

Отчёт по лабораторной работе №4

Администрирование сетевых подсистем

Ищенко Ирина НПИбд-02-22

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	17
4	Ответы на контрольные вопросы	18

Список иллюстраций

2.1	Установка веб-сервера	6
2.2	Установка веб-сервера	7
2.3	Конфигурационный файл /etc/httpd/conf/httpd.conf	7
2.4	Конфигурационный файл /etc/httpd/conf.d/autoindex.conf	8
2.5	Внесение изменений в настройки межсетевого экрана	8
2.6	Внесение изменений в настройки межсетевого экрана, запуск HTTP-сервера	8
2.7	Расширенный лог системных сообщений	9
2.8	Тестовая страницы HTTP-сервера	9
2.9	Запись в лог ошибок	10
2.10	Запись в мониторинге доступа	10
2.11	Добавление записи в конец файла прямой DNS-зоны	11
2.12	Добавление записи в конец файла обратной DNS-зоны	11
2.13	Удаление файлов журналов DNS, создание файлов	12
2.14	Редактирование server.ioithenko.net.conf	12
2.15	Редактирование www.ioithenko.net.conf	12
2.16	Создание каталога и файла главной страницы	13
2.17	Заполнение файла	13
2.18	Создание файлов	13
2.19	Заполнение файла	13
2.20	Корректирование прав доступа, восстановление контекста безопасности	13
2.21	Перезагрузка httpd	14
2.22	Доступ к server.ioithenko.net	14
2.23	Доступ к www.ioithenko.net	14
2.24	Изменения в настройки внутреннего окружения	15
2.25	Создание скрипта http.sh	15
2.26	Изменение Vagrantfile	16

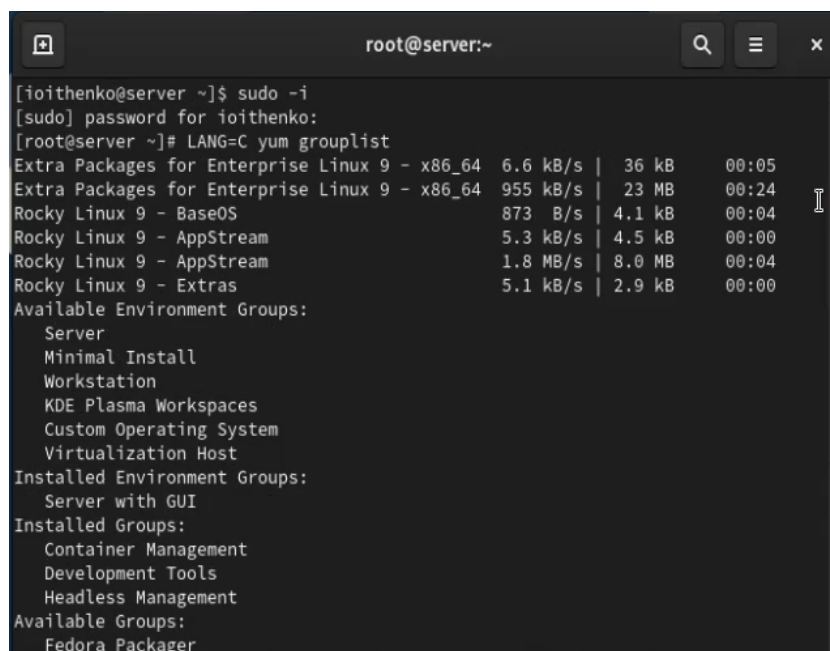
Список таблиц

1 Цель работы

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

2 Выполнение лабораторной работы

Запускаем ВМ через рабочий каталог. На ВМ server входим под собственным пользователем и переходим в режим суперпользователя. Устанавливаем стандартный веб-сервер (рис. 2.1) и (рис. 2.2).



```
root@server:~  
[ioithenko@server ~]$ sudo -i  
[sudo] password for ioithenko:  
[root@server ~]# LANG=C yum grouplist  
Extra Packages for Enterprise Linux 9 - x86_64 6.6 kB/s | 36 kB 00:05  
Extra Packages for Enterprise Linux 9 - x86_64 955 kB/s | 23 MB 00:24  
Rocky Linux 9 - BaseOS 873 B/s | 4.1 kB 00:04  
Rocky Linux 9 - AppStream 5.3 kB/s | 4.5 kB 00:00  
Rocky Linux 9 - AppStream 1.8 MB/s | 8.0 MB 00:04  
Rocky Linux 9 - Extras 5.1 kB/s | 2.9 kB 00:00  
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  
  Headless Management  
Available Groups:  
  Fedora Packager
```

