

Лабораторная работа №3

Администрирование сетевых подсистем

Машков Илья Евгеньевич

Содержание

1 Цель работы	6
2 Задание	7
3 Выполнение лабораторной работы	8
3.1 Установка DHCP-сервера	8
3.2 Конфигурирование DHCP-сервера	9
3.3 Анализ работы DHCP-сервера	13
3.4 Настройка обновления DNS-зоны	15
3.5 Анализ работы DHCP-сервера после настройки обновления DNS-зоны	21
3.6 Внесение изменений в настройки внутреннего окружения виртуальной машины	22
4 Выводы	25
Список литературы	26

Список иллюстраций

3.1 Установка dhcp	8
3.2 Сохранение конфигурационного файла	9
3.3 Замена шаблона domain-name	9
3.4 Создание собственной dhcp-сети	10
3.5 Привязка dhcpcd к eth1	10
3.6 Проверка	10
3.7 Перезагрузка и включение автозапуска DHCP-сервера	11
3.8 Изменения в файле прямой зоны	11
3.9 Изменения в файле обратной зоны	11
3.10 Перезапуск named	11
3.11 Обращение к dhcp-серверу	12
3.12 Внесение изменений в настройки межсетевого экрана	12
3.13 Восстановление контекста безопасности	12
3.14 Проверка корректной работы dhcp-сервера	13
3.15 01-routing.sh	13
3.16 Внесение изменений в Vagrantfile	14
3.17 Информация в kea-leases4.csv	14
3.18 Информация о сетевых интерфейсах client	15
3.19 Создание ключа	15
3.20 dhcp_updater.key	16
3.21 Изменения прав доступа	16
3.22 Подключение ключа	17
3.23 Изменения в user.net	17
3.24 Проверка и перезапуск DNS-сервера	18
3.25 tsig-keys.json	18
3.26 Смена владельца и прав доступа	18
3.27 Изменения в kea-dhcp-ddns.conf	19
3.28 Смена владельца и проверка файла, запуск службы ddns	19
3.29 Статус службы	20
3.30 Изменения в kea-dhcp4.conf	20
3.31 Проверка файла после изменений	20
3.32 Статус dhcp-сервера	21
3.33 Переполучение адреса на клиенте	21
3.34 Обращение client.user.net	22
3.35 Внесение изменений в настройки внутреннего окружения	22
3.36 Замена конфигов dns-сервера	23
3.37 dhcp.sh	23

3.38 Изменения в файле Vagrantfile 24

Список таблиц

1 Цель работы

Приобретение практических навыков по установке и конфигурированию DHCP-сервера.

2 Задание

1. Установите на виртуальной машине server DHCP-сервер.
2. Настройте виртуальную машину server в качестве DHCP-сервера для виртуальной внутренней сети.
3. Проверьте корректность работы DHCP-сервера в виртуальной внутренней сети путём запуска виртуальной машины client и применения соответствующих утилит диагностики.
4. Настройте обновление DNS-зоны при появлении в виртуальной внутренней сети новых узлов.
5. Проверьте корректность работы DHCP-сервера и обновления DNS-зоны в виртуальной внутренней сети путём запуска виртуальной машины client и применения соответствующих утилит диагностики.
6. Напишите скрипт для Vagrant, фиксирующий действия по установке и настройке DHCP-сервера во внутреннем окружении виртуальной машины server. Соответствующим образом внести изменения в Vagrantfile.

