

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: Пакавира Арсениу Висенте Луиш

Студ. билет № 1032225105

Группа: НФИбд-02-23

МОСКВА

2025 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/user/vagrant
```

Далее запустим виртуальную машину server (Рис. 1.1):

```
make server-up
```

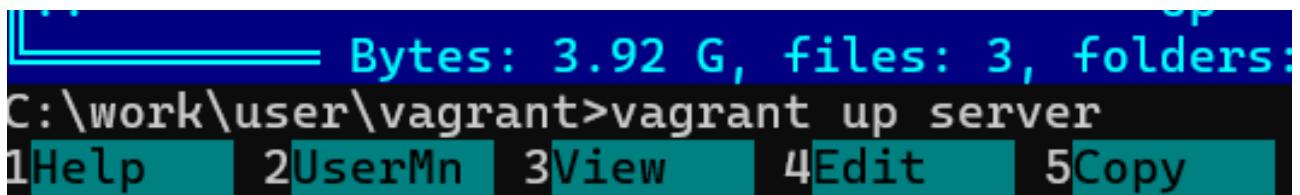


Рис. 1.1. Открытие рабочего каталога с проектом и запуск виртуальной машины server.

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
```

```
[user@server.user.net ~]$ sudo -i
[sudo] password for user:
[root@server.user.net ~]# dnf -y install bind bind-utils
Last metadata expiration check: 0:23:11 ago on Tue 06 Jan 2026 01:18:33 PM UTC.
Package bind-utils-32:9.16.23-18.el9_4.1.x86_64 is already installed.
Dependencies resolved.
=====
 Package           Architecture   Version        Repository      Size
=====
Installing:
 bind              x86_64        32:9.16.23-34.el9_7.1    appstream     488 k
Upgrading:
 bind-libs          x86_64        32:9.16.23-34.el9_7.1    appstream     1.2 M
bind-license        noarch       32:9.16.23-34.el9_7.1    appstream     13 k
bind-utils          x86_64        32:9.16.23-34.el9_7.1    appstream     199 k
openssl            x86_64        1:3.5.1-4.el9_7         baseos        1.4 M
openssl-devel       x86_64        1:3.5.1-4.el9_7         appstream     3.4 M
openssl-libs        x86_64        1:3.5.1-4.el9_7         baseos        2.3 M
Installing dependencies:
 bind-dnssec-doc    noarch       32:9.16.23-34.el9_7.1    appstream     45 k
openssl-fips-provider x86_64      1:3.5.1-4.el9_7         baseos        812 k
python3-bind        noarch       32:9.16.23-34.el9_7.1    appstream     61 k
python3-ply          noarch       3.11-14.el9.0.1        baseos        103 k
Installing weak dependencies:
 bind-dnssec-utils  x86_64        32:9.16.23-34.el9_7.1    appstream     113 k
=====
Transaction Summary
=====
Install 6 Packages
Upgrade 6 Packages
```

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:

- Заменим строку option domain-name
- Заменим 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).

```

# dhcpcd.conf
#
# Sample configuration file for ISC dhcpcd
#
# option definitions common to all supported networks...
option domain-name "claudely.net";
option domain-name-servers ns.claudely.net;

default-lease-time 600;
max-lease-time 7200;

# Use this to enable / disable dynamic dns updates globally.
#ddns-update-style none;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
authoritative;

# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
log-facility local7;

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.30 192.168.1.199;
    option routers 192.168.1.1;
    option broadcast-address 192.168.1.255;
}

```

Рис. 2.2. Открытие файла /etc/dhcp/dhcpcd.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/dhcpcd.service на редактирование и заменим в нём строку (Рис. 2.4):

