

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра теории вероятностей и кибербезопасности

ОТЧЁТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №4

дисциплина: Администрирование сетевых подсистем

Студент: Махорин Иван Сергеевич

Студ. билет № 1032211221

Группа: НПИбд-02-21

МОСКВА

2023 г.

Цель работы:

Целью данной работы является приобретение практических навыков по установке и базовому конфигурированию HTTP-сервера Apache.

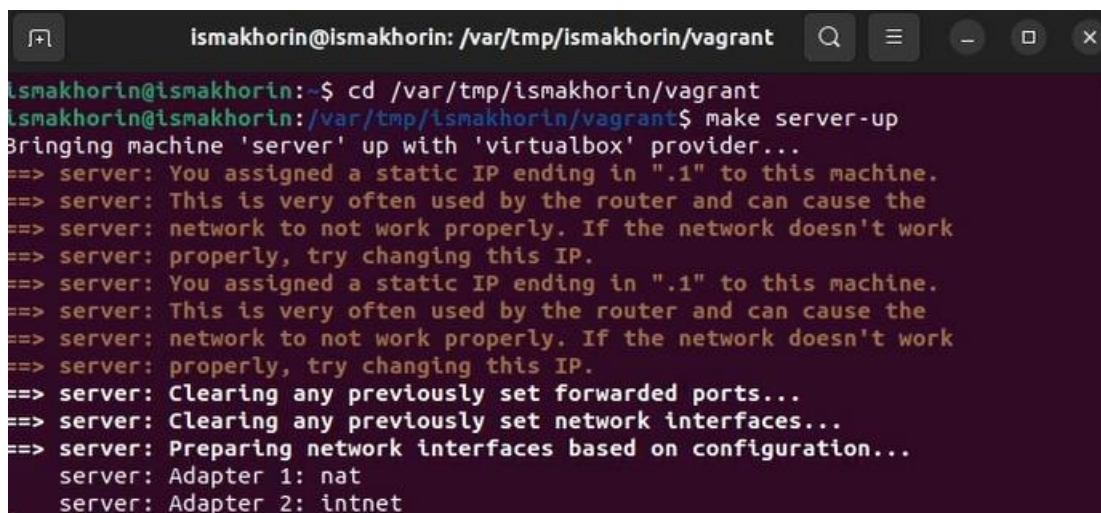
Выполнение работы:

Загрузим нашу операционную систему и перейдём в рабочий каталог с проектом:

```
cd /var/tmp/ismakhorin/vagrant
```

Далее запустим виртуальную машину server (Рис. 1.1):

```
make server-up
```



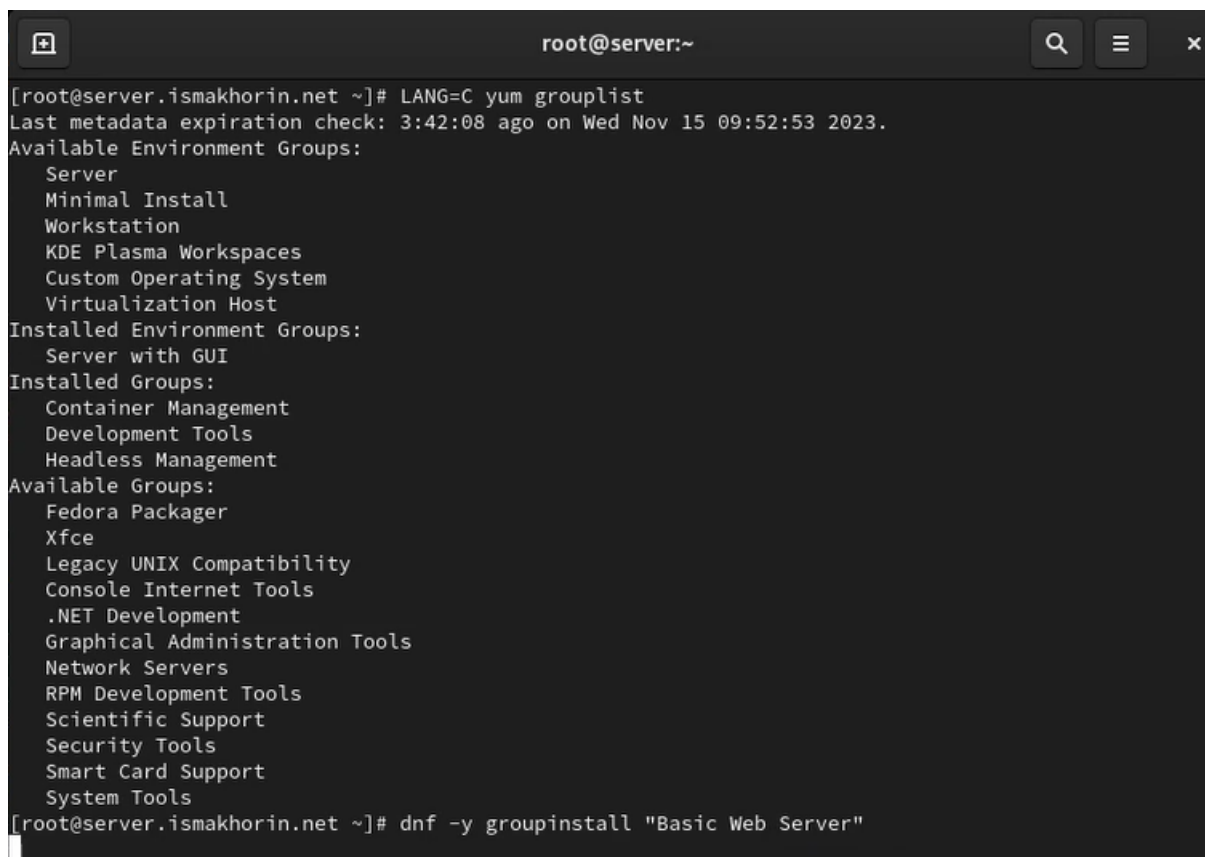
```
ismakhorin@ismakhorin: /var/tmp/ismakhorin/vagrant
ismakhorin@ismakhorin:~$ cd /var/tmp/ismakhorin/vagrant
ismakhorin@ismakhorin:/var/tmp/ismakhorin/vagrant$ make server-up
Bringing machine 'server' up with 'virtualbox' provider...
==> server: You assigned a static IP ending in ".1" to this machine.
==> server: This is very often used by the router and can cause the
==> server: network to not work properly. If the network doesn't work
==> server: properly, try changing this IP.
==> server: You assigned a static IP ending in ".1" to this machine.
==> server: This is very often used by the router and can cause the
==> server: network to not work properly. If the network doesn't work
==> server: properly, try changing this IP.
==> server: Clearing any previously set forwarded ports...
==> server: Clearing any previously set network interfaces...
==> server: Preparing network interfaces based on configuration...
server: Adapter 1: nat
server: Adapter 2: intnet
```

Рис. 1.1. Открытие рабочего каталога с проектом и запуск виртуальной машины server.