3 Выполнение лабораторной работы

3.1 Установка DHCP-сервера

Произвожу установку dhcp (рис. [3.1]).

```
[root@server ~]# dnf -y install kea
Last metadata expiration check: 0:02:16 ago on Sun 16 Nov 2025 02:03:03 AM MSK.
Dependencies resolved.
=====
Package           Architecture Version      Repository  Size
=====
Installing:
  kea               x86_64       2.6.4-1.el9    epel        1.3 M
Installing dependencies:
  kea-libs          x86_64       2.6.4-1.el9    epel        3.0 M
  log4cplus         x86_64       2.0.5-15.el9   epel        337 k
  mariadb-connector-c x86_64     3.2.6-1.el9_0  appstream   195 k
  mariadb-connector-c-config x86_64     3.2.6-1.el9_0  appstream   9.8 k
  postgresql-private-libs x86_64     13.22-1.el9_0 appstream   137 k
Transaction Summary
=====
Install 6 Packages

Total download size: 5.0 M
Installed size: 18 M
Downloading Packages:
(1/6): log4cplus-2.0.5-15.el9.x86_64.rpm                                164 kB/s | 337 kB   00:02
(2/6): mariadb-connector-c-3.2.6-1.el9_0.x86_64.rpm                         1.0 MB/s | 195 kB   00:00
(3/6): postgresql-private-libs-13.22-1.el9_6.x86_64.rpm                   759 kB/s | 137 kB   00:00
(4/6): mariadb-connector-c-config-3.2.6-1.el9_0.noarch.rpm                50 kB/s | 9.8 kB   00:00
(5/6): kea-2.6.4-1.el9.x86_64.rpm                                         265 kB/s | 1.3 MB   00:04
(6/6): kea-libs-2.6.4-1.el9.x86_64.rpm                                     376 kB/s | 3.0 MB   00:08
Total                                         434 kB/s | 5.0 MB   00:11

Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing :                                                       1/1
  Installing : mariadb-connector-c-config-3.2.6-1.el9_0.noarch             1/6
  Installing : mariadb-connector-c-3.2.6-1.el9_0.x86_64                    2/6
  Installing : postgresql-private-libs-13.22-1.el9_6.x86_64              3/6
  Installing : log4cplus-2.0.5-15.el9.x86_64                            4/6
  Installing : kea-libs-2.6.4-1.el9.x86_64                               5/6
  Running scriptlet: kea-2.6.4-1.el9.x86_64                           6/6
  Installing : kea-2.6.4-1.el9.x86_64                               6/6
  Running scriptlet: kea-2.6.4-1.el9.x86_64                           6/6
  Verifying  : kea-2.6.4-1.el9.x86_64                               1/6
  Verifying  : kea-libs-2.6.4-1.el9.x86_64                          2/6
  Verifying  : log4cplus-2.0.5-15.el9.x86_64                         3/6
  Verifying  : mariadb-connector-c-3.2.6-1.el9_0.x86_64                  4/6
  Verifying  : postgresql-private-libs-13.22-1.el9_6.x86_64            5/6
  Verifying  : mariadb-connector-c-config-3.2.6-1.el9_0.noarch          6/6

Installed:
  kea-2.6.4-1.el9.x86_64                                              kea-libs-2.6.4-1.el9.x86_64
  log4cplus-2.0.5-15.el9.x86_64                                         mariadb-connector-c-3.2.6-1.el9_0.x86_64
  mariadb-connector-c-config-3.2.6-1.el9_0.noarch                         postgresql-private-libs-13.22-1.el9_6.x86_64
```

Рис. 3.1: Установка dhcp

3.2 Конфигурирование DHCP-сервера

На всякий случай сохраняю изначальную версию конфиговского файла (рис. [3.2]).

```
[root@server ~]# cp /etc/kea/kea-dhcp4.conf /etc/kea/kea-dhcp4.conf_$(date -I)
[root@server ~]# nano /etc/kea/kea-dhcp4.conf
```

Рис. 3.2: Сохранение конфигурационного файла

Затем открываю “**kea-dhcp4.conf**” и меняю в нём шаблон “**“domain-name”**” (рис. [3.3]).

```
// Options
{
    "name": "domain-name-servers",
    "data": "192.168.1.1"
},
// Typically people prefer to refer to options by their names, so they
// don't need to remember the code names. However, some people like
// to use numerical values. For example, option "domain-name" uses
// option code 15, so you can reference to it either by
// "name": "domain-name" or "code": 15.
{
    "code": 15,
    "data": "user.net"
},
// Domain search is also a popular option. It tells the client to
// attempt to resolve names within those specified domains. For
// example, name "foo" would be attempted to be resolved as
// foo.mydomain.example.com and if it fails, then as foo.example.com
{
    "name": "domain-search",
    "data": "user.net"
},
```