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

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



```
[Unit]
Description=DHCPv4 Server Daemon
Documentation=man:dhcpcd(8) man:dhcpcd.conf(5)
Wants=network-online.target
After=network-online.target
After=time-sync.target

[Service]
Type=notify
EnvironmentFile=-/etc/sysconfig/dhcpcd
ExecStart=/usr/sbin/dhcpcd -f -cf /etc/dhcp/dhcpcd.conf -user dhcpcd -group dhcpcd --no-pid eth1
StandardError=null

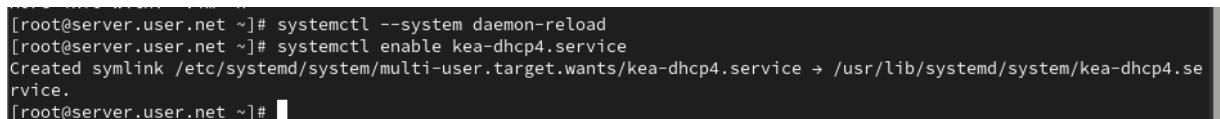
[Install]
WantedBy=multi-user.target
```

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

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

systemctl --system daemon-reload

systemctl enable dhcpcd



```
[root@server.user.net ~]# systemctl --system daemon-reload
[root@server.user.net ~]# systemctl enable kea-dhcp4.service
Created symlink /etc/systemd/system/multi-user.target.wants/kea-dhcp4.service → /usr/lib/systemd/system/kea-dhcp4.service.
[root@server.user.net ~]#
```

Рис. 2.5. Перезагрузка конфигурации dhcpcd и разрешение загрузки DHCPсервера при запуске виртуальной машины server.

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

```
dhcp      A      192.168.1.1
```



```
$TTL 1D
@      IN SOA  @ server.claudely.net. (
                        2024072700; serial
                        1D      ; refresh
                        1H      ; retry
                        1W      ; expire
                        3H )    ; minimum
NS      @
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 DNSзоны /var/named/master/fz/user.net.

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

```
1      PTR      dhcp.user.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      @
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 server.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):