На виртуальной машине server войдём под нашим пользователем и откроем терминал. Далее перейдём в режим суперпользователя и установим из репозитория стандартный веб-сервер (HTTP-сервер и утилиты httpd, криптоутилиты и пр.) (Рис. 1.2):

```
LANG=C yum grouplist
```

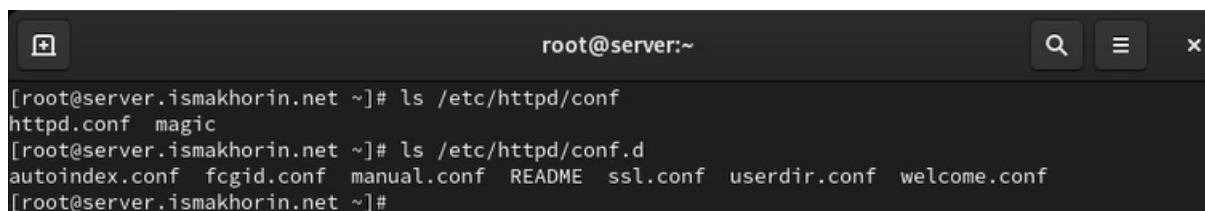
```
dnf -y groupinstall "Basic Web Server"
```



```
root@server:~  
[root@server.ismakhorin.net ~]# LANG=C yum grouplist  
Last metadata expiration check: 3:42:08 ago on Wed Nov 15 09:52:53 2023.  
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  
  Xfce  
  Legacy UNIX Compatibility  
  Console Internet Tools  
  .NET Development  
  Graphical Administration Tools  
  Network Servers  
  RPM Development Tools  
  Scientific Support  
  Security Tools  
  Smart Card Support  
  System Tools  
[root@server.ismakhorin.net ~]# dnf -y groupinstall "Basic Web Server"
```

Рис. 1.2. Переход в режим суперпользователя и установка из репозитория стандартного веб-сервера.

Посмотрим какие конфигурационные файлы есть в каталогах /etc/httpd/conf и /etc/httpd/conf.d (Рис. 2.1):



```
root@server:~  
[root@server.ismakhorin.net ~]# ls /etc/httpd/conf  
httpd.conf  magic  
[root@server.ismakhorin.net ~]# ls /etc/httpd/conf.d  
autoindex.conf  fcgid.conf  manual.conf  README  ssl.conf  userdir.conf  welcome.conf  
[root@server.ismakhorin.net ~]#
```

Рис. 2.1. Просмотр конфигурационных файлов в каталогах /etc/httpd/conf и /etc/httpd/conf.d.

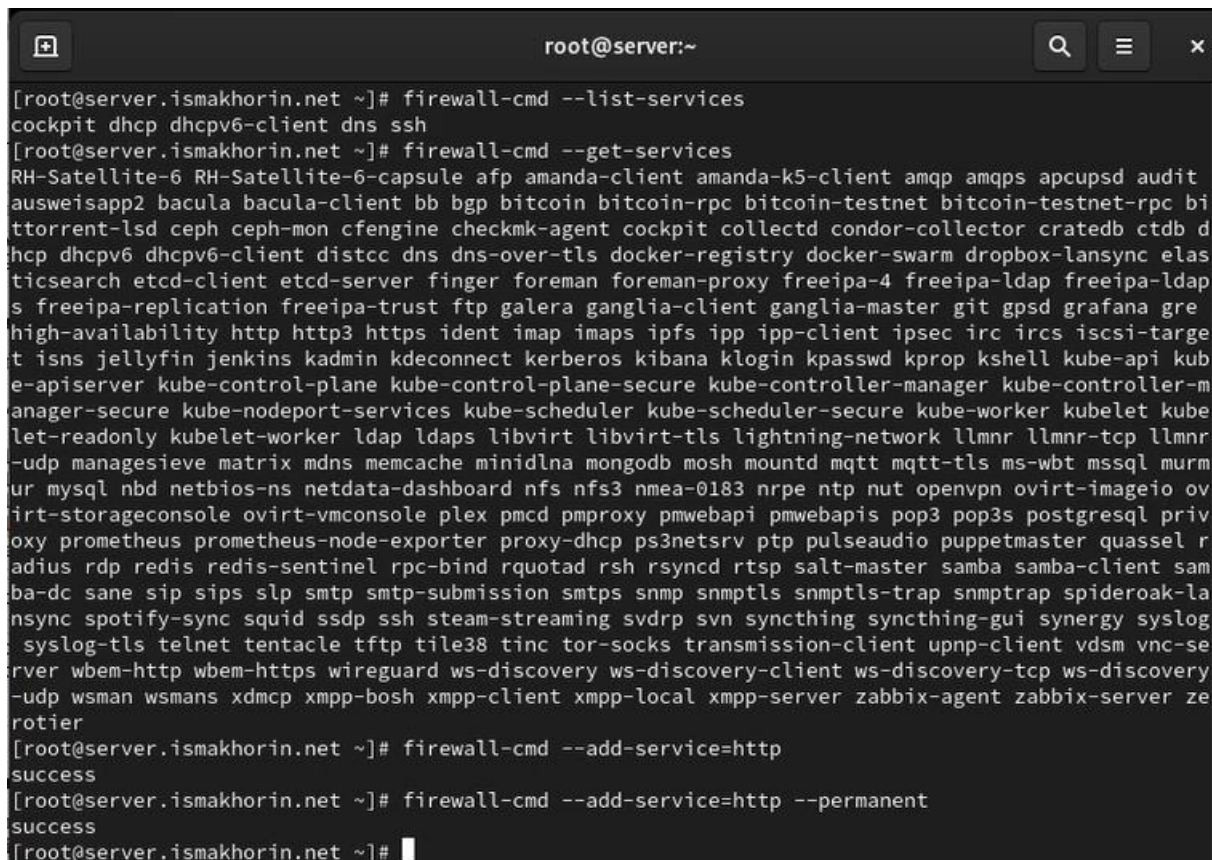
Внесём изменения в настройки межсетевого экрана узла server, разрешив работу с http:

```
firewall-cmd --list-services
```

```
firewall-cmd --get-services
```

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

```
firewall-cmd --add-service=http --permanent (Рис. 2.2).
```

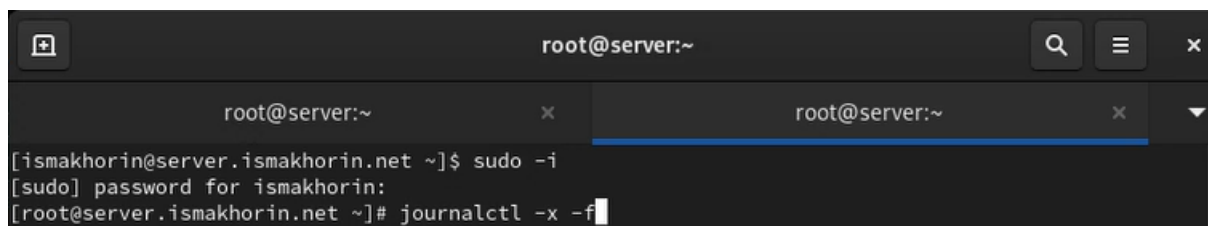


```
root@server:~  
[root@server.ismakhorin.net ~]# firewall-cmd --list-services  
cockpit dhcp dhcpv6-client dns ssh  
[root@server.ismakhorin.net ~]# firewall-cmd --get-services  
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp amqps apcupsd audit  
ausweisapp2 bacula bacula-client bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rpc bi  
ttorrent-lsd ceph ceph-mon cfengine checkmk-agent cockpit collectd condor-collector cratedb ctdb d  
hcp dhcpv6 dhcpv6-client distcc dns dns-over-tls docker-registry docker-swarm dropbox-lansync elas  
ticsearch etcd-client etcd-server finger foreman foreman-proxy freeipa-4 freeipa-ldap freeipa-ldap  
s freeipa-replication freeipa-trust ftp galera ganglia-client ganglia-master git gpsd grafana gre  
high-availability http http3 https ident imap imaps ipfs ipp ipp-client ipsec irc ircs iscsi-targe  
t isns jellyfin jenkins kadmin kdeconnect kerberos kibana klogin kpasswd kprop kshell kube-api kub  
e-apiserver kube-control-plane kube-control-plane-secure kube-controller-manager kube-controller-m  
anager-secure kube-nodeport-services kube-scheduler kube-scheduler-secure kube-worker kubelet kube  
let-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr llmnr-tcp llmnr  
-udp managesieve matrix mdns memcache minidlna mongodb mosh mountd mqtt mqtt-tls ms-wbt mssql murm  
ur mysql nbd netbios-ns netdata-dashboard nfs nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ov  
irt-storageconsole ovirt-vmconsole plex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresql priv  
oxy prometheus prometheus-node-exporter proxy-dhcp ps3netsrv ptp pulseaudio puppetmaster quassel r  
adius rdp redis redis-sentinel rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba-client sam  
ba-dc sane sip sips slp smtp smtp-submission smtps snmp snmptls snmptls-trap snmptrap spideroak-la  
nsync spotify-sync squid ssdp steam-streaming svdrp svn syncthing syncthing-gui synergy syslog  
syslog-tls telnet tentacle tftp tile38 tinc tor-socks transmission-client upnp-client vdsm vnc-se  
rver wbem-http wbem-https wireguard ws-discovery ws-discovery-client ws-discovery-tcp ws-discovery  
-udp wsman wsmans xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-server ze  
rotier  
[root@server.ismakhorin.net ~]# firewall-cmd --add-service=http  
success  
[root@server.ismakhorin.net ~]# firewall-cmd --add-service=http --permanent  
success  
[root@server.ismakhorin.net ~]#
```