Рис. 2.1: Установка веб-сервера

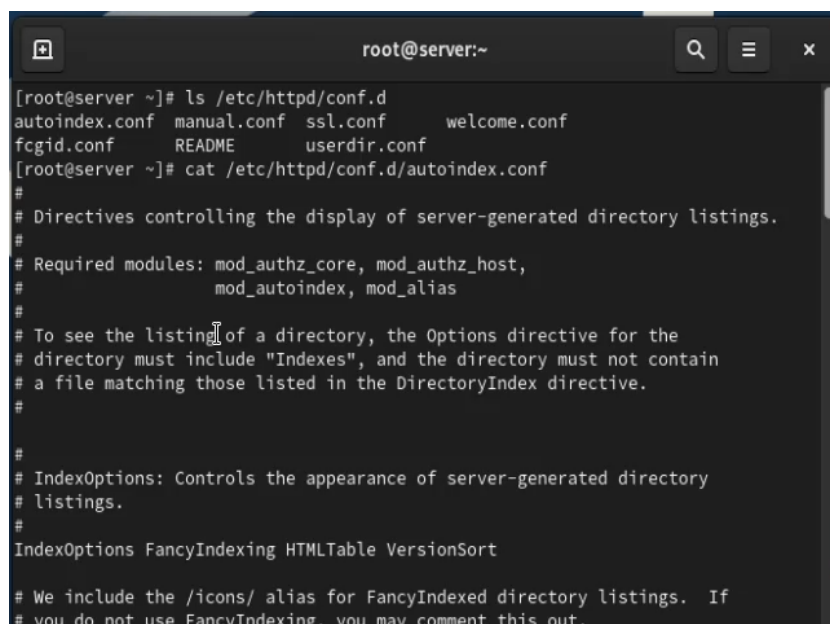
```
[root@server ~]# dnf -y groupinstall "Basic Web Server"
Last metadata expiration check: 0:00:57 ago on Wed 25 Sep 2024 11:35:58 AM UTC.
Dependencies resolved.
=====
Package                Arch      Version      Repository    Size
=====
Installing group/module packages:
httpd                  x86_64    2.4.57-11.el9_4.1    appstream     44 k
httpd-manual            noarch    2.4.57-11.el9_4.1    appstream    2.2 M
mod_fcgid               x86_64    2.3.9-28.el9         appstream     74 k
mod_ssl                x86_64    1:2.4.57-11.el9_4.1  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    2.4.57-11.el9_4.1    appstream    1.4 M
httpd-filesystem       noarch    2.4.57-11.el9_4.1    appstream     11 k
=====
```

Рис. 2.2: Установка веб-сервера

Просматриваем содержание конфигурационных файлов в каталогах /etc/httpd/conf и /etc/httpd/conf.d (рис. 2.3) и (рис. 2.4).

```
root@server:~
[root@server ~]# cat /etc/httpd/conf/httpd.conf
#
# This is the main Apache HTTP server configuration file. It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.4/> for detailed information.
# In particular, see
# <URL:http://httpd.apache.org/docs/2.4/mod/directives.html>
# for a discussion of each configuration directive.
#
# See the httpd.conf(5) man page for more information on this configuration,
# and httpd.service(8) on using and configuring the httpd service.
#
# Do NOT simply read the instructions in here without understanding
# what they do. They're here only as hints or reminders. If you are unsure
# consult the online docs. You have been warned.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32), the
# server will use that explicit path. If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so 'log/access_log'
# with ServerRoot set to '/www' will be interpreted by the
# server as '/www/log/access_log', where as '/log/access_log' will be
# interpreted as '/log/access_log'.
```

