

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра теории вероятностей и кибербезопасности

ОТЧЁТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №3

дисциплина: Администрирование сетевых подсистем

Студент: Бансимба Клодели Дьегра

Студ. билет № 1032215651

Группа: НПИбд-02-22

МОСКВА

2024 г.

Цель работы:

Целью данной работы является приобретение практических навыков по установке и конфигурированию DHCP-сервера.

Выполнение работы:

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

```
cd /var/tmp/claudey/vagrant
```

Далее запустим виртуальную машину server (Рис. 1.1):

```
make server-up
```

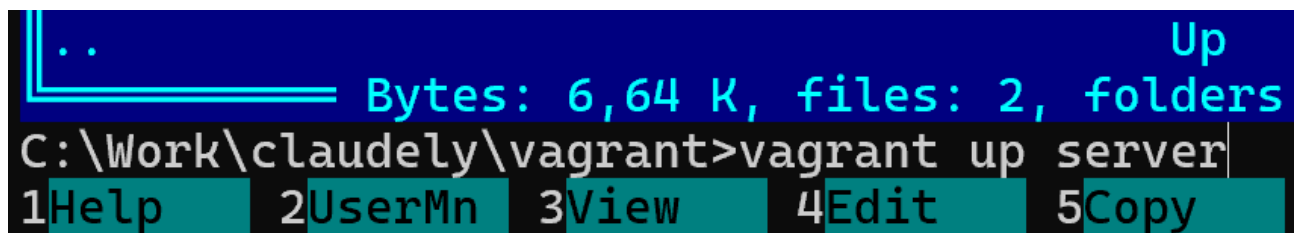


Рис. 1.1. Открытие рабочего каталога с проектом и запуск виртуальной машины server.

На виртуальной машине server войдём под нашим пользователем и откроем терминал. Перейдём в режим суперпользователя:

```
sudo -i
```

И установим dhcp (Рис. 1.2):

```
dnf -y install dhcp-server
```

```
root@server:~  
[claudely@server.claudely.net ~]$ sudo -i  
[sudo] password for claudely:  
[root@server.claudely.net ~]# dnf -y install dhcp-server  
Last metadata expiration check: 0:44:31 ago on Fri 27 Sep 2024 05:17:53 PM UTC.  
Dependencies resolved.  
=====
```

Package	Architecture	Version	Repository	Size
Installing:				
dhcp-server	x86_64	12:4.4.2-19.b1.el9	baseos	1.2 M
Installing dependencies:				
dhcp-common	noarch	12:4.4.2-19.b1.el9	baseos	128 k

```
Transaction Summary  
=====
```

Transaction Summary			
Install 2 Packages			
Total download size: 1.3 M			
Installed size: 4.2 M			
Downloading Packages:			
(1/2): dhcp-common-4.4.2-19.b1.el9.noarch.rpm	55 kB/s	128 kB	00:02
(2/2): dhcp-server-4.4.2-19.b1.el9.x86_64.rpm	345 kB/s	1.2 MB	00:03

Total	119 kB/s	1.3 MB	00:11

```
Running transaction check  
Transaction check succeeded.  
Running transaction test  
Transaction test succeeded.  
Running transaction  
  Preparing      :                                1/1  
  Installing     : dhcp-common-12:4.4.2-19.b1.el9.noarch 1/2  
  Running scriptlet: dhcp-server-12:4.4.2-19.b1.el9.x86_64 2/2  
  Installing     : dhcp-server-12:4.4.2-19.b1.el9.x86_64 2/2  
  Running scriptlet: dhcp-server-12:4.4.2-19.b1.el9.x86_64 2/2  
  Verifying      : dhcp-server-12:4.4.2-19.b1.el9.x86_64 1/2  
  Verifying      : dhcp-common-12:4.4.2-19.b1.el9.noarch 2/2  
  
Installed:  
  dhcp-common-12:4.4.2-19.b1.el9.noarch      dhcp-server-12:4.4.2-19.b1.el9.x86_64  
  
Complete!  
[root@server.claudely.net ~]#
```

Рис. 1.2. Переход в режим суперпользователя и установка dhcp.

Скопируем файл примера конфигурации DHCP `dhcpd.conf.example` из каталога `/usr/share/doc/dhcp*` в каталог `/etc/dhcp` и переименуем его в файл с названием `dhcpd.conf` (Рис. 2.1):

```
cd /etc/dhcp
```

```
cp /usr/share/doc/dhcp*/dhcpd.conf.example /etc/dhcp
```

```
mv /etc/dhcp/dhcpd.conf.example /etc/dhcp/dhcpd.conf
```

```
[root@server.claudely.net ~]#  
[root@server.claudely.net ~]# cd /etc/dhcp  
[root@server.claudely.net dhcp]# cp /usr/share/doc/dhcp*/dhcpd.conf.example /etc/dhcp  
[root@server.claudely.net dhcp]# mv /etc/dhcp/dhcpd.conf.example /etc/dhcp/dhcpd.conf  
mv: overwrite '/etc/dhcp/dhcpd.conf'? yes  
[root@server.claudely.net dhcp]#
```

Рис. 2.1. Копирование файла примера конфигурации DHCP и изменение его названия.

Откроем файл `/etc/dhcp/dhcpd.conf` на редактирование. В этом файле:

- Заменяем строку `option domain-name`
- Заменяем строку `option domain-name-servers`
- Раскомментируем строку `authoritative`
- На базе одного из приведённых в файле примеров конфигурирования подсети зададим собственную конфигурацию dhcp-сети, задав адрес подсети, диапазон адресов для распределения клиентам, адрес маршрутизатора и broadcast-адрес (Рис. 2.2).

Рис. 2.2. Открытие файла /etc/dhcp/dhcpd.conf на редактирование. Замена строки option domain-name и option domain-name-servers, снятие комментария со строки authoritative, создание собственной конфигурации dhcp-сети.