Рис. 3.3: Замена шаблона domain-name

Затем создаю собственную конфигурацию dhcp-сети, используя примеры, предоставленные в самом файле. Задаю адрес подсети, диапазон адресов для распределения клиентам, адрес маршрутизатора и broadcast-адрес (рис. [3.4]).

```
"subnet4": [
    {
        // This defines the whole subnet. Kea will use this information to
        // determine where the clients are connected. This is the whole
        // subnet in your network.

        // Subnet identifier should be unique for each subnet.
        "id": 1,

        // This is mandatory parameter for each subnet.
        "subnet": "192.168.1.0/24",

        // Pools define the actual part of your subnet that is governed
        // by Kea. Technically this is optional parameter, but it's
        // almost always needed for DHCP to do its job. If you omit it,
        // clients won't be able to get addresses, unless there are
        // host reservations defined for them.
        "pools": [ { "pool": "192.168.1.30 - 192.168.1.199" } ],

        // These are options that are subnet specific. In most cases,
        // you need to define at least routers option, as without this
        // option your clients will not be able to reach their default
        // gateway and will not have Internet connectivity.
        "option-data": [
            {
                // For each IPv4 subnet you most likely need to specify at
                // least one router.
                "name": "routers",
                "data": "192.168.1.1"
            }
        ],
    },
]
```

Рис. 3.4: Создание собственной dhcp-сети

Настраиваю привязку dhcpcd к интерфейсу eth1 машины server (рис. [3.5]).

```
"interfaces-config": {
    // See section 8.2.4 for more details. You probably want to add just
    // interface name (e.g. "eth0" or specific IPv4 address on that
    // interface (e.g. "eth0/192.0.2.1").
    "interfaces": [ "eth1" ]
```

Рис. 3.5: Привязка dhcpcd к eth1

Проверяю правильность проделанных действий (рис. [3.6]).

Рис. 3.6: Проверка

Перезагружаю конфигурацию `dhcpd` и разрешаю загрузку dhcp-сервера при запуске системы (рис. [3.7]).

```
[root@server ~]# systemctl --system daemon-reload
[root@server ~]# 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 ~]#
```

Рис. 3.7: Перезагрузка и включение автозапуска DHCP-сервера

Добавляю A запись в конец файла прямой зоны и PTR запись в конце файла обратной зоны (рис. [3.8]), (рис. [3.9]).

```
GNU nano 5.6.1                               /var/named/master/fz/user.net
$TTL 1D
@      IN SOA  @ server.user.net. (
                      2024072700      ; serial
                      1D              ; refresh
                      1H              ; retry
                      1W              ; expire
                      3H )            ; minimum
NS      @
A      192.168.1.1
$ORIGIN user.net.
server A      192.168.1.1
ns     A      192.168.1.1
dhcp   A      192.168.1.1
```

Рис. 3.8: Изменения в файле прямой зоны

```
Open ▾ 192.168.1
/var/named/master/rz
1 $TTL 1D
2 @      IN SOA  @ server.user.net. (
3                      2023072700      ; serial
4                      1D              ; refresh
5                      1H              ; retry
6                      1W              ; expire
7                      3H )            ; minimum
8      NS      @
9      A      192.168.1.1
10     PTR     server.user.net.
11 $ORIGIN 1.168.192.in-addr.arpa.
12 1    PTR     server.user.net.
13 1    PTR     ns.user.net
14 1    PTR     dhcp.user.net
```

Рис. 3.9: Изменения в файле обратной зоны

Перезапускаю named (рис. [3.10]).

```
[root@server ~]# systemctl restart named
[root@server ~]#
```