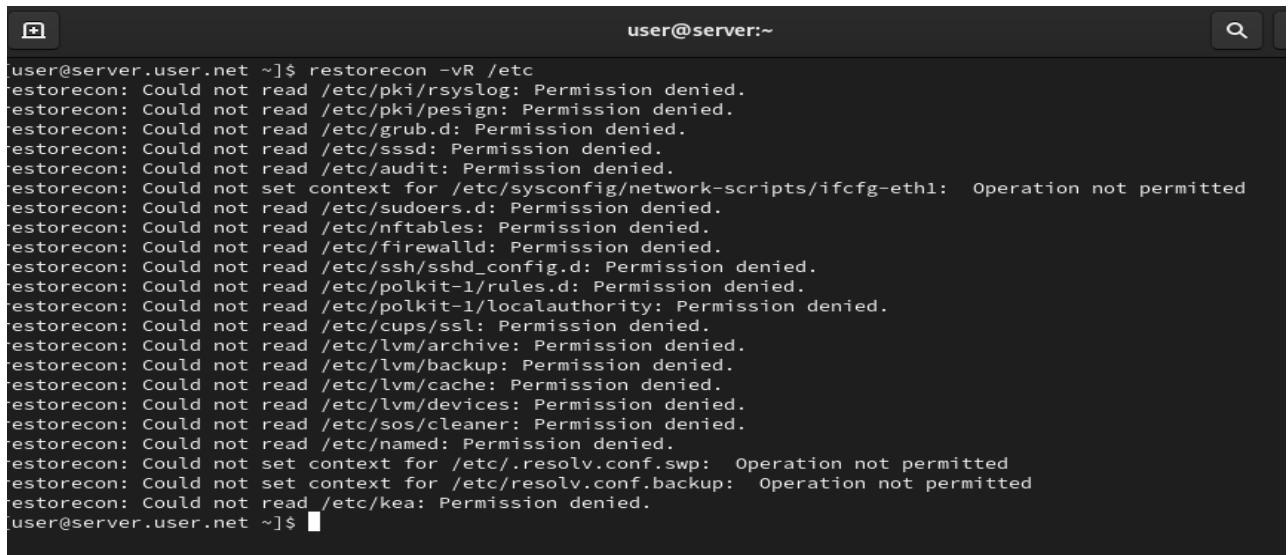
```
firewall-cmd --list-services firewall-cmd  
--get-services firewall-cmd --add-  
service=dhcp  
firewall-cmd --add-service=dhcp --permanent
```

```
[user@server.user.net ~]$ firewall-cmd --list-services  
cockpit dhcpv6-client dns http https imap imaps ntp pop3 pop3s samba smtp ssh ssh-custom  
[user@server.user.net ~]$ firewall-cmd --get-services  
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp amqps apcupsd audit ausweisapp2 ba-  
bacula-client bareos-director bareos-filedaemon bareos-storage bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitco-  
estnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent cockpit collectd condor-collector d-  
db ctdb dds dds-multicast dds-unicast dhcp dhcpcv6 dhcpcv6-client distcc dns dns-over-tls docker-registry docker-s-  
dropbox-lansync elasticsearch etcd-client etcd-server finger foreman foreman-proxy freeipa-4 freeipa-ldap free-  
ipa-dnsservice freeipa-replication freeipa-trust ftp galera ganglia-client ganglia-master git gpgsql grafana gre high-availa-  
bility http http3 https ident imap imaps ipfs ipp ipp-client ipsec irc ircs iscsi-target isns jenkins kadmin kdecon-  
kerberos kibana klogin kpasswd kprop kshell kube-api kube-apiserver kube-control-plane kube-control-plane-secure-  
e-controller-manager kube-controller-manager-secure kube-nodeport-services kube-scheduler kube-scheduler-secure-  
-worker kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr llmnr-cl-  
llmnr-tcp llmnr-udp managesieve matrix mdns memcache minidlna mongod mosh mountd mqtt mqtt-tls ms-wbt mssql mun-  
mysql nbd nebula netbios-ns netdata-dashboard nfs nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-stora-  
sole ovirt-vmconsole plex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy prometheus-no-  
xporter proxy-dhcp ps2link ps3netsrv ptp pulseaudio puppetmaster quassel radius rdp redis redis-sentinel rpc-bin-  
uotad rsh rsyncd rtsp salt-master samba samba-client samba-dc sane sip sips slp smtp smtp-submission smtps snmp  
tls snmptrap snmptrap spideroak-lansync spotify-sync squid ssdp ssh ssh-custom steam-streaming svdrp svn syn-  
ng syncthing-gui syncthing-relay synergy syslog syslog-tls telnet tentacle tftp tile38 tor-socks transmissi-  
lient upnp-client vdsm vnc-server warpinator wbem-http wbem-https wireguard ws-discovery ws-discovery-client ws-  
over-tcp ws-discovery-udp wsman wsmans xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-s-  
r zerotier  
[user@server.user.net ~]$ firewall-cmd --add-service=dhcp  
success  
[user@server.user.net ~]$ firewall-cmd --add-service=dhcp --permanent  
success  
[user@server.user.net ~]$ █
```

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/
```



```
user@server.user.net ~]$ restorecon -vR /etc
restorecon: Could not read /etc/pki/rsyslog: Permission denied.
restorecon: Could not read /etc/pki/pesign: Permission denied.
restorecon: Could not read /etc/grub.d: Permission denied.
restorecon: Could not read /etc/sssd: Permission denied.
restorecon: Could not read /etc/audit: Permission denied.
restorecon: Could not set context for /etc/sysconfig/network-scripts/ifcfg-eth1: Operation not permitted
restorecon: Could not read /etc/sudoers.d: Permission denied.
restorecon: Could not read /etc/nftables: Permission denied.
restorecon: Could not read /etc/firewalld: Permission denied.
restorecon: Could not read /etc/ssh/sshd_config.d: Permission denied.
restorecon: Could not read /etc/polkit-1/rules.d: Permission denied.
restorecon: Could not read /etc/polkit-1/localauthority: Permission denied.
restorecon: Could not read /etc/cups/ssl: Permission denied.
restorecon: Could not read /etc/lvm/archive: Permission denied.
restorecon: Could not read /etc/lvm/backup: Permission denied.
restorecon: Could not read /etc/lvm/cache: Permission denied.
restorecon: Could not read /etc/lvm/devices: Permission denied.
restorecon: Could not read /etc/sos/cleaner: Permission denied.
restorecon: Could not read /etc/named: Permission denied.
restorecon: Could not set context for /etc/resolv.conf.swp: Operation not permitted
restorecon: Could not set context for /etc/resolv.conf.backup: Operation not permitted
restorecon: Could not read /etc/kea: Permission denied.
user@server.user.net ~]$
```

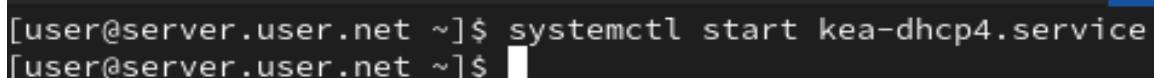
Рис. 2.10. Восстановление контекста безопасности в SELinux.

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 dhcpcd
```



```
[user@server.user.net ~]$ systemctl start kea-dhcp4.service
[user@server.user.net ~]$
```