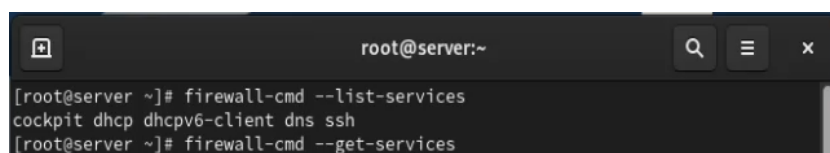
Рис. 2.3: Конфигурационный файл /etc/httpd/conf/httpd.conf



```
root@server:~  
[root@server ~]# ls /etc/httpd/conf.d  
autoindex.conf  manual.conf  ssl.conf      welcome.conf  
fcgid.conf      README      userdir.conf  
[root@server ~]# cat /etc/httpd/conf.d/autoindex.conf  
#  
# Directives controlling the display of server-generated directory listings.  
#  
# Required modules: mod_authz_core, mod_authz_host,  
#                  mod_autoindex, mod_alias  
#  
# To see the listing of a directory, the Options directive for the  
# directory must include "Indexes", and the directory must not contain  
# a file matching those listed in the DirectoryIndex directive.  
#  
#  
# IndexOptions: Controls the appearance of server-generated directory  
# listings.  
#  
IndexOptions FancyIndexing HTMLTable VersionSort  
  
# We include the /icons/ alias for FancyIndexed directory listings. If  
# you do not use FancyIndexing, you may comment this out.
```

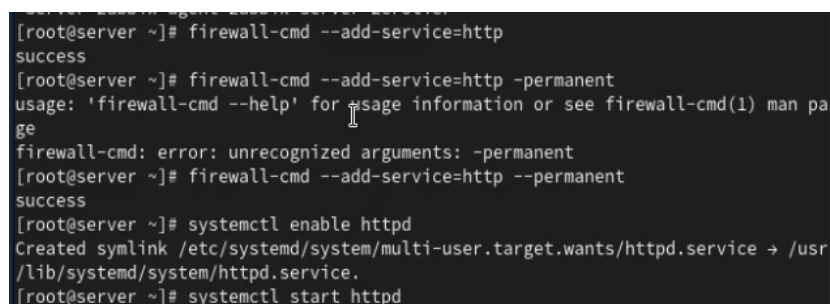
Рис. 2.4: Конфигурационный файл /etc/httpd/conf.d/autoindex.conf

Вносим изменения в настройки межсетевого экрана узла server, разрешив работу с http (рис. 2.5). Во втором терминале открываем расширенный лог системных сообщений. В первом терминале активируем и запускаем HTTP-сервер (рис. 2.6).



```
root@server:~  
[root@server ~]# firewall-cmd --list-services  
cockpit dhcp dhcpv6-client dns ssh  
[root@server ~]# firewall-cmd --get-services
```

Рис. 2.5: Внесение изменений в настройки межсетевого экрана



```
[root@server ~]# firewall-cmd --add-service=http  
success  
[root@server ~]# firewall-cmd --add-service=http --permanent  
usage: 'firewall-cmd --help' for usage information or see firewall-cmd(1) man page  
firewall-cmd: error: unrecognized arguments: --permanent  
[root@server ~]# firewall-cmd --add-service=http --permanent  
success  
[root@server ~]# systemctl enable httpd  
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.  
[root@server ~]# systemctl start httpd
```

Рис. 2.6: Внесение изменений в настройки межсетевого экрана, запуск HTTP-сервера

В логе системных сообщений вижу, что веб-сервер запущен (рис. 2.7)

```
Sep 25 11:49:32 server systemd[1]: Starting The Apache HTTP Server...
Subject: A start job for unit httpd.service has begun execution
Defined-By: systemd
Support: https://wiki.rockylinux.org/rocky/support

A start job for unit httpd.service has begun execution.

The job identifier is 2966.
Sep 25 11:49:32 server httpd[6651]: AH00558: httpd: Could not reliably determine
the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName'
directive globally to suppress this message
Sep 25 11:49:32 server systemd[1]: Started The Apache HTTP Server.
Subject: A start job for unit httpd.service has finished successfully
Defined-By: systemd
Support: https://wiki.rockylinux.org/rocky/support

A start job for unit httpd.service has finished successfully.

The job identifier is 2966.
Sep 25 11:49:32 server httpd[6651]: Server configured, listening on: port 443, p
ort 80
```

Рис. 2.7: Расширенный лог системных сообщений

На виртуальной машине server открываем лог ошибок работы веб-сервера: `tail -f /var/log/httpd/error_log`; и мониторинг доступа к веб-серверу: `tail -f /var/log/httpd/access_log`.