Рис. 3.10: Перезапуск named

Обращаюсь к dhcp-серверу по имени и получаю ответ (рис. [3.11]).

```
[root@server ~]# ping dhcp.user.net
PING dhcp.user.net (192.168.1.1) 56(84) bytes of data.
64 bytes from dhcp.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=1 ttl=64 time=0.099 ms
64 bytes from server.user.net (192.168.1.1): icmp_seq=2 ttl=64 time=0.290 ms
64 bytes from ns.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=3 ttl=64 time=0.051 ms
64 bytes from ns.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=4 ttl=64 time=0.048 ms
64 bytes from server.user.net (192.168.1.1): icmp_seq=5 ttl=64 time=0.059 ms
64 bytes from dhcp.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=6 ttl=64 time=0.064 ms
64 bytes from ns.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=7 ttl=64 time=0.085 ms
64 bytes from server.user.net (192.168.1.1): icmp_seq=8 ttl=64 time=0.066 ms
64 bytes from dhcp.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=9 ttl=64 time=0.062 ms
64 bytes from ns.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=10 ttl=64 time=0.050 ms
64 bytes from ns.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=11 ttl=64 time=0.061 ms
64 bytes from dhcp.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=12 ttl=64 time=0.062 ms
64 bytes from ns.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=13 ttl=64 time=0.061 ms
64 bytes from server.user.net (192.168.1.1): icmp_seq=14 ttl=64 time=0.064 ms
64 bytes from dhcp.user.net.1.168.192.in-addr.arpa (192.168.1.1): icmp_seq=15 ttl=64 time=0.068 ms
64 bytes from server.user.net (192.168.1.1): icmp_seq=16 ttl=64 time=0.063 ms
```

Рис. 3.11: Обращение к dhcp-серверу

Вношу изменения в настройки межсетевого экрана узла server, тем самым разрешая работу с dhcp (рис. [3.12]).

```
[root@server ~]# firewall-cmd --list-services
cockpit dhcpcv6-client dns ssh
[root@server ~]# firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqps apcupsd audit ausweisapp2 bacula bacula-client bareos-director bareos-filedaemon bareos-storage bb bpg bitcoin bitcoin-rpc bitcoin-testnet bitcoind-testnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent cockpit collectd condor-collector cratedb ctdb dds-multicast dds-unicast dhcp dhcpcv6 dhcpcv6-client distcc dns 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-ftp galera ganglia-client ganglia-master git gpd grafana graphite high-availability http http3 https ident imap imaps ipfs ipp ipp-client ipsec ircs iscsi-target isns jenkins kadmin kdeconnect kerberos kibana klogin kpasswd kshell kube-api kube-apiserver kube-control-plane kube-control-plane-secure kube-controller-manager kube-controller-manager-secure kube-nodeport-services kube-scheduler kube-scheduler-secure kube-worker kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr llmnr-client llmnr-tcp llmnr-udp managesieve matrix mdns memcache minidlna mongodb mosh mountd mqtt mqtt-tls ms-wbt mssql murmur mysql nbd nebula netbios-ns netdata-dashboard nfs nfs3 nmea-0183 ntp nut opencv ovirt-imageio ovirt-storageconsole plesk pmcd pmwebapi pop3 pop3s postgresql privoxy prometheus prometheus-node-exporter proxy-dhcp ps2link ps3netsrvr ptp pulseaudio puppetmaster quassel radius redis redis-gateway rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba-client samba-dr sane sip sips sld smtp smtp-submission smtps smtp smptlts smptlts-trap smpttrap spiderOak-lansync spotify-sync squid ssdp ssh steam-streaming svdrp svr syncthing syncthing-gui syncthing-relay synergy syslog syslog-tls telnet tentacle tftp tile38 tinc tor-socks transmission-client upnp-client vdsm vnc-server warpinator wbem-http wbem-https wireguard ws-discovery ws-discovery-client ws-discovery-tcp ws-discovery-udp wsmans xmpp-bosh xmpp-client xmpp-total xmpp-server zabbix-agent zabbix-server zerotier
[root@server ~]# firewall-cmd --add-service=dhcp
success
[root@server ~]# firewall-cmd --add-service=dhcp --permanent
success
[root@server ~]#
```