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
```

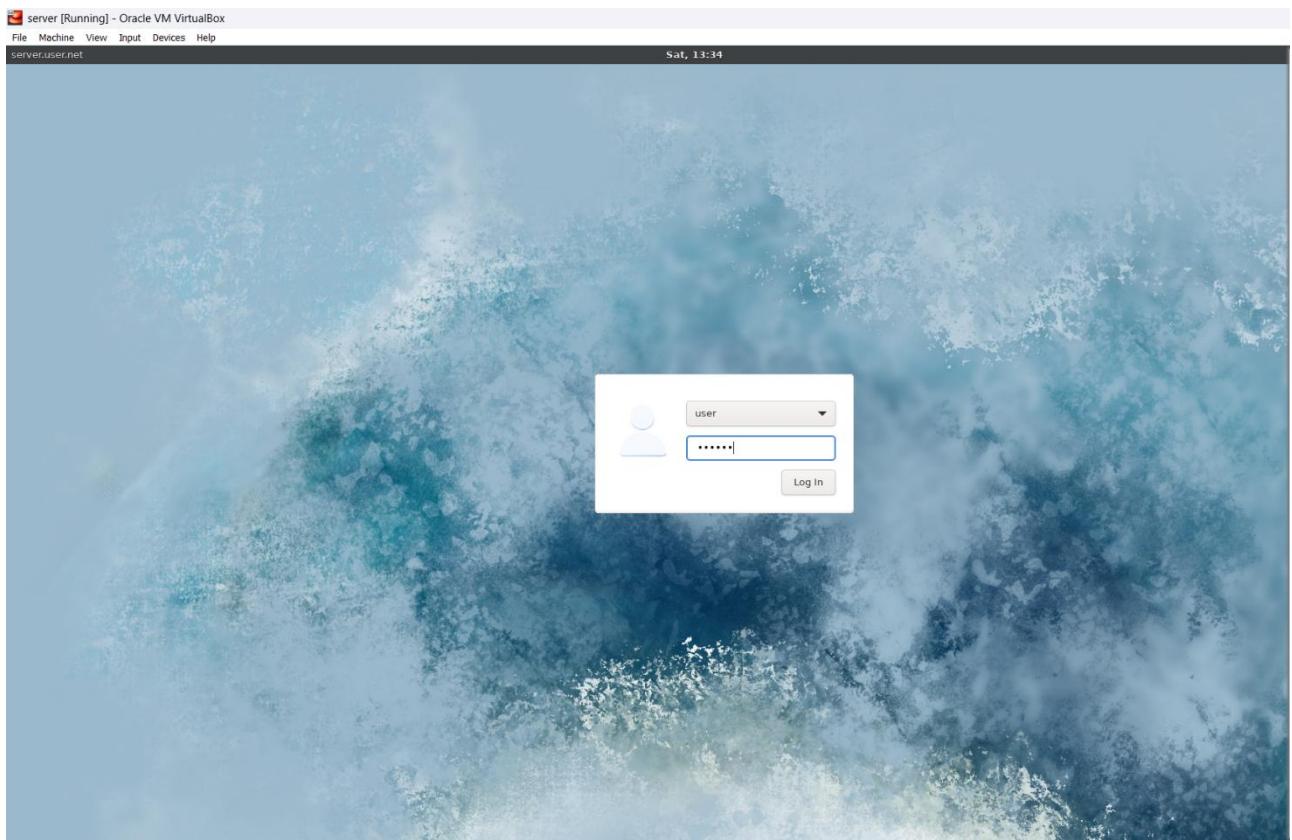


Рис. 3.1. Фиксация внесённых изменений для внутренних настроек виртуальной машины client и её запуск.

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 выдаче 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):

The screenshot shows a terminal window with two tabs. The left tab is titled 'root@server:/etc/dhcp' and the right tab is titled 'root@server:~'. The right tab is active and displays log entries from the 'named' daemon. The log entries show various 'REFUSED' responses for DNS queries from clients with IP addresses such as 198.97.190.53, 193.0.14.129, 199.7.83.42, 192.112.36.4, 192.33.4.12, 198.41.0.4, 192.5.5.241, and 192.36.148.17. It also shows 'network unreachable' responses for various IPv6 addresses (e.g., 2001:dc3::35, 2001:503:c27::2:30, 2001:500:a8::e, 2001:500:200:b, 2001:500:1::53, 2001:7fd::1, 2001:500:9f::42, 2001:500:12::d0d, 2001:500:2::c, 2001:503:ba3e::2:30) and a 'resolver priming query complete' message.