Откроем файл /etc/systemd/system/dhcpd.service на редактирование и заменим в нём строку (Рис. 2.4):

ExecStart=/usr/sbin/dhcpd -f -cf /etc/dhcp/dhcpd.conf -user dhcpd -group dhcpd --no-pid

на строку

ExecStart=/usr/sbin/dhcpd -f -cf /etc/dhcp/dhcpd.conf -user dhcpd -group dhcpd --no-pid eth1

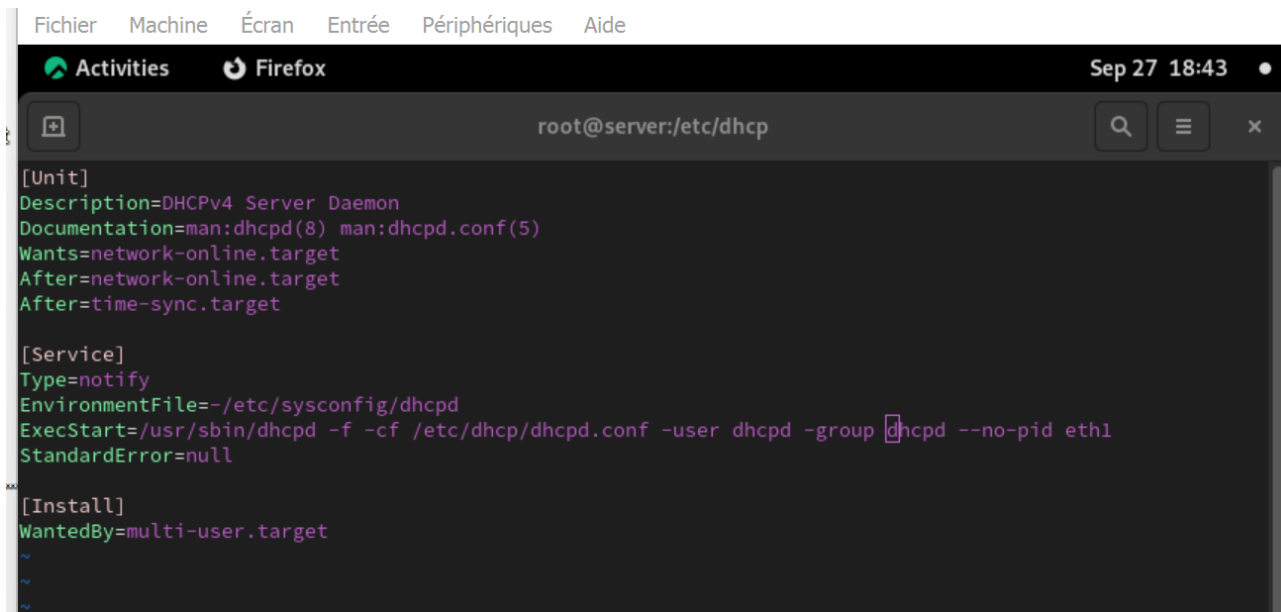


Рис. 2.4. Открытие файла `/etc/systemd/system/dhcpd.service` на редактирование и замена в нём строки.

Перезагрузим конфигурацию `dhcpd` и разрешим загрузку DHCP-сервера при запуске виртуальной машины `server` (Рис. 2.5):

```
systemctl --system daemon-reload
```

```
systemctl enable dhcpd
```

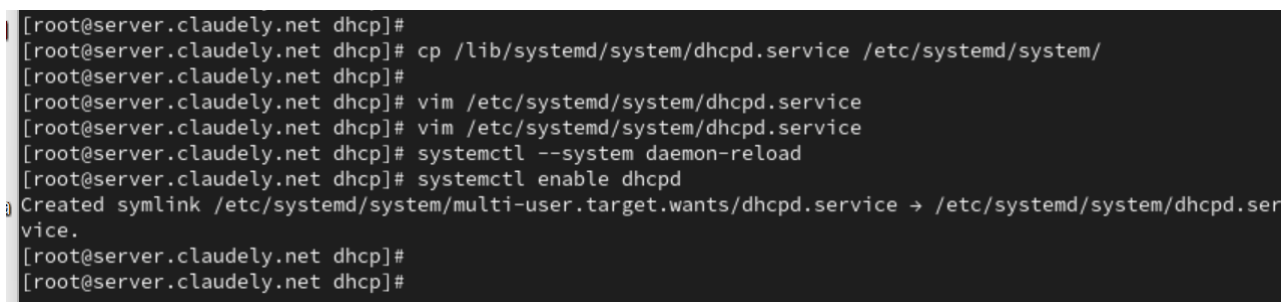
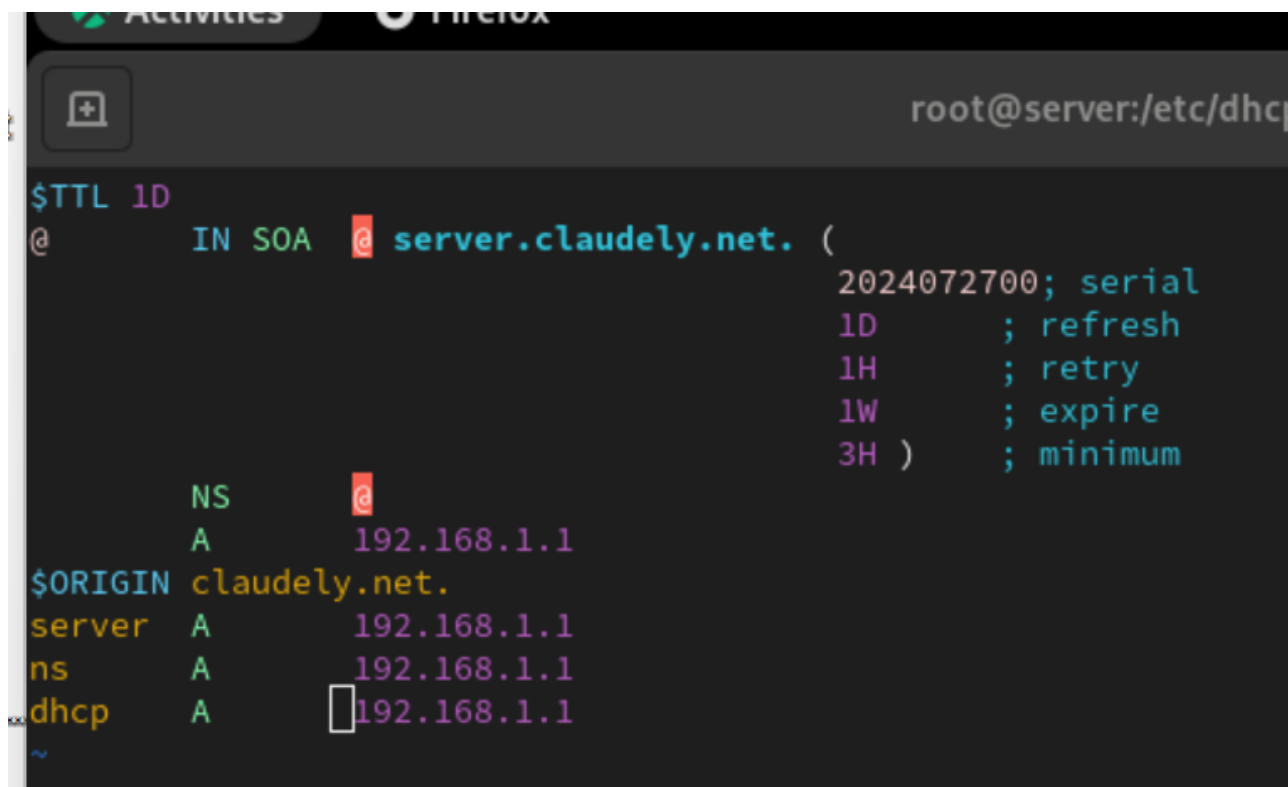


Рис. 2.5. Перезагрузка конфигурации `dhcpd` и разрешение загрузки DHCP-сервера при запуске виртуальной машины `server`.