Запускаем VM client, открываем браузер и в адресной строке вводим 192.168.1.1 (рис. 2.8)

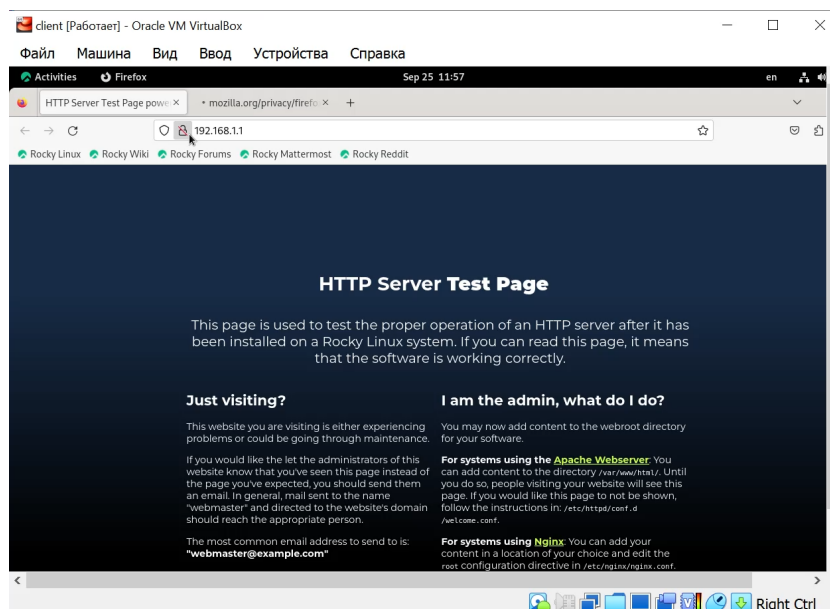


Рис. 2.8: Тестовая страницы HTTP-сервера

Видим записи в логе ошибок (рис. 2.9) и в мониторинге доступа о подключении к серверу (рис. 2.10).

```
[root@server ~]# tail -f /var/log/httpd/error_log
[Wed Sep 25 11:49:32.661878 2024] [core:notice] [pid 6651:tid 6651] SELinux policy enabled; httpd running as context system_u:system_r:httpd_t:s0
[Wed Sep 25 11:49:32.675102 2024] [suexec:notice] [pid 6651:tid 6651] AH01232: suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
[Wed Sep 25 11:49:32.756154 2024] [lbmethod_heartbeat:notice] [pid 6651:tid 6651] AH02282: No slotmem from mod_heartbeat
[Wed Sep 25 11:49:32.807594 2024] [mpm_event:notice] [pid 6651:tid 6651] AH00489: Apache/2.4.57 (Rocky Linux) OpenSSL/3.0.7 mod_fcgid/2.3.9 configured -- resuming normal operations
[Wed Sep 25 11:49:32.810432 2024] [core:notice] [pid 6651:tid 6651] AH00094: Command line: '/usr/sbin/httpd -D FOREGROUND'
[Wed Sep 25 11:57:50.173418 2024] [autoindex:error] [pid 6656:tid 6778] [client 192.168.1.30:37102] AH01276: Cannot serve directory /var/www/html/: No matching DirectoryIndex (index.html) found, and server-generated directory index forbidden by Options directive
```

Рис. 2.9: Запись в логе ошибок

```
[root@server ~]# tail -f /var/log/httpd/access_log
192.168.1.30 - - [25/Sep/2024:11:57:50 +0000] "GET / HTTP/1.1" 403 7620 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
192.168.1.30 - - [25/Sep/2024:11:57:54 +0000] "GET /icons/poweredby.png HTTP/1.1" 200 15443 "http://192.168.1.1/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
192.168.1.30 - - [25/Sep/2024:11:57:54 +0000] "GET /poweredby.png HTTP/1.1" 200 5714 "http://192.168.1.1/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
192.168.1.30 - - [25/Sep/2024:11:57:54 +0000] "GET /favicon.ico HTTP/1.1" 404 196 "http://192.168.1.1/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
```

Рис. 2.10: Запись в мониторинге доступа

Останавливаем работу DNS-сервера. Добавляем запись для HTTP-сервера в конце файла прямой DNS-зоны (рис. 2.11).

```

root@server:~
root@serv... x ioithenko@... x root@serv... x root@serv... x
$ORIGIN .
$TTL 86400 ; 1 day
ioithenko.net IN SOA ioithenko.net. server.ioithenko.net. (
    2024091908 ; serial
    86400 ; refresh (1 day)
    3600 ; retry (1 hour)
    604800 ; expire (1 week)
    10800 ; minimum (3 hours)
)
NS ioithenko.net.
A 192.168.1.1

$ORIGIN ioithenko.net.
$TTL 300 ; 5 minutes
client A 192.168.1.30
TXT "31bf245e3ac05a2542837a17eb06320c06"

$TTL 86400 ; 1 day
dhcp A 192.168.1.1
ns A 192.168.1.1
server A 192.168.1.1
www A 192.1

~
~
-- INSERT --
20,14-38 All