```
13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 198.97.190.53#53
13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 193.0.14.129#53
13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 199.7.83.42#53
13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 192.112.36.4#53
13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 192.33.4.12#53
13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 198.41.0.4#53
13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 192.5.5.241#53
13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 192.36.148.17#53

13:01:55 server named[11665]: REFUSED unexpected RCODE resolving './NS/IN': 199.7.91.13#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:dc3::35#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:503:c27::2:30#53

13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:a8::e#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:200:b#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:1::53#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:7fd::1#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:9f::42#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:12::d0d#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:2::c#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:503:ba3e::2:30#53

13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:2f::f#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:7fe::53#53
13:01:55 server named[11665]: network unreachable resolving './NS/IN': 2001:500:2d::d#53
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):

```
[root@elabdev ~]# 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
```

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):

```

# dhcpcd.conf
#
# Sample configuration file for ISC dhcpcd
#
# option definitions common to all supported networks...
option domain-name "claudely.net";
option domain-name-servers ns.claudely.net;

default-lease-time 600;
max-lease-time 7200;

# Use this to enable / disable dynamic dns updates globally.
#ddns-update-style none;

ddns-updates on;
ddns-update-style interim;
ddns-domainname "claudely.net.";
ddns-rev-domainname "in-addr.arpa.";

zone claudely.net. {
    primary 127.0.0.1;
}

zone 1.168.192.in-addr.arpa. {
    primary 127.0.0.1;
}

```

Рис. 4.3. Внесение изменений в конфигурационный файл /etc/dhcp/dhcpcd.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 dhcpcd

```

[root@server.claudely.net dhcpcd]#
[root@server.claudely.net dhcpcd]# systemctl restart dhcpcd
[root@server.claudely.net dhcpcd]#
[root@server.claudely.net dhcpcd]# ls /var/named/master/fz
claudely.net  claudely.net.jnl
[root@server.claudely.net dhcpcd]# █

```

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.user.net
```

```
[claudely@client.claudely.net ~]$  
[claudely@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  
  
[claudely@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" in the Activities menu. The window title bar indicates the session is root at "server:/vagrant/provision". The terminal content displays a bash script for provisioning a DHCP server. The script includes commands for installing the package, copying configuration files, changing ownership, and configuring the firewall. It also starts the dhcpcd service.

```
#!/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 dhcpcd:dhcpcd /etc/dhcp

restorecon -vR /etc
restorecon -vR /var/lib/dhcpcd

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

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

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 code editor window with a tab bar at the top. The tabs are labeled "vagrant-2.4.1-1.x86_64.rpm", "ac_test.prg", and "Vagrantfile". Below the tabs, there are three menu items: "Fichier", "Modifier", and "Affichage". The main area of the editor contains a Vagrantfile script. The script defines a server configuration, sets up SSH access, configures a private network, and provisions the server with three shell scripts: "01-dummy.sh", "dns.sh", and "dhcp.sh".

```
## 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.