Добавим запись для DHCP-сервера в конце файла прямой DNS-зоны `/var/named/master/fz/claudey.net` (Рис. 2.6):

dhcp A 192.168.1.1



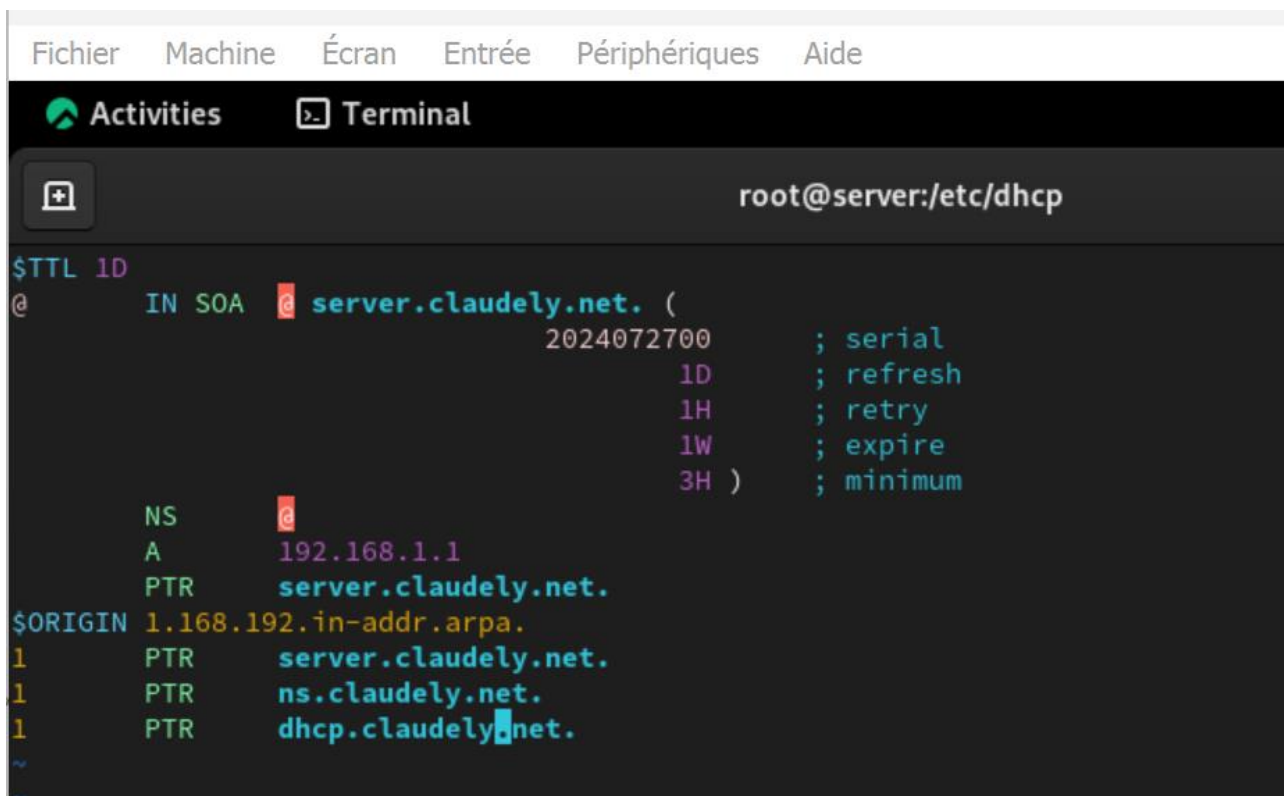
The screenshot shows a terminal window with a dark background. The title bar at the top indicates the user is root@server:/etc/dhcp. The terminal content shows a DNS zone file configuration for the domain claudely.net. It starts with a TTL of 1D, followed by an SOA record for server.claudely.net. with various refresh, retry, expire, and minimum values. Then, there is an NS record pointing to 192.168.1.1. The \$ORIGIN is set to claudely.net. Finally, three A records are listed: server pointing to 192.168.1.1, ns pointing to 192.168.1.1, and dhcp pointing to 192.168.1.1. A cursor is visible at the end of the dhcp line.

```
$TTL 1D
@ IN SOA server.claudely.net. (
                                2024072700; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H      ; minimum
    NS server.claudely.net.
    A 192.168.1.1
$ORIGIN claudely.net.
server A 192.168.1.1
ns     A 192.168.1.1
dhcp   A 192.168.1.1
~
```

Рис. 2.6. Добавление записи для DHCP-сервера в конце файла прямой DNS-зоны /var/named/master/fz/claudey.net.

И в конце файла обратной зоны /var/named/master/rz/192.168.1 (Рис. 2.7):

1 PTR dhcp.claudely.net.



```
Fichier  Machine  Écran  Entrée  Périphériques  Aide
Activities  Terminal

root@server:/etc/dhcp

$TTL 1D
@      IN SOA  server.claudely.net. (
                                2024072700      ; serial
                                1D              ; refresh
                                1H              ; retry
                                1W              ; expire
                                3H              ; minimum

    NS   server.claudely.net.
    A    192.168.1.1
    PTR  server.claudely.net.
$ORIGIN 1.168.192.in-addr.arpa.
1      PTR  server.claudely.net.
1      PTR  ns.claudely.net.
1      PTR  dhcp.claudely.net.
```

Рис. 2.7. Добавление записи для DHCP-сервера в конце файла обратной DNS-зоны /var/named/master/rz/192.168.1.