```

Рис. 2.11: Добавление записи в конец файла прямой DNS-зоны

Добавляем также запись в конце файла обратной зоны (рис. 2.12)

```

root@server:~
root@serv... x ioithenko@... x root@serv... x root@serv... x
$TTL 86400 ; 1 day
1.168.192.in-addr.arpa IN SOA 1.168.192.in-addr.arpa. server.ioithenko.net. (
                                2024091906 ; serial
                                86400 ; refresh (1 day)
                                3600 ; retry (1 hour)
                                604800 ; expire (1 week)
                                10800 ; minimum (3 hours)
                                )
                                NS 1.168.192.in-addr.arpa.
                                A 192.168.1.1
                                PTR server.ioithenko.net.
$ORIGIN 1.168.192.in-addr.arpa.
1 PTR server.ioithenko.net.
PTR ns.ioithenko.net.
PTR dhcp.ioithenko.net.
$TTL 300 ; 5 minutes
30 PTR client.ioithenko.net.
1 PTR www.ioithenko.net.

```

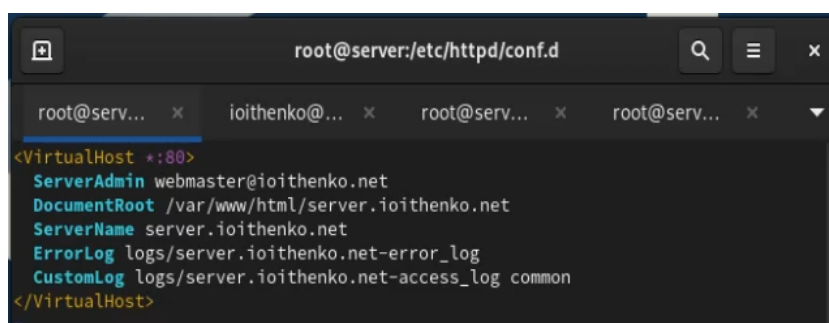
Рис. 2.12: Добавление записи в конец файла обратной DNS-зоны

Удаляем из соответствующих каталогов файлы журналов DNS. Перезапускаем DNS-сервер. В каталоге /etc/httpd/conf.d создаю файлы server.ioithenko.net.conf и www.ioithenko.net.conf (рис. 2.13).

```
[root@server ~]# systemctl stop named
[root@server ~]# vim /var/named/master/fz/ioithenko.net
[root@server ~]# vim /var/named/master/rz/192.168.1
[root@server ~]# rm /var/named/master/fz/ioithenko.net.jnl
rm: remove regular file '/var/named/master/fz/ioithenko.net.jnl'? y
[root@server ~]# rm /var/named/master/rz/192.168.1.jnl
rm: remove regular file '/var/named/master/rz/192.168.1.jnl'? y
[root@server ~]# systemctl start named
[root@server ~]# cd /etc/httpd/conf.d
[root@server conf.d]# touch server.ioithenko.net.conf
```

Рис. 2.13: Удаление файлов журналов DNS, создание файлов

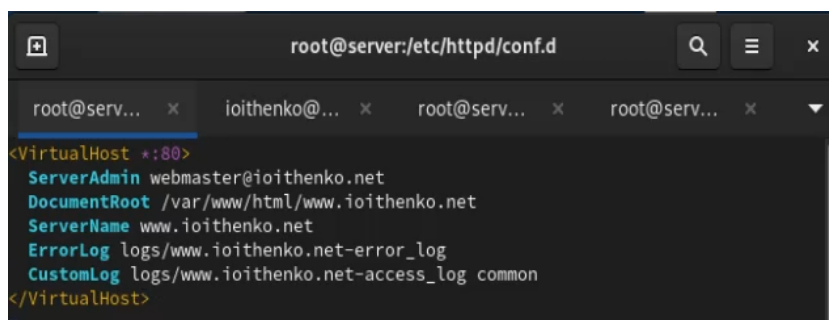
Редактируем server.ioithenko.net.conf (рис. 2.14).



```
root@server:/etc/httpd/conf.d
<VirtualHost *:80>
  ServerAdmin webmaster@ioithenko.net
  DocumentRoot /var/www/html/server.ioithenko.net
  ServerName server.ioithenko.net
  ErrorLog logs/server.ioithenko.net-error_log
  CustomLog logs/server.ioithenko.net-access_log common
</VirtualHost>
```