Рис. 2.2. Внесение изменений в настройки межсетевого экрана узла
server, разрешив работу с http.

В дополнительном терминале запустим в режиме реального времени расширенный лог системных сообщений, чтобы проверить корректность работы системы (Рис. 2.3):

```
journalctl -x -f
```



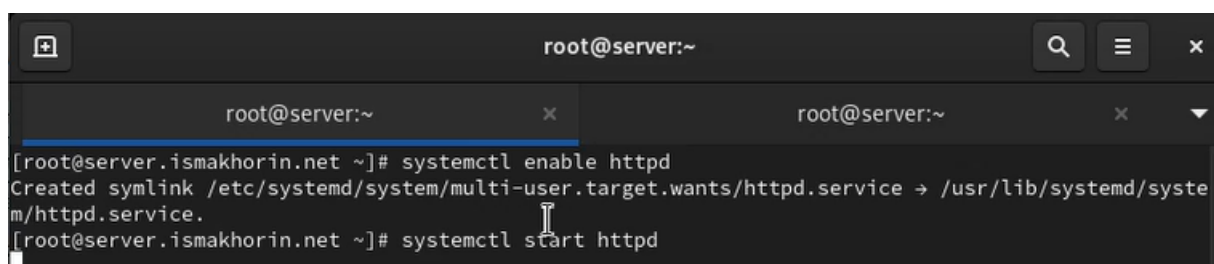
```
root@server:~  
[ismakhorin@server.ismakhorin.net ~]$ sudo -i  
[sudo] password for ismakhorin:  
[root@server.ismakhorin.net ~]# journalctl -x -f
```

Рис. 2.3. Запуск в дополнительном терминале в режиме реального времени расширенного лога системных сообщений для проверки корректности работы системы.

В первом терминале активируем и запустим HTTP-сервер (Рис. 2.4):

```
systemctl enable httpd
```

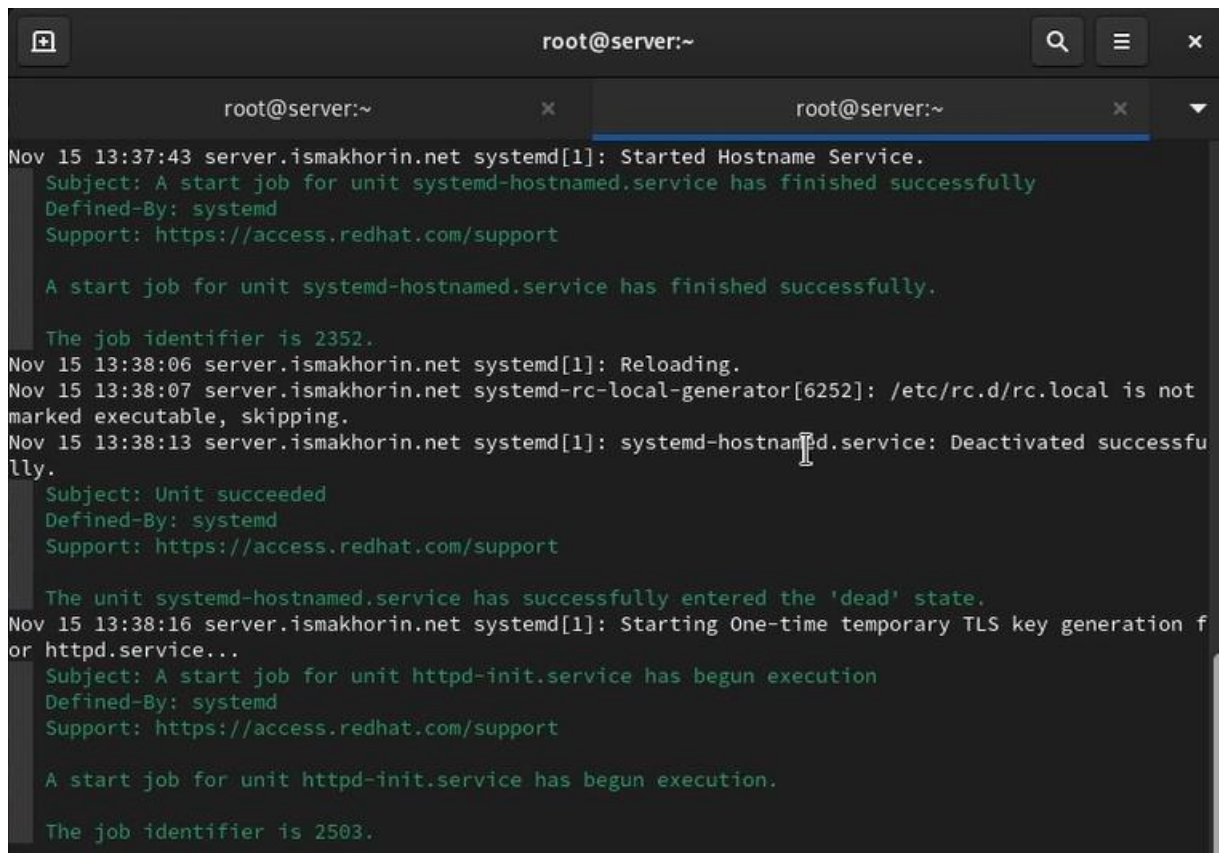
```
systemctl start httpd
```



```
root@server:~  
[root@server.ismakhorin.net ~]# systemctl enable httpd  
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.  
[root@server.ismakhorin.net ~]# systemctl start httpd
```

Рис. 2.4. Активация и запуск HTTP-сервера.