Перезапустим named:

```
systemctl restart named
```

И проверим, что можно обратиться к DHCP-серверу по имени (Рис. 2.8):

```
ping dhcp.claudely.net
```

```
[root@server.claudely.net dhcp]#
[root@server.claudely.net dhcp]# systemctl restart named
[root@server.claudely.net dhcp]# ping dhcp.claudely.net
PING dhcp.claudely.net (192.168.1.1) 56(84) bytes of data.
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=1 ttl=64 time=0.567 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=2 ttl=64 time=0.082 ms
64 bytes from server.claudely.net (192.168.1.1): icmp_seq=3 ttl=64 time=0.110 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=4 ttl=64 time=0.079 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=5 ttl=64 time=0.060 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=6 ttl=64 time=0.051 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=7 ttl=64 time=0.074 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=8 ttl=64 time=0.067 ms
64 bytes from dhcp.claudely.net (192.168.1.1): icmp_seq=9 ttl=64 time=0.107 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=10 ttl=64 time=0.055 ms
64 bytes from server.claudely.net (192.168.1.1): icmp_seq=11 ttl=64 time=0.082 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=12 ttl=64 time=0.071 ms
64 bytes from dhcp.claudely.net (192.168.1.1): icmp_seq=13 ttl=64 time=0.074 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=14 ttl=64 time=0.063 ms
64 bytes from server.claudely.net (192.168.1.1): icmp_seq=15 ttl=64 time=0.064 ms
64 bytes from ns.claudely.net (192.168.1.1): icmp_seq=16 ttl=64 time=0.067 ms
```


Рис. 2.8. Перезапуск named и выполнение проверки, что можно обратиться к DHCP-серверу по имени.

Внесём изменения в настройки межсетевого экрана узла server, разрешив работу с DHCP (рис. 2.9):

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

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

```
firewall-cmd --add-service=dhcp
```

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



```
[root@server.claudely.net dhcp]# firewall-cmd --list-services
cockpit dhcpv6-client dns ssh
[root@server.claudely.net dhcp]# firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp amqps apcupsd audit ausweis
app2 bacula bacula-client bareos-director bareos-filedaemon bareos-storage bb bgp bitcoin bitcoin-rpc bit
coin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent cockpi
t collectd condor-collector cratedb ctdb dds dds-multicast dds-unicast dhcp dhcpv6 dhcpv6-client distcc d
ns dns-over-tls docker-registry docker-swarm dropbox-lansync elasticsearch etcd-client etcd-server finger
foreman foreman-proxy freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp galera
ganglia-client ganglia-master git gpsd grafana gre high-availability http http3 https ident imap imaps ip
fs ipp ipp-client ipsec irc ircs iscsi-target isns jenkins kadmin kdeconnect kerberos kibana klogin kpass
kprop kshell kube-api kube-apiserver kube-control-plane kube-control-plane-secure kube-controller-mana
ger kube-controller-manager-secure kube-nodeport-services kube-scheduler kube-scheduler-secure kube-worke
r kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr llmnr-cl
ient llmnr-tcp llmnr-udp managesieve matrix mdns memcache minidlna mongodb mosh mountd mqtt mqtt-tls ms-w
bt mssql murmur mysql nbd nebula netbios-ns netdata-dashboard nfs nfs3 nmea-0183 nrpe ntp nut openvpn ovi
rt-imageio ovirt-storageconsole ovirt-vmconsole plex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresq
l privoxy prometheus prometheus-node-exporter proxy-dhcp ps2link ps3netsrv ptp pulseaudio puppetmaster qu
assel radius 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-lansync s
potify-sync squid ssdp ssh steam-streaming svdrp svn syncthing syncthing-gui syncthing-relay synergy sysl
og syslog-tls telnet tentacle tftp tile38 tinc tor-socks transmission-client upnp-client vdsms vnc-server
warpinator 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 zerotier
[root@server.claudely.net dhcp]# firewall-cmd --add-service=dhcp
success
[root@server.claudely.net dhcp]# firewall-cmd --add-service=dhcp --permanent
success
[root@server.claudely.net dhcp]#
```

Рис. 2.9. Внесение изменений в настройки межсетевого экрана узла server, разрешив работу с DHCP.

Восстановим контекст безопасности в SELinux (рис. 2.10):

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

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

```
restorecon -vR /var/lib/dhcpd/
```

```
[root@server.claudely.net dhcp]#  
[root@server.claudely.net dhcp]# restorecon -vR /etc  
Relabeled /etc/systemd/system/dhcpd.service from unconfined_u:object_r:systemd_unit_file_t:s0 to unconfined_u:object_r:dhcpd_unit_file_t:s0  
[root@server.claudely.net dhcp]# restorecon -vR /var/named  
[root@server.claudely.net dhcp]# restorecon -vR /var/lib/dhcpd/  
[root@server.claudely.net dhcp]# tail -f /var/log/messages  
Sep 27 19:36:42 server named[13359]: network unreachable resolving 'safebrowsing.googleapis.com/AAAA/IN': 2001:4860:4802:36::a#53  
Sep 27 19:36:42 server named[13359]: network unreachable resolving 'safebrowsing.googleapis.com/A/IN': 2001:4860:4802:38::a#53  
Sep 27 19:36:42 server systemd-journald[453]: Data hash table of /run/log/journal/bb344eeb9a684c4bafa12b93fea4dac6/system.journal has a fill level at 75.0 (2629 of 3505 items, 2019328 file size, 768 bytes per hash table item), suggesting rotation.  
Sep 27 19:36:42 server systemd-journald[453]: /run/log/journal/bb344eeb9a684c4bafa12b93fea4dac6/system.journal: Journal header limits reached or header out-of-date, rotating.  
Sep 27 19:36:42 server named[13359]: network unreachable resolving 'safebrowsing.googleapis.com/AAAA/IN': 2001:4860:4802:38::a#53  
Sep 27 19:36:42 server named[13359]: network unreachable resolving 'safebrowsing.googleapis.com/A/IN': 2001:4860:4802:34::a#53  
Sep 27 19:36:42 server named[13359]: network unreachable resolving 'safebrowsing.googleapis.com/AAAA/IN': 2001:4860:4802:34::a#53  
Sep 27 19:36:42 server named[13359]: network unreachable resolving 'safebrowsing.googleapis.com/A/IN': 2001:4860:4802:32::a#53
```

Рис. 2.10. Восстановление контекста безопасности в SELinux.

В дополнительном терминале запустим мониторинг происходящих в системе процессов в реальном времени (рис. 2.11):

```
tail -f /var/log/messages
```

В основном рабочем терминале запустим DHCP-сервер (рис. 2.12):

```
systemctl start dhcpd
```

```
[root@server.claudely.net dhcp]#  
[root@server.claudely.net dhcp]# systemctl start dhcpd  
[root@server.claudely.net dhcp]#  
[root@server.claudely.net dhcp]#
```

Рис. 2.12. Запуск в основном рабочем терминале DHCP-сервера.