Рис. 3.12: Внесение изменений в настройки межсетевого экрана

Восстановливаю контекст безопасности SELinux (рис. [3.13]).

```
[root@server ~]# restorecon -vR /etc
Relabeled '/etc/sysconfig/network-scripts/ifcfg-eth1' from unconfined_u:object_r:user_t:s0 to unconfined_u:object_r:net_conf_t:s0
[root@server ~]# restorecon -vR /var/named
[root@server ~]# restorecon -vR /var/lib/kea/
[root@server ~]#
```

Рис. 3.13: Восстановление контекста безопасности

В дополнительном окне терминала запускаю мониторинг происходящих в

системе процессов, а в основном запускаю dhcp-сервер(Это я не скринил). Как мы видим запуск прошёл успешно (рис. [3.14]).

```
[root@server ~]# tail -f /var/log/messages
Nov 16 00:28:57 server named[44692]: timed out resolving 'ns3.fastly.net/AAAA/IN': 192.168.1.1#53
Nov 16 00:28:57 server named[44692]: timed out resolving 'ns3.fastly.net/AAAA/IN': 192.168.1.1#53
Nov 16 00:28:57 server named[44692]: timed out resolving 'ns1.fastly.net/AAAA/IN': 192.168.1.1#53
Nov 16 00:28:57 server named[44692]: timed out resolving 'ns4.fastly.net/AAAA/IN': 192.168.1.1#53
Nov 16 00:28:57 server named[44692]: timed out resolving 'ns2.fastly.net/AAAA/IN': 192.168.1.1#53
Nov 16 00:28:57 server named[44692]: timed out resolving 'ns4.fastly.net/AAAA/IN': 192.168.1.1#53
Nov 16 00:28:57 server named[44692]: timed out resolving 'ns2.fastly.net/AAAA/IN': 192.168.1.1#53
Nov 16 00:28:57 server named[44692]: timed out resolving 'ns1.fastly.net/AAAA/IN': 192.168.1.1#53
Nov 16 00:35:12 server systemd[1]: Starting Hostname Service...
Nov 16 00:35:12 server systemd[1]: Started Hostname Service.
Nov 16 00:35:42 server systemd[1]: systemd-hostnamed.service: Deactivated successfully.
Nov 16 00:35:55 server systemd[1]: Started Kea DHCPv4 Server.
Nov 16 00:35:55 server kea-dhcp4[44781]: 2025-11-16 03:35:55.327 INFO [kea-dhcp4.dhcp4/44781.140249854556288] DHCP4_STARTING Kea DHCPv4 server version 2.6.4 (stable) starting
Nov 16 00:35:55 server kea-dhcp4[44781]: 2025-11-16 03:35:55.330 INFO [kea-dhcp4.commands/44781.140249854556288] COMMAND_RECEIVED Received command 'config-set'
```

Рис. 3.14: Проверка корректной работы dhcp-сервера

3.3 Анализ работы DHCP-сервера

Создаю скрипт “01-routing.sh” для того, чтобы весь трафик на client шёл через eth1 (рис. [3.15]).

```
#!/bin/bash

echo "Provisioning script $0"
|
nmcli connection modify "System eth1" ipv4.gateway "192.168.1.1"
nmcli connection up "System eth1"

nmcli connection modify eth0 ipv4.never-default true
nmcli connection modify eth0 ipv6.never-default true

nmcli connection down eth0
nmcli connection up eth0

# systemctl restart NetworkManager
```

Рис. 3.15: 01-routing.sh

Также автоматизирую этот процесс, внося изменения в Vagrantfile для того, чтобы скрипт запускался при каждом запуске машины client (рис. [3.16]).

```

config.vm.define "client", autostart: false