Просмотрим расширенный лог системных сообщений, убедимся, что веб-сервер успешно запустился (Рис. 2.5):



```
root@server:~
Nov 15 13:37:43 server.ismakhorin.net systemd[1]: Started Hostname Service.
Subject: A start job for unit systemd-hostnamed.service has finished successfully
Defined-By: systemd
Support: https://access.redhat.com/support

A start job for unit systemd-hostnamed.service has finished successfully.

The job identifier is 2352.
Nov 15 13:38:06 server.ismakhorin.net systemd[1]: Reloading.
Nov 15 13:38:07 server.ismakhorin.net systemd-rc-local-generator[6252]: /etc/rc.d/rc.local is not
marked executable, skipping.
Nov 15 13:38:13 server.ismakhorin.net systemd[1]: systemd-hostnamed.service: Deactivated successfully.
Subject: Unit succeeded
Defined-By: systemd
Support: https://access.redhat.com/support

The unit systemd-hostnamed.service has successfully entered the 'dead' state.
Nov 15 13:38:16 server.ismakhorin.net systemd[1]: Starting One-time temporary TLS key generation for httpd.service...
Subject: A start job for unit httpd-init.service has begun execution
Defined-By: systemd
Support: https://access.redhat.com/support

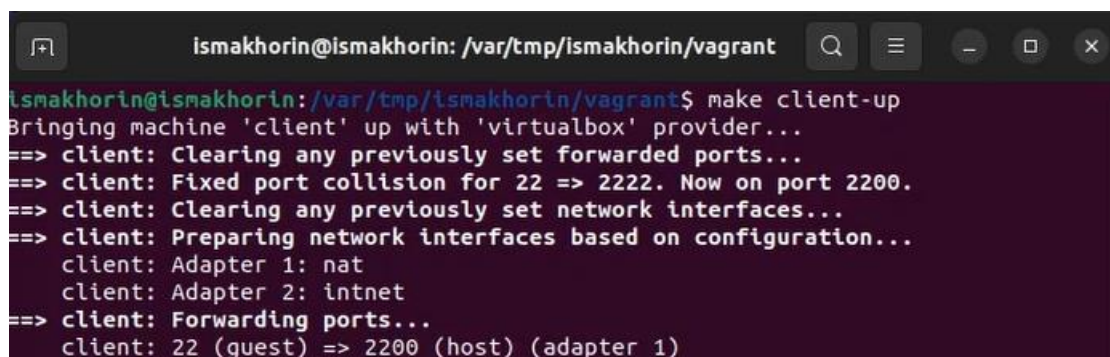
A start job for unit httpd-init.service has begun execution.

The job identifier is 2503.
```

Рис. 2.5. Просмотр расширенного лога системных сообщений.

Запустим виртуальную машину client (Рис. 3.1):

make client-up



```
ismakhorin@ismakhorin: /var/tmp/ismakhorin/vagrant
ismakhorin@ismakhorin:/var/tmp/ismakhorin/vagrant$ make client-up
Bringing machine 'client' up with 'virtualbox' provider...
==> client: Clearing any previously set forwarded ports...
==> client: Fixed port collision for 22 => 2222. Now on port 2200.
==> client: Clearing any previously set network interfaces...
==> client: Preparing network interfaces based on configuration...
client: Adapter 1: nat
client: Adapter 2: intnet
==> client: Forwarding ports...
client: 22 (guest) => 2200 (host) (adapter 1)
```

Рис. 3.1. Запуск виртуальной машины client.

На виртуальной машине server посмотрим лог ошибок работы веб-сервера:

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

Далее запустим мониторинг доступа к веб-серверу (Рис. 3.2):

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



```
[root@server.ismakhorin.net ~]# tail -f /var/log/httpd/access_log
192.168.1.30 - - [15/Nov/2023:13:50:44 +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 - - [15/Nov/2023:13:50:45 +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 - - [15/Nov/2023:13:50:45 +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 - - [15/Nov/2023:13:50:46 +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. 3.2. Запуск мониторинга доступа к веб-серверу на виртуальной машине server.

На виртуальной машине client запустим браузер и в адресной строке введём 192.168.1.1 (Рис. 3.3):

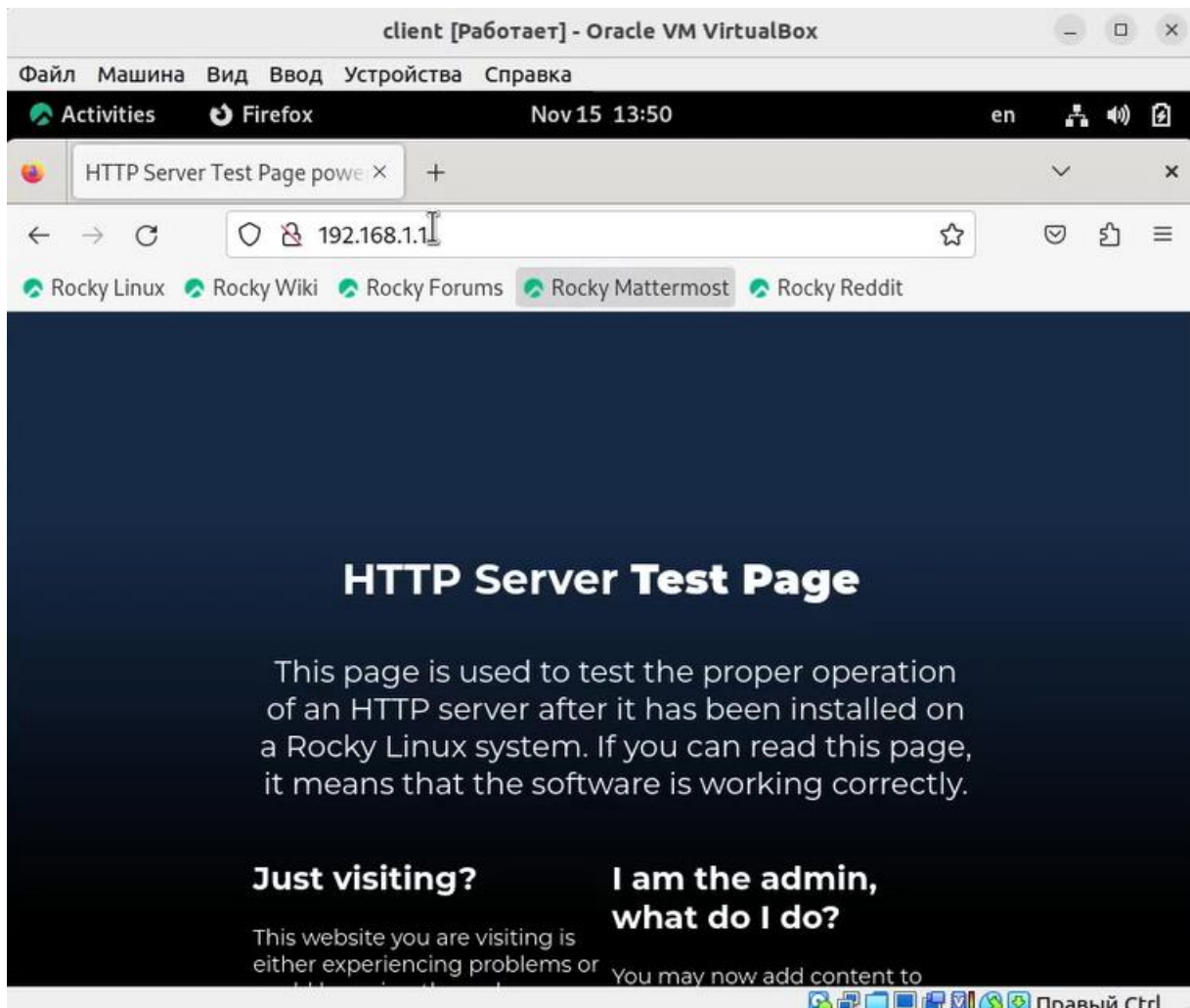


Рис. 3.3. Запуск браузера на виртуальной машине client и ввод в адресной строке 192.168.1.1.

Остановим работу DNS-сервера для внесения изменений в файлы описания DNS-зон (рис. 4.1):