Зафиксируем внесённые изменения для внутренних настроек виртуальной машины client и запустим её, введя в терминале (рис. 3.1):

```
make client-provision
```

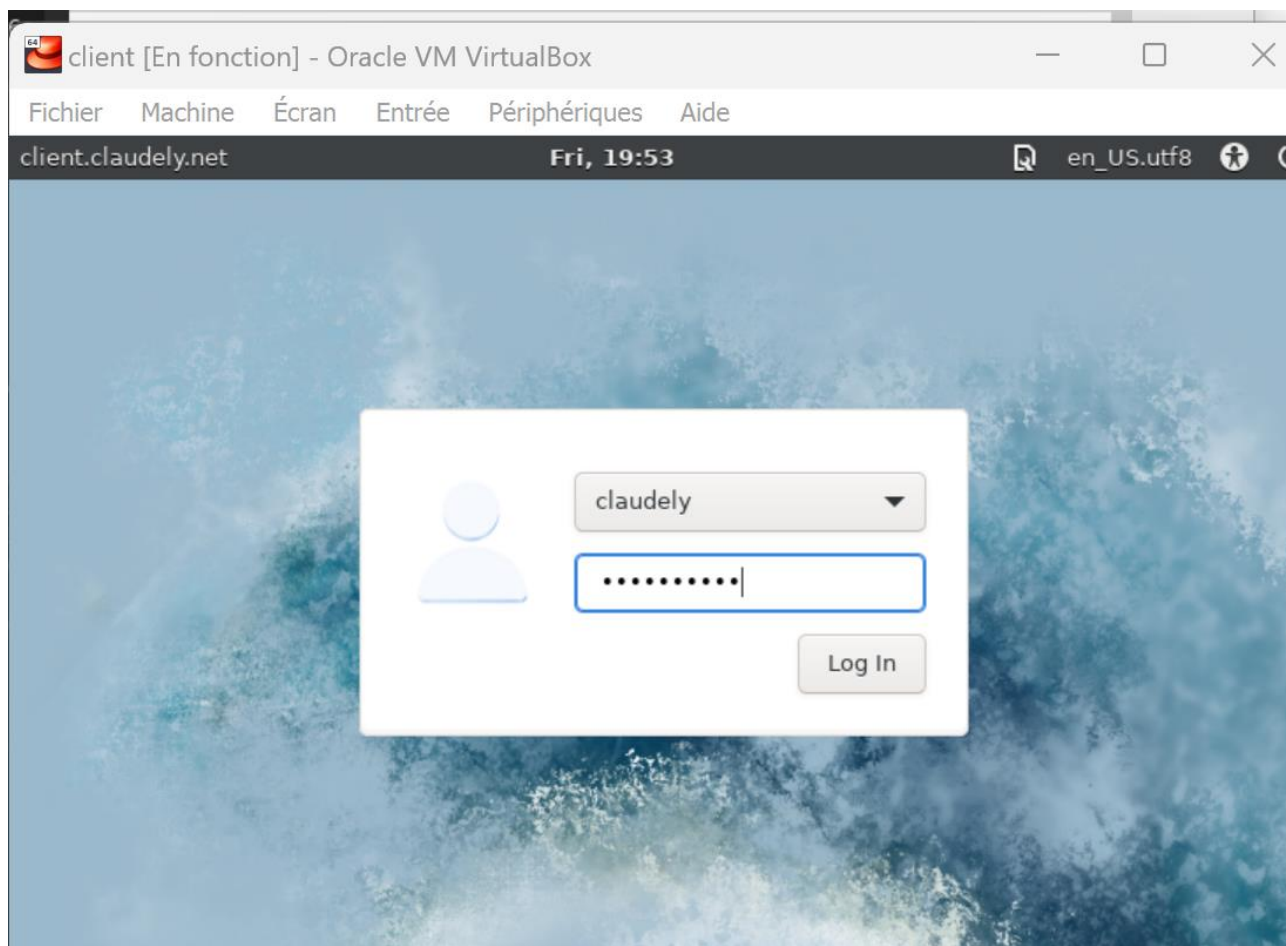
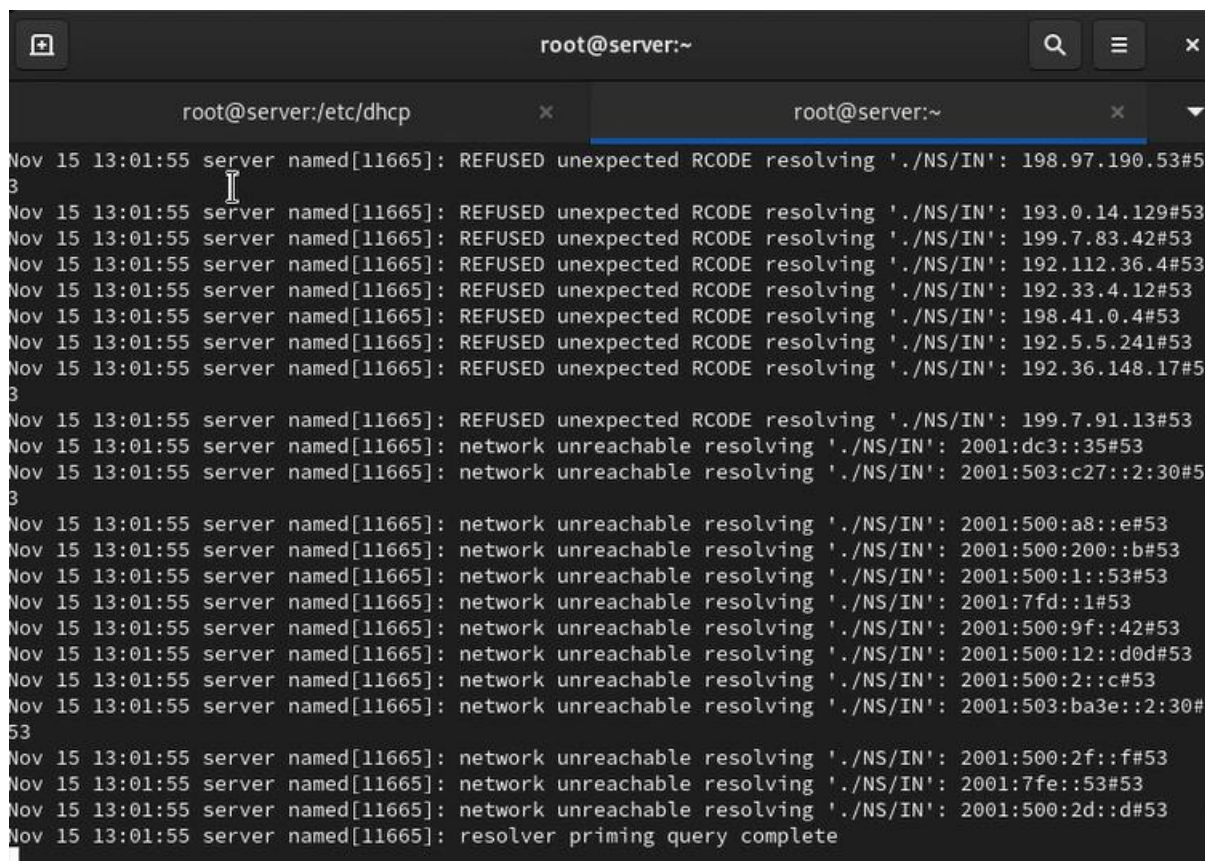