Рис. 2.14: Редактирование server.ioithenko.net.conf

Редактируем www.ioithenko.net.conf (рис. 2.15).



```
root@server:/etc/httpd/conf.d
<VirtualHost *:80>
  ServerAdmin webmaster@ioithenko.net
  DocumentRoot /var/www/html/www.ioithenko.net
  ServerName www.ioithenko.net
  ErrorLog logs/www.ioithenko.net-error_log
  CustomLog logs/www.ioithenko.net-access_log common
</VirtualHost>
```

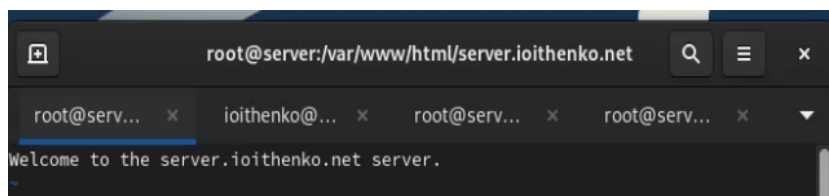
Рис. 2.15: Редактирование www.ioithenko.net.conf

Переходим в /var/www/html создаем каталог server.ioithenko.net и в нем файл index.html (рис. 2.16). Вводим в файл сообщение Welcome to

the server.ioithenko.net server. Аналогично создаем каталог и файл для www.ioithenko.net.

```
[root@server conf.d]# cd /var/www/html
[root@server html]# mkdir server.ioithenko.net
```

Рис. 2.16: Создание каталога и файла главной страницы

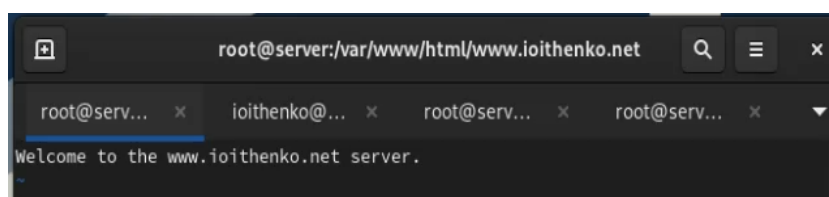


The screenshot shows a terminal window with the title bar 'root@server:/var/www/html/server.ioithenko.net'. The terminal output shows the command 'touch index.html' being executed, and the message 'Welcome to the server.ioithenko.net server.' is displayed at the bottom.

Рис. 2.17: Заполнение файла

```
[root@server html]# mkdir www.ioithenko.net
[root@server html]# cd /var/www/html/www.ioithenko.net
[root@server www.ioithenko.net]# touch index.html
[root@server www.ioithenko.net]# vim index.html
```

Рис. 2.18: Создание файлов



The screenshot shows a terminal window with the title bar 'root@server:/var/www/html/www.ioithenko.net'. The terminal output shows the command 'touch index.html' being executed, and the message 'Welcome to the www.ioithenko.net server.' is displayed at the bottom.

Рис. 2.19: Заполнение файла

Корректируем права доступа в каталог с веб-контентом, восстанавливаем контекст безопасности в SELinux, перезагружаем HTTP-сервер (рис. 2.20) и (рис. 2.21).

```
[root@server www.ioithenko.net]# chown -R apache:apache /var/www
[root@server www.ioithenko.net]# restorecon -vR /etc
[root@server www.ioithenko.net]# restorecon -vR /var/named
[root@server www.ioithenko.net]# restorecon -vR /var/www
[root@server www.ioithenko.net]# systemctl restart httpd
```

Рис. 2.20: Корректирование прав доступа, восстановление контекста безопасности

```
[root@server html]# systemctl restart httpd
[root@server html]#
```

Рис. 2.21: Перезагрузка httpd

На VM client вводим в адресную строку браузера `server.ioithenko.net` (рис. 2.22) и `www.ioithenko.net` (рис. 2.23).