```
systemctl stop named
```

```
[root@server.ismakhorin.net ~]# systemctl stop named  
[root@server.ismakhorin.net ~]#
```

Рис. 4.1. Остановка работы DNS-сервера для внесения изменений в файлы описания DNS-зон.

Теперь добавим запись для HTTP-сервера в конце файла прямой DNS-зоны /var/named/master/fz/ismakhorin.net (рис. 4.2):

Server	A	192.168.1.1
www	A	192.168.1.1

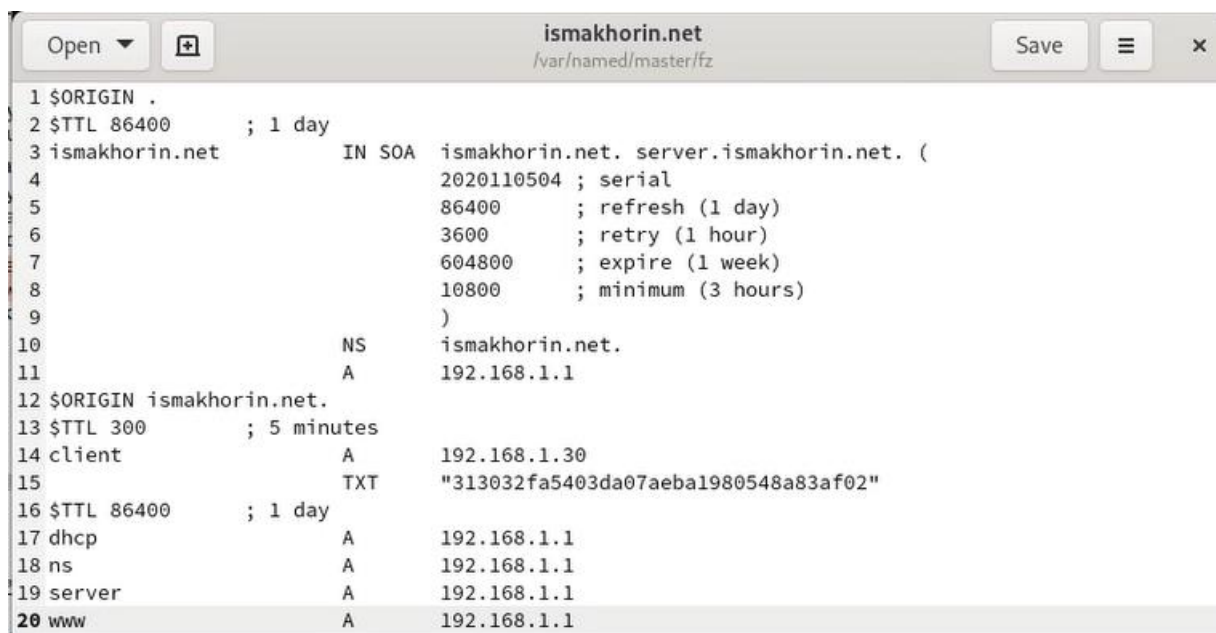
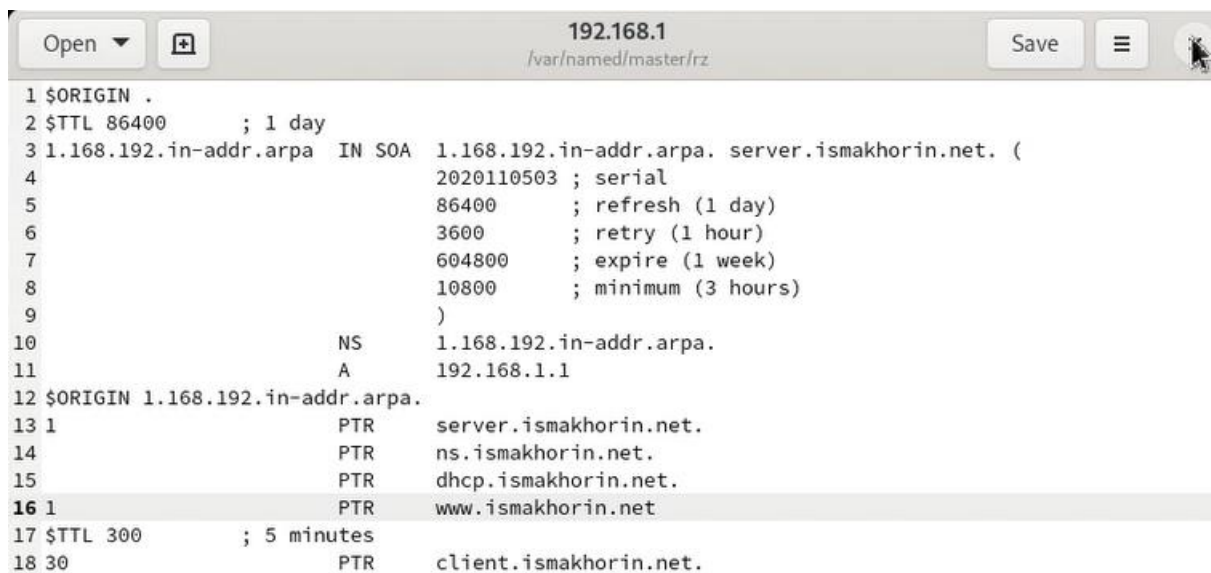


Рис. 4.2. Добавление записи для HTTP-сервера в конце файла прямой DNS-зоны /var/named/master/fz/ismakhorin.net.

Также в конце файла обратной зоны /var/named/master/rz/192.168.1 (рис. 4.3):

1	PTR	server.ismakhorin.net.
1	PTR	www.ismakhorin.net.

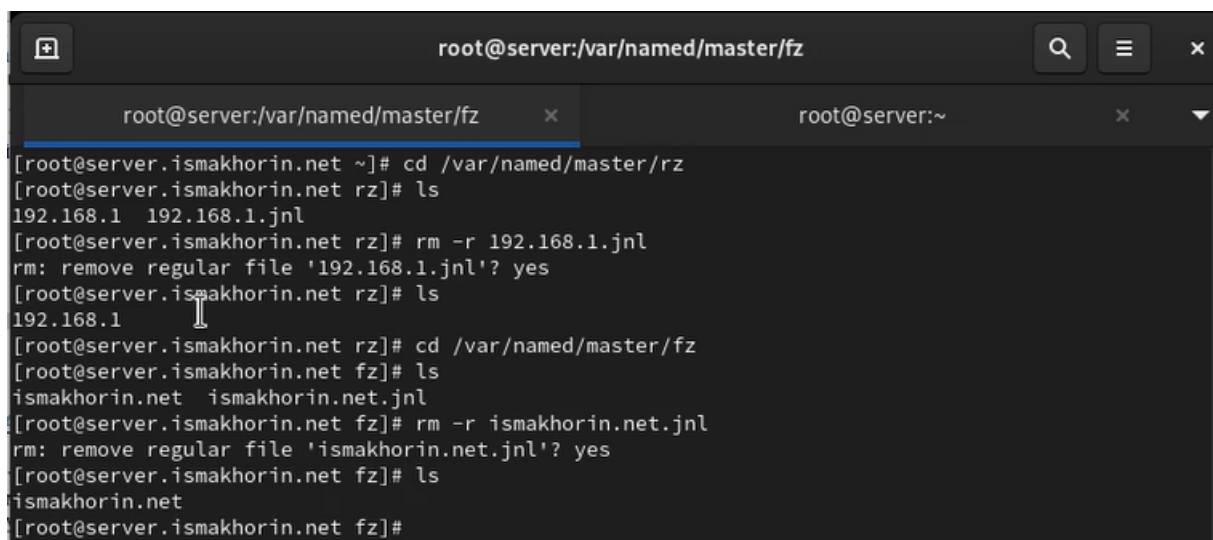


```
1 $ORIGIN .
2 $TTL 86400      ; 1 day
3 1.168.192.in-addr.arpa  IN SOA  1.168.192.in-addr.arpa. server.ismakhorin.net. (
4                                2020110503 ; serial
5                                86400      ; refresh (1 day)
6                                3600       ; retry (1 hour)
7                                604800     ; expire (1 week)
8                                10800      ; minimum (3 hours)
9                                )
10                             NS      1.168.192.in-addr.arpa.
11                             A       192.168.1.1
12 $ORIGIN 1.168.192.in-addr.arpa.
13 1                           PTR     server.ismakhorin.net.
14                             PTR     ns.ismakhorin.net.
15                             PTR     dhcp.ismakhorin.net.
16 1                           PTR     www.ismakhorin.net
17 $TTL 300        ; 5 minutes
18 30                 PTR     client.ismakhorin.net.
```

Рис. 4.3. Добавление записи для HTTP-сервера в конце файла обратной DNS-зоны /var/named/master/rz/192.168.1.

Нам требуется из соответствующих каталогов удалить файлы журналов DNS (рис. 4.4):

ismakhorin.net.jnl и 192.168.1.jnl. (рис. 4.4):