Рис. 3.1. Фиксация внесённых изменений для внутренних настроек виртуальной машины client и её запуск.

После загрузки виртуальной машины client мы можем увидеть на виртуальной машине server на терминале с мониторингом происходящих в системе процессов записи о подключении к виртуальной внутренней сети узла client и выдачи ему IP-адреса из соответствующего диапазона адресов. Также информацию о работе DHCP-сервера можно наблюдать в файле `/var/lib/dhcpd/dhcpd.leases` (рис. 3.2):



```
Nov 15 13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 198.97.190.53#53
Nov 15 13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 193.0.14.129#53
Nov 15 13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 199.7.83.42#53
Nov 15 13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 192.112.36.4#53
Nov 15 13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 192.33.4.12#53
Nov 15 13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 198.41.0.4#53
Nov 15 13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 192.5.5.241#53
Nov 15 13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 192.36.148.17#53
Nov 15 13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 199.7.91.13#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:dc3::35#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:503:c27::2:30#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:a8::e#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:200::b#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:1::53#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:7fd::1#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:9f::42#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:12::d0d#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:2::c#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:503:ba3e::2:30#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:2f::f#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:7fe::53#53
Nov 15 13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:2d::d#53
Nov 15 13:01:55 server named[11665]: resolver priming query complete
```

Рис. 3.2. Просмотр записей о подключении к виртуальной внутренней сети узла client и выдачи ему IP-адреса из соответствующего диапазона адресов.

Войдём в систему виртуальной машины client под нашим пользователем и откроем терминал. В терминале введём ifconfig. На экран теперь выведена информация об имеющихся интерфейсах (рис. 3.3):


```
client [En fonction] - Oracle VM VirtualBox
Fichier  Machine  Écran  Entrée  Périphériques  Aide
Activities  Terminal  Sep 27 19:57
claudey@client:~
[claudey@client.claudey.net ~]$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fea9:337c  prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:a9:33:7c  txqueuelen 1000  (Ethernet)
    RX packets 1446  bytes 161844 (158.0 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 1246  bytes 190563 (186.0 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.1.30  netmask 255.255.255.0  broadcast 192.168.1.255
    inet6 fe80::a00:27ff:feea:d6fa  prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:ea:d6:fa  txqueuelen 1000  (Ethernet)
    RX packets 85  bytes 12948 (12.6 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 378  bytes 39350 (38.4 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 17  bytes 2045 (1.9 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 17  bytes 2045 (1.9 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

[claudey@client.claudey.net ~]$
```

Рис. 3.3. Вывод на экран информации об имеющихся интерфейсах.

Внесём изменения в конфигурационный файл `/etc/dhcp/dhcpd.conf`, добавив в него разрешение на динамическое обновление DNS-записей с локального узла прямой и обратной зон (рис. 4.3):

Рис. 4.3. Внесение изменений в конфигурационный файл `/etc/dhcp/dhcpd.conf`, добавив в него разрешение на динамическое обновление DNS-записей с локального узла прямой и обратной зон.

Перезапустим DHCP-сервер (рис. 4.4):

`systemctl restart dhcpd`

Рис. 4.4. Перезапуск DHCP-сервера.

На виртуальной машине `client` под нашим пользователем откроем терминал и с помощью утилиты `dig` убедимся в наличии DNS-записи о клиенте в прямой DNS-зоне (рис. 5):

