сможет получить полный контроль над системой или получить доступ к файлам, принадлежащим root или другим пользователям.

Где располагаются лог-файлы веб-сервера? Что можно по ним отслеживать?

Лог-файлы Apache очень важны для мониторинга, отладки и анализа работы веб-сервера.

Расположение по умолчанию:

1. На Debian/Ubuntu: /var/log/apache2/
2. На RHEL/CentOS/Fedora: /var/log/httpd/

Примечание: для каждого VirtualHost могут быть настроены свои собственные файлы логов.

Основные типы лог-файлов и что по ним можно отслеживать:

1. ***access.log (или access_log):***

1. ***Что отслеживает:***
    1. Все входящие запросы к веб-серверу.

2. ***Содержит информацию о:***

1. IP-адрес клиента, сделавшего запрос.
2. Дата и время запроса.
3. Метод запроса (GET, POST и т.д.) и запрашиваемый URL.
4. Код состояния HTTP (200 OK, 404 Not Found, 500 Internal Server Error и т.д.).
5. Размер ответа в байтах.
6. Referer (URL, с которого пришел пользователь).
7. User-Agent (информация о браузере/ОС клиента).
8. Время обработки запроса.
    2.8.1. Для чего используется:
        2.8.1.1. Анализ трафика
        2.8.1.2. Поведенческий анализ пользователей
        2.8.1.3. Выявление проблем с доступностью контента
        2.8.1.4. Мониторинг производительности.

3. ***error.log (или error_log):***
    1. **Что отслеживает:** Ошибки, предупреждения и диагностические сообщения, возникающие на стороне сервера.
    2. **Содержит информацию о:**
        3.2.1. Внутренние ошибки сервера (например, 500 Internal Server Error).
        3.2.2. Проблемы с конфигурацией Apache.
        3.2.3. Ошибки при обработке скриптов (например, PHP-ошибки, если используется mod_php).
        3.2.4. Проблемы с правами доступа к файлам.
        3.2.5. Несуществующие файлы, к которым был запрос.
    3. **Для чего используется:**
        3.3.1. Отладка проблем сервера
        3.3.2. Выявление уязвимостей
        3.3.3. Мониторинг стабильности работы.

4. **Другие логи:** Могут быть специфические логи для SSL, перезаписи URL (mod_rewrite) и т.д, если они настроены.