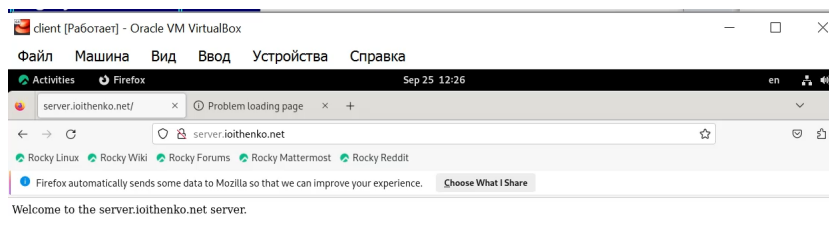


Рис. 2.22: Доступ к `server.ioithenko.net`

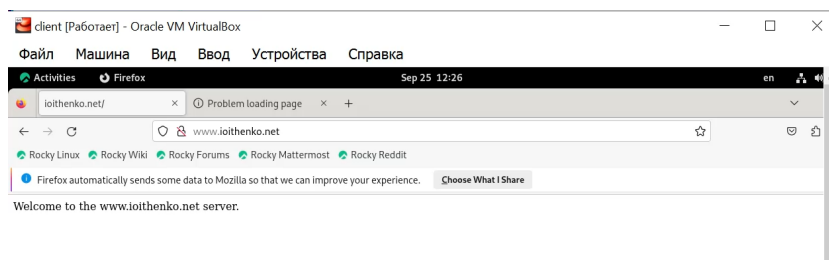


Рис. 2.23: Доступ к `www.ioithenko.net`

На VM server переходим в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/`, создаем в нём каталог `http`, в который помещаем в соответствующие подкаталоги конфигурационные файлы HTTP-сервера. Заменяем конфигурационные файлы DNS-сервера (рис. 2.24).

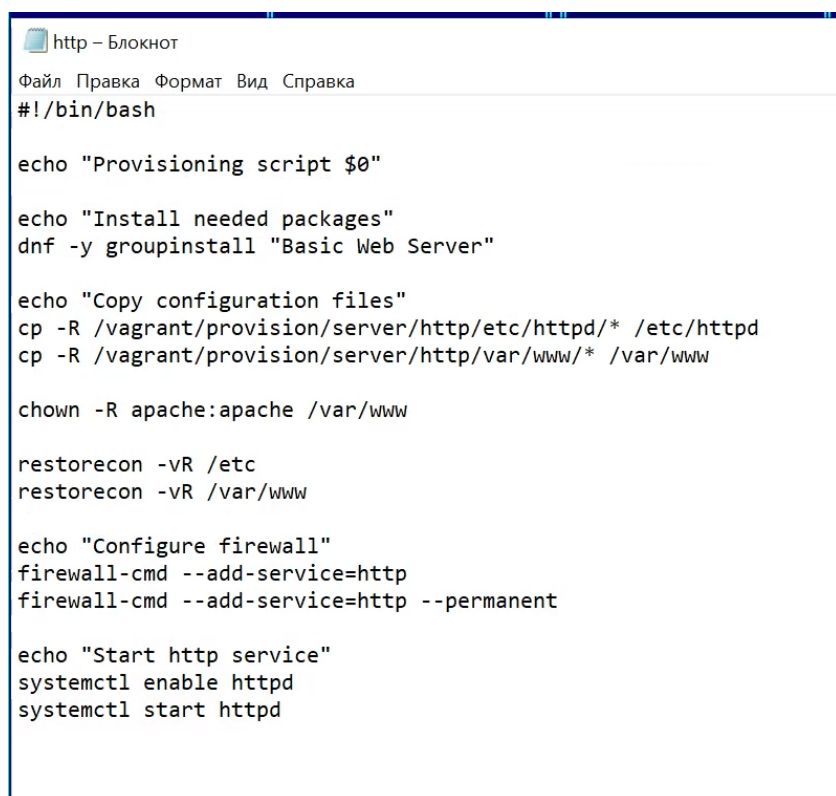
```

[root@server html]# cd /vagrant/provision/server
[root@server server]# mkdir /vagrant/provision/server/http/etc/httpd/conf.d
mkdir: cannot create directory '/vagrant/provision/server/http/etc/httpd/conf.d'
: No such file or directory
[root@server server]# mkdir -p /vagrant/provision/server/http/etc/httpd/conf.d
[root@server server]# cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/e
tc/httpd/conf.d/
[root@server server]# mkdir -p /vagrant/provision/server/http/var/www/html
[root@server server]# cp -R /var/www/html/* /vagrant/provision/server/http/var/w
ww/html
[root@server server]# cd /server/dns
-bash: cd: /server/dns: No such file or directory
[root@server server]# cd /dns
-bash: cd: /dns: No such file or directory
[root@server server]# cd /vagrant/provision/server/dns/
[root@server dns]# cp /var/named/* /vagrant/provision/server/dns/var/named/