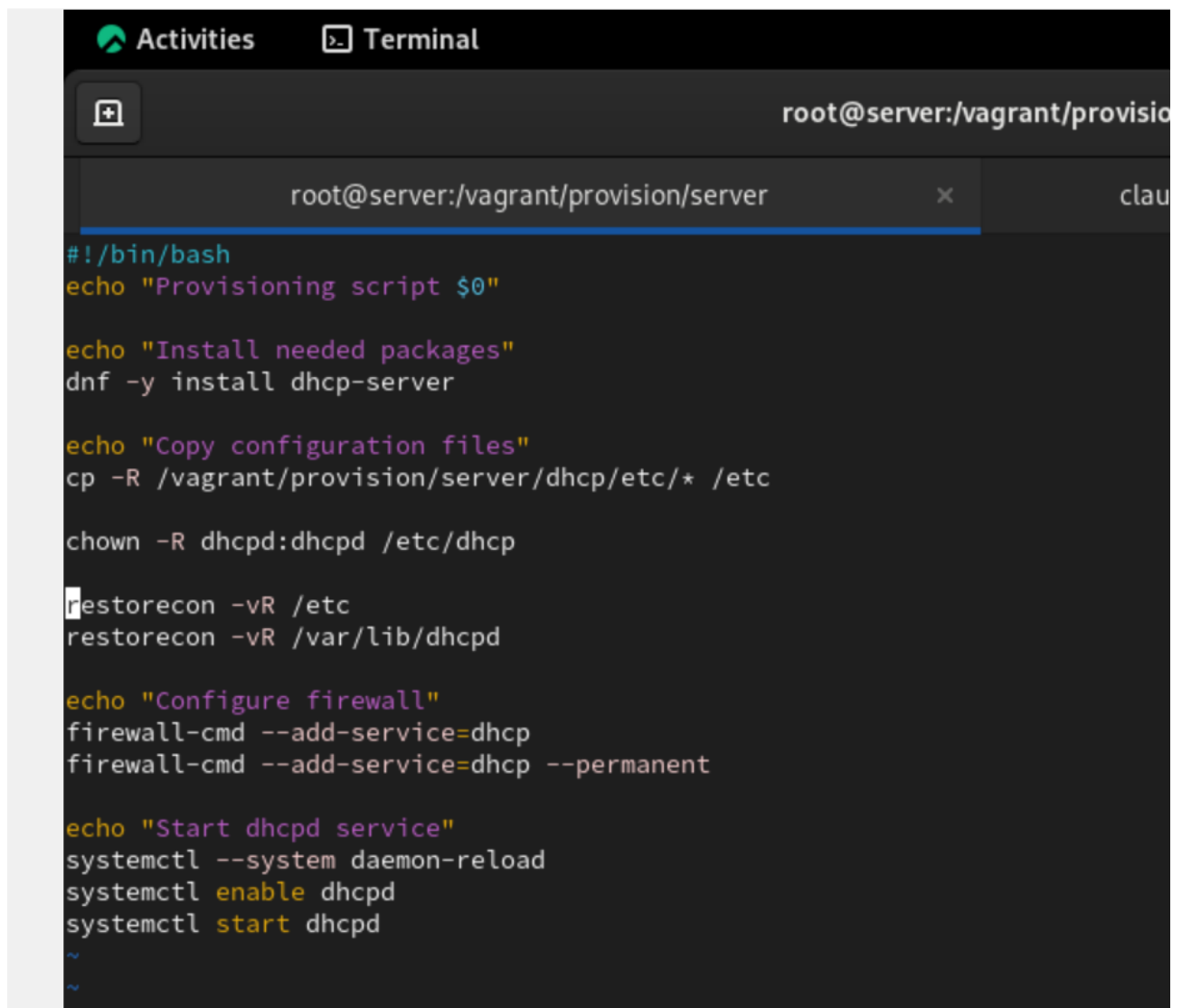
dig @192.168.1.1 client.claudely.net

```
[claudey@client.claudely.net ~]$  
[claudey@client.claudely.net ~]$ dig @192.168.1.1 client.claudely.net  
  
; <<>> DiG 9.16.23-RH <<>> @192.168.1.1 client.claudely.net  
; (1 server found)  
;; global options: +cmd  
;; Got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 40938  
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1  
  
;; OPT PSEUDOSECTION:  
; EDNS: version: 0, flags:; udp: 1232  
; COOKIE: dbce045803fe19800100000066f710d42aledale23b1386b (good)  
;; QUESTION SECTION:  
;client.claudely.net.          IN      A  
  
;; ANSWER SECTION:  
client.claudely.net.  300     IN      A      192.168.1.30  
  
;; Query time: 2 msec  
;; SERVER: 192.168.1.1#53(192.168.1.1)  
;; WHEN: Fri Sep 27 20:08:51 UTC 2024  
;; MSG SIZE rcvd: 92  
  
[claudey@client.claudely.net ~]$
```

Рис. 5. Проверка наличия DNS-записи о клиенте в прямой DNS-зоне.

окружения /vagrant/provision/server/, создание в нём каталога dhcp. Замена конфигурационных файлов DNS-сервера. Создание в каталоге /vagrant/provision/server исполняемого файла dhcp.sh.

Откроем этот файл на редактирование и пропишем в нём скрипт из лабораторной работы (рис. 6.2):



The screenshot shows a terminal window titled 'Terminal' with a tab for 'root@server:/vagrant/provision/server'. The terminal displays a series of commands for setting up a DHCP server. The commands are: `#!/bin/bash`, `echo "Provisioning script $0"`, `echo "Install needed packages"`, `dnf -y install dhcp-server`, `echo "Copy configuration files"`, `cp -R /vagrant/provision/server/dhcp/etc/* /etc`, `chown -R dhcpd:dhcpd /etc/dhcp`, `restorecon -vR /etc`, `restorecon -vR /var/lib/dhcpd`, `echo "Configure firewall"`, `firewall-cmd --add-service=dhcp`, `firewall-cmd --add-service=dhcp --permanent`, `echo "Start dhcpd service"`, `systemctl --system daemon-reload`, `systemctl enable dhcpd`, and `systemctl start dhcpd`. The terminal ends with two tilde characters (~).

```
#!/bin/bash
echo "Provisioning script $0"

echo "Install needed packages"
dnf -y install dhcp-server

echo "Copy configuration files"
cp -R /vagrant/provision/server/dhcp/etc/* /etc

chown -R dhcpd:dhcpd /etc/dhcp

restorecon -vR /etc
restorecon -vR /var/lib/dhcpd

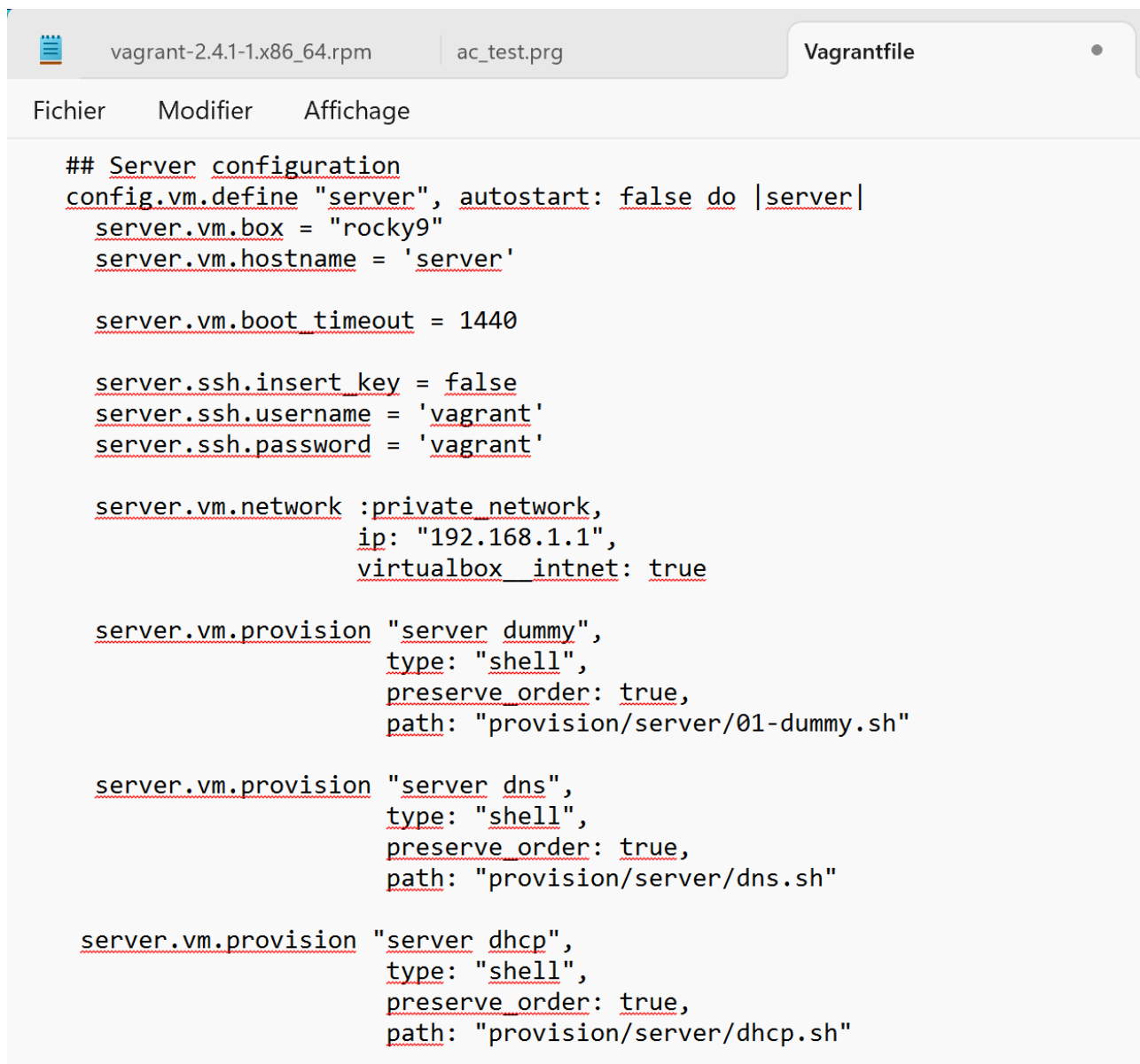
echo "Configure firewall"
firewall-cmd --add-service=dhcp
firewall-cmd --add-service=dhcp --permanent

echo "Start dhcpd service"
systemctl --system daemon-reload
systemctl enable dhcpd
systemctl start dhcpd
~
~
```