```
root@server:/var/named/master/fz

root@server:/var/named/master/fz x root@server:~ x
[root@server.ismakhorin.net ~]# cd /var/named/master/rz
[root@server.ismakhorin.net rz]# ls
192.168.1 192.168.1.jnl
[root@server.ismakhorin.net rz]# rm -r 192.168.1.jnl
rm: remove regular file '192.168.1.jnl'? yes
[root@server.ismakhorin.net rz]# ls
192.168.1
[root@server.ismakhorin.net rz]# cd /var/named/master/fz
[root@server.ismakhorin.net fz]# ls
ismakhorin.net ismakhorin.net.jnl
[root@server.ismakhorin.net fz]# rm -r ismakhorin.net.jnl
rm: remove regular file 'ismakhorin.net.jnl'? yes
[root@server.ismakhorin.net fz]# ls
ismakhorin.net
[root@server.ismakhorin.net fz]#
```

Рис. 4.4. Удаление файлов журналов DNS.

Перезапустим DNS-сервер:

systemctl start named

В каталоге /etc/httpd/conf.d создадим файлы server.ismakhorin.net.conf и www.ismakhorin.net.conf (рис. 4.5):

```
cd /etc/httpd/conf.d
```

```
touch server.user.net.conf
```

```
touch www.user.net.conf
```

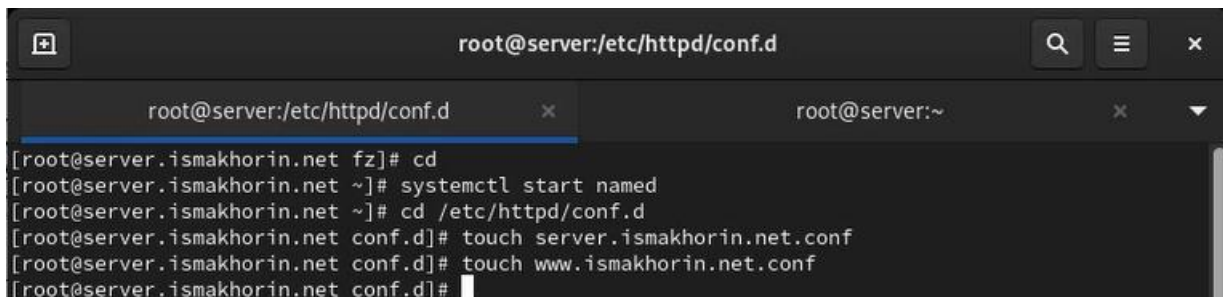


Рис. 4.5. Перезапуск DNS-сервера и создание в каталоге /etc/httpd/conf.d файлов server.ismakhorin.net.conf и www.ismakhorin.net.conf.

Откроем на редактирование файл server.ismakhorin.net.conf и внесём туда следующее содержание (рис. 4.6):



Рис. 4.6. Открытие на редактирование файла server.ismakhorin.net.conf и добавление содержания из лабораторной работы.

Откроем на редактирование файл www.ismakhorin.net.conf и внесём туда следующее содержание (рис. 4.7):



Рис. 4.7. Открытие на редактирование файла `www.ismakhorin.net.conf` и добавление содержания из лабораторной работы.

Перейдём в каталог `/var/www/html`, в котором находятся файлы с содержимым (контентом) веб-серверов, и создадим тестовые страницы для виртуальных веб-серверов `server.ismakhorin.net` и `www.ismakhorin.net`. Для виртуального веб-сервера `server.ismakhorin.net`:

```
cd /var/www/html
```

```
mkdir server.ismakhorin.net
```

```
cd /var/www/html/server.ismakhorin.net
```

```
touch index.html (рис. 4.8):
```

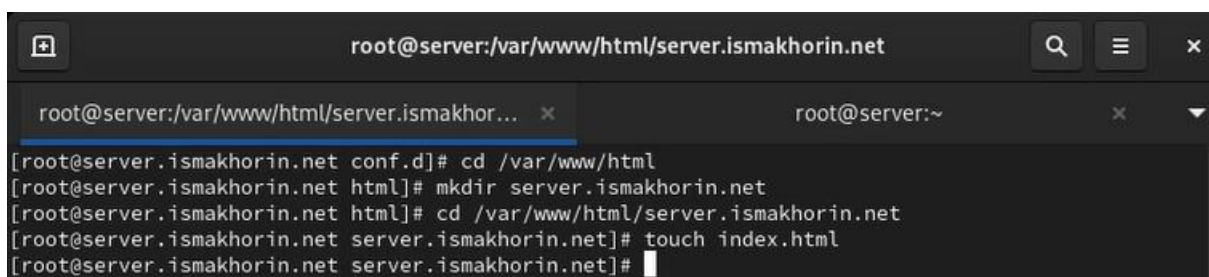


Рис. 4.8. Открытие каталога `/var/www/html` и создание тестовой страницы для виртуального веб-сервера `server.ismakhorin.net`.

Откроем на редактирование файл `index.html` и внесём следующее содержание (рис. 4.9):

```
Welcome to the server.ismakhorin.net server.
```



Рис. 4.9. Открытие на редактирование файла index.html и внесение содержания.

Для виртуального веб-сервера www.ismakhorin.net (рис. 4.10):

```
cd /var/www/html
```

```
mkdir www.ismakhorin.net
```

```
cd /var/www/html/www.ismakhorin.net
```

```
touch index.html
```

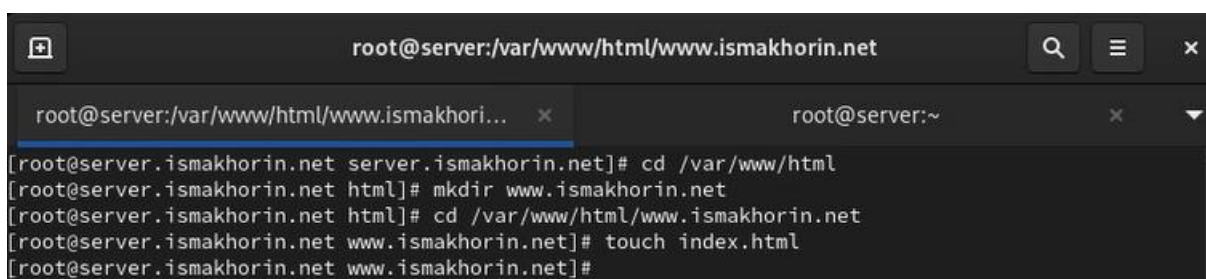


Рис. 4.10. Открытие каталога /var/www/html и создание тестовой страницы для виртуального веб-сервера www.ismakhorin.net.

Откроем на редактирование файл index.html и внесём следующее содержание (рис. 4.11):

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

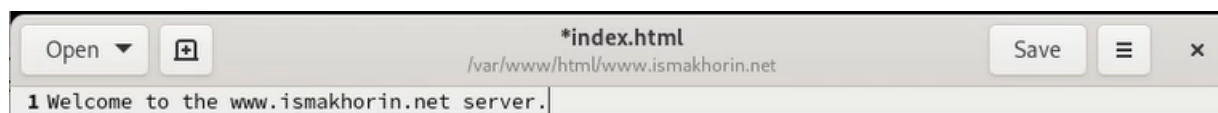


Рис. 4.11. Открытие на редактирование файла index.html и внесение содержания.

Скорректируем права доступа в каталог с веб-контентом:

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

Далее восстановим контекст безопасности в SELinux:

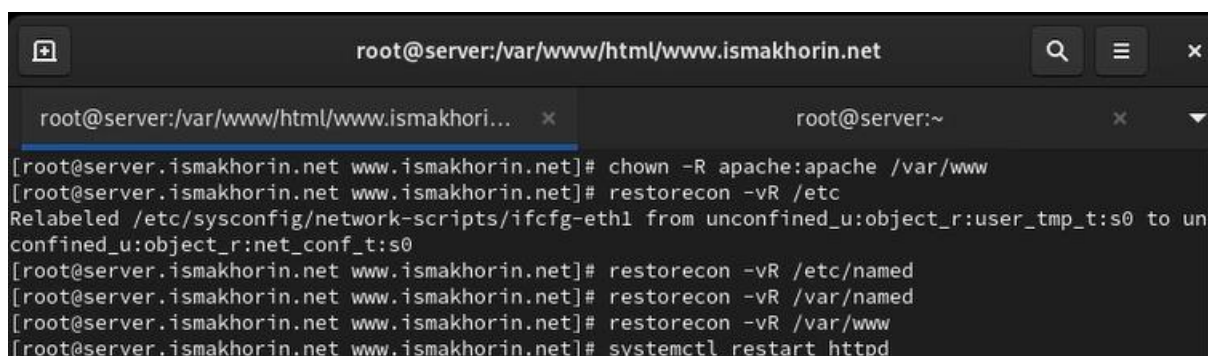
```
restorecon -vR /etc/named
```

```
restorecon -vR /var/named
```

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

Теперь перезапустим HTTP-сервер (рис. 4.12):

```
systemctl restart httpd
```

A screenshot of a terminal window with a dark background. The title bar shows 'root@server:/var/www/html/www.ismakhorin.net'. There are two tabs: 'root@server:/var/www/html/www.ismakhorin...' and 'root@server:~'. The terminal output shows the following commands and their results:

```
[root@server.ismakhorin.net www.ismakhorin.net]# chown -R apache:apache /var/www
[root@server.ismakhorin.net www.ismakhorin.net]# restorecon -vR /etc
Relabeled /etc/sysconfig/network-scripts/ifcfg-eth1 from unconfined_u:object_r:user_tmp_t:s0 to unconfined_u:object_r:net_conf_t:s0
[root@server.ismakhorin.net www.ismakhorin.net]# restorecon -vR /etc/named
[root@server.ismakhorin.net www.ismakhorin.net]# restorecon -vR /var/named
[root@server.ismakhorin.net www.ismakhorin.net]# restorecon -vR /var/www
[root@server.ismakhorin.net www.ismakhorin.net]# systemctl restart httpd
```

Рис. 4.12. Исправление прав доступа в каталог с веб-контентом, восстановление контекста безопасности в SELinux и перезапуск HTTP-сервера.

На виртуальной машине client убедимся в корректном доступе к веб-серверу по адресам server.ismakhorin.net (рис. 4.13) и www.ismakhorin.net (рис. 4.14) в адресной строке веб-браузера.

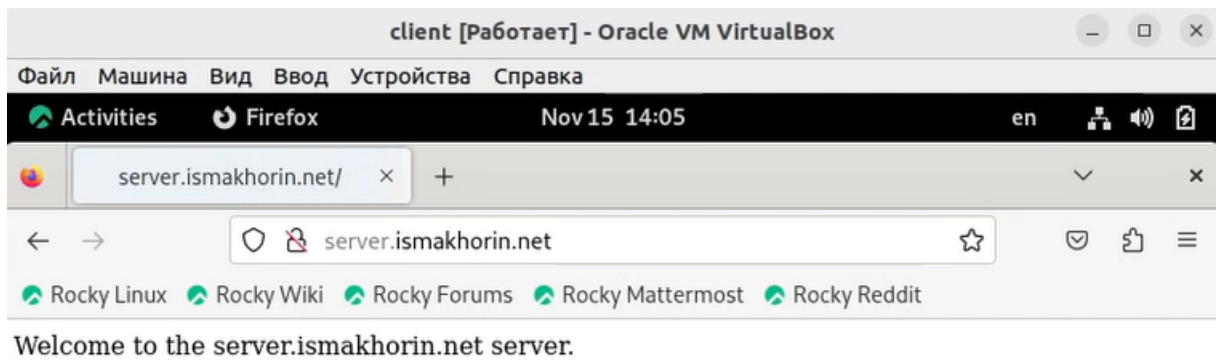


Рис. 4.13. Проверка корректного доступа на виртуальной машине client к веб-серверу по адресу server.ismakhorin.net.

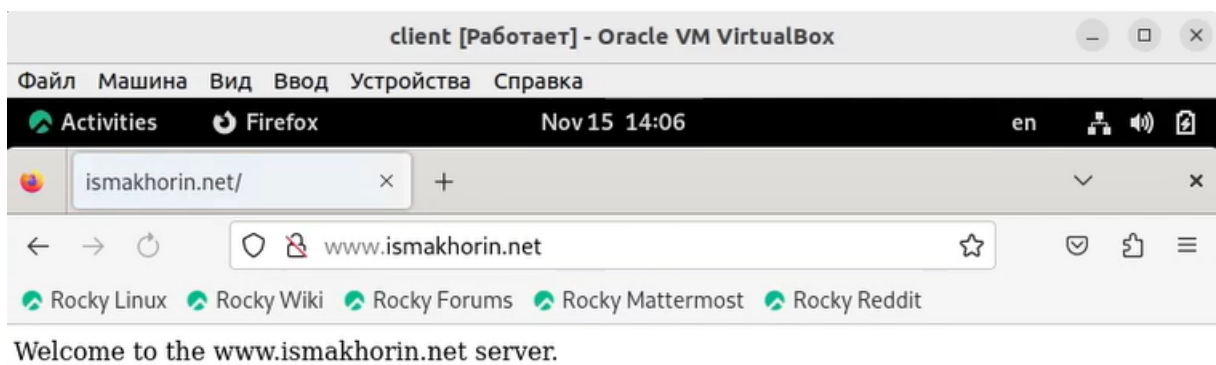
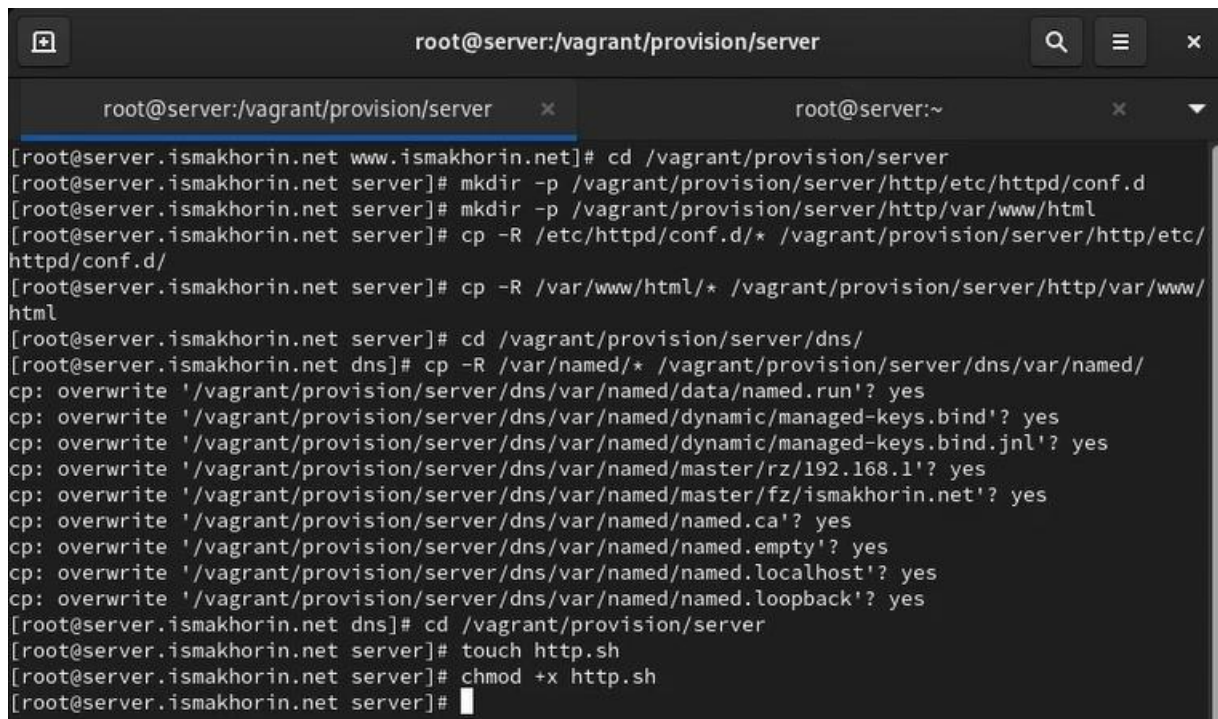


Рис. 4.14. Проверка корректного доступа на виртуальной машине client к веб-серверу по адресу www.ismakhorin.net.

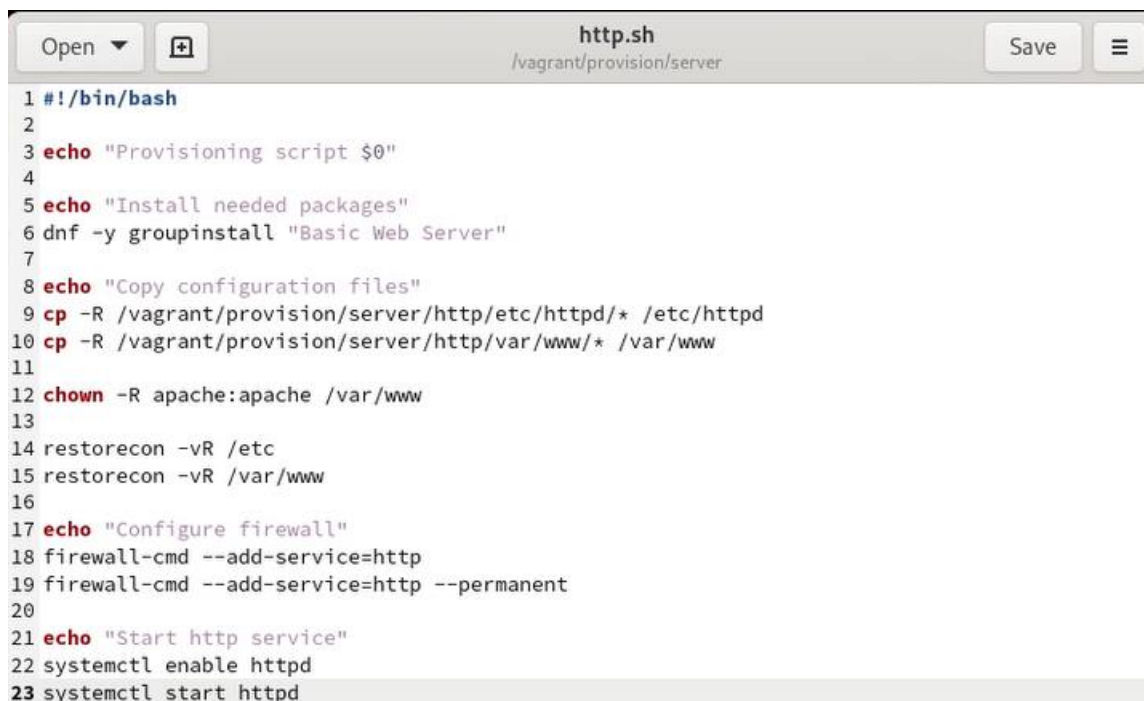
На виртуальной машине server перейдём в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/`, создадим в нём каталог `http`, в который поместим в соответствующие подкаталоги конфигурационные файлы HTTP-сервера. Теперь заменим конфигурационные файлы DNS-сервера. В каталоге `/vagrant/provision/server` создадим исполняемый файл `http.sh` (рис. 5.1):



```
root@server:~/vagrant/provision/server
root@server:~/vagrant/provision/server x root@server:~ x
[root@server.ismakhorin.net www.ismakhorin.net]# cd /vagrant/provision/server
[root@server.ismakhorin.net server]# mkdir -p /vagrant/provision/server/http/etc/httpd/conf.d
[root@server.ismakhorin.net server]# mkdir -p /vagrant/provision/server/http/var/www/html
[root@server.ismakhorin.net server]# cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/httpd/conf.d/
[root@server.ismakhorin.net server]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
[root@server.ismakhorin.net server]# cd /vagrant/provision/server/dns/
[root@server.ismakhorin.net dns]# cp -R /var/named/* /vagrant/provision/server/dns/var/named/
cp: overwrite '/vagrant/provision/server/dns/var/named/data/named.run'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind.jnl'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/192.168.1'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/ismakhorin.net'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/named.ca'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/named.empty'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/named.localhost'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/named.loopback'? yes
[root@server.ismakhorin.net dns]# cd /vagrant/provision/server
[root@server.ismakhorin.net server]# touch http.sh
[root@server.ismakhorin.net server]# chmod +x http.sh
[root@server.ismakhorin.net server]#
```

Рис. 5.1. Открытие на виртуальной машине server каталога для внесения изменений в настройки внутреннего окружения, создание в нём каталога http. Замена конфигурационных файлов DNS-сервера и создание исполняемого файла http.sh.

Откроем созданный файл на редактирование и пропишем в нём скрипт (рис. 5.2):



```
1 #!/bin/bash
2
3 echo "Provisioning script $0"
4
5 echo "Install needed packages"
6 dnf -y groupinstall "Basic Web Server"
7
8 echo "Copy configuration files"
9 cp -R /vagrant/provision/server/http/etc/httpd/* /etc/httpd
10 cp -R /vagrant/provision/server/http/var/www/* /var/www
11
12 chown -R apache:apache /var/www
13
14 restorecon -vR /etc
15 restorecon -vR /var/www
16
17 echo "Configure firewall"
18 firewall-cmd --add-service=http
19 firewall-cmd --add-service=http --permanent
20
21 echo "Start http service"
22 systemctl enable httpd
23 systemctl start httpd
```

Рис. 5.2. Открытие созданного файла на редактирование и прописывание скрипта.

Для отработки созданного скрипта во время загрузки виртуальных машин в конфигурационном файле Vagrantfile добавим в конфигурации сервера следующую запись (рис. 5.3):

```
41 path: "provision/server/01-dummy.sh"
42
43 server.vm.provision "server dns",
44 type: "shell",
45 preserve_order: true,
46 path: "provision/server/dns.sh"
47
48 server.vm.provision "server dhcp",
49 type: "shell",
50 preserve_order: true,
51 path: "provision/server/dhcp.sh"
52
53 server.vm.provision "server http",
54 type: "shell",
55 preserve_order: true,
56 path: "provision/server/http.sh"
57
58 server.vm.provider :virtualbox do |v|
59 v.linked_clone = true
60 # Customize the amount of memory on the VM
61 v.memory = 1024
```

Рис. 5.3. Добавление записи для отработки созданного скрипта во время загрузки виртуальных машин.

Вывод:

В ходе выполнения лабораторной работы были приобретены практические навыки по установке и базовому конфигурированию HTTP-сервера Apache.

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 Ubuntu логи Apache хранятся в /var/log/apache2/, а в CentOS - 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 hostить несколько сайтов на одном сервере. Это можно сделать двумя способами:

Базовый виртуальный хостинг: Разные сайты обслуживаются на одном сервере с использованием разных директорий DocumentRoot.

Name-based (или Host-based) виртуальный хостинг: 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.