```

Рис. 2.24: Изменения в настройке внутреннего окружения

В каталоге `/vagrant/provision/server` создаем исполняемый файл `http.sh` (рис. 2.25).



```

http - Блокнот
Файл Правка Формат Вид Справка
#!/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

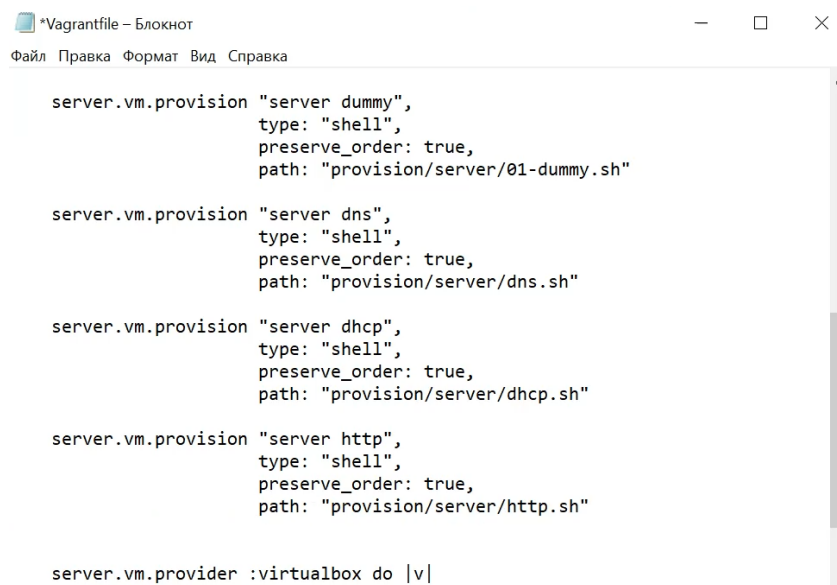
echo "Configure firewall"
firewall-cmd --add-service=http
firewall-cmd --add-service=http --permanent

echo "Start http service"
systemctl enable httpd
systemctl start httpd

```

Рис. 2.25: Создание скрипта `http.sh`

Для отработки скрипта во время запуска добавляем в `Vagrantfile` в разделе конфигурации для сервера (рис. 2.26).



The image shows a screenshot of a text editor window titled "*Vagrantfile – Блокнот". The menu bar includes "Файл", "Правка", "Формат", "Вид", and "Справка". The editor contains the following Vagrantfile configuration:

```
server.vm.provision "server dummy",
  type: "shell",
  preserve_order: true,
  path: "provision/server/01-dummy.sh"

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

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

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

server.vm.provider :virtualbox do |v|
```

Рис. 2.26: Изменение Vagrantfile

3 Выводы

В ходе выполнения работы я приобрела практические навыки по установке и базовому конфигурированию HTTP-сервера Apache.

4 Ответы на контрольные вопросы

1. Через какой порт по умолчанию работает Apache?

По умолчанию Apache работает через порт 80 для HTTP и порт 443 для HTTPS.

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

Apache обычно запускается от имени пользователя `www-data` (или `apache`, в зависимости от дистрибутива) и относится к группе с тем же именем.

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

Лог-файлы веб-сервера обычно располагаются в директории логов. Например, в `/etc/httpd/logs/`. Лог-файлы содержат информацию о запросах к серверу, ошибки, статусы запросов и другие события, что позволяет администраторам отслеживать активность и выявлять проблемы.

4. Где по умолчанию содержится контент веб-серверов?

Контент веб-серверов по умолчанию обычно находится в директории, называемой “DocumentRoot”. Например, в Apache на Linux DocumentRoot по умолчанию установлен в `/var/www/html/`. В этой директории содержатся файлы, которые веб-сервер отдает при запросах.

5. Каким образом реализуется виртуальный хостинг? Что он даёт?

Виртуальный хостинг в Apache позволяет хостить несколько сайтов на одном сервере. Разные сайты обслуживаются на одном IP-адресе, но на разных доменных именах. Это основывается на значении заголовка “Host” в HTTP-запросе, который используется для определения, какой виртуальный хост должен обработать запрос. Виртуальный хостинг позволяет хозяину сервера размещать несколько сайтов на одном физическом сервере, управлять ими независимо, и предоставлять услуги хостинга для различных клиентов или проектов.