Рис. 6.2. Открытие файла на редактирование и помещение в него скрипта.

Для отработки созданного скрипта во время загрузки виртуальной машины `server` в конфигурационном файле `Vagrantfile` добавим в разделе конфигурации для сервера (рис. 6.3):

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

The screenshot shows a text editor window titled 'Vagrantfile' with tabs for 'vagrant-2.4.1-1.x86_64.rpm' and 'ac_test.prg'. The menu bar includes 'Fichier', 'Modifier', and 'Affichage'. The code is a Vagrantfile configuration for a virtual machine named 'server'. It defines the VM's box as 'rocky9', sets the hostname to 'server', and configures the boot timeout to 1440 seconds. SSH settings are set to 'vagrant' for both username and password, with 'insert_key' set to false. The network is configured as a private network with IP '192.168.1.1' and 'virtualbox____intnet' set to true. Three provision scripts are listed: 'server_dummy', 'server_dns', and 'server_dhcp', all using 'shell' type and 'preserve_order: true'.

```
## Server configuration
config.vm.define "server", autostart: false do |server|
  server.vm.box = "rocky9"
  server.vm.hostname = 'server'

  server.vm.boot_timeout = 1440

  server.ssh.insert_key = false
  server.ssh.username = 'vagrant'
  server.ssh.password = 'vagrant'

  server.vm.network :private_network,
    ip: "192.168.1.1",
    virtualbox__intnet: true

  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"
```

Рис. 6.3. Настройка отработки созданного скрипта во время загрузки виртуальной машины server.

Вывод:

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

Ответы на контрольные вопросы:

1. В каких файлах хранятся настройки сетевых подключений? - В наиболее популярных операционных системах, таких как Windows и Linux, настройки сетевых подключений хранятся в различных файлах:

В Windows, основные настройки сетевых подключений, такие как IP-адрес, маска подсети, шлюз, DNS-серверы и другие, хранятся в реестре. Однако, конфигурационные данные также могут быть сохранены в текстовых файлах, таких как `ipconfig` или в файле конфигурации подключения.

В Linux, настройки сети обычно хранятся в текстовых файлах в директории `/etc/network/` или `/etc/sysconfig/network-scripts/`.

2. За что отвечает протокол DHCP? - Протокол DHCP (Dynamic Host Configuration Protocol) отвечает за автоматическое присвоение сетевых настроек устройствам в сети, таких как IP-адресов, маски подсети, шлюза, DNS-серверов и других параметров.

3. Поясните принцип работы протокола DHCP. Какими сообщениями обмениваются клиент и сервер, используя протокол DHCP? - Принцип работы протокола DHCP:

Discover (Обнаружение): Клиент отправляет в сеть запрос на обнаружение DHCP-сервера.

Offer (Предложение): DHCP-сервер отвечает клиенту, предлагая ему конфигурацию сети.

Request (Запрос): Клиент принимает предложение и отправляет запрос на использование предложенной конфигурации.

Acknowledgment (Подтверждение): DHCP-сервер подтверждает клиенту, что предложенная конфигурация принята и может быть использована.

4. В каких файлах обычно находятся настройки DHCP-сервера? За что отвечает каждый из файлов?
- Настройки DHCP-сервера обычно хранятся в файлах конфигурации, таких как:

В Linux: `/etc/dhcp/dhcpd.conf`

В Windows: `%SystemRoot%\System32\dhcp\dhcpd.conf`

Они содержат информацию о диапазонах IP-адресов, параметрах сети и других опциях DHCP.

5. Что такое DDNS? Для чего применяется DDNS? - **DDNS (Dynamic Domain Name System)** - это система динамического доменного имени. Она используется для автоматического обновления записей DNS, когда IP-адрес узла изменяется. DDNS применяется, например, в домашних сетях, где IP-адреса часто изменяются посредством DHCP.
6. Какую информацию можно получить, используя утилиту `ifconfig`? Приведите примеры с использованием различных опций. - **Утилита `ifconfig` используется для получения информации о сетевых интерфейсах.**

Примеры:

`ifconfig`: Показывает информацию обо всех активных сетевых интерфейсах.

`ifconfig eth0`: Показывает информацию о конкретном интерфейсе (в данном случае, `eth0`).

7. Какую информацию можно получить, используя утилиту `ping`? Приведите примеры с использованием различных опций. - **Утилита `ping` используется для проверки доступности узла в сети.**

Примеры:

`ping google.com`: Пингует домен `google.com`.

`ping -c 4 192.168.1.1`: Пингует IP-адрес `192.168.1.1` и отправляет 4 эхо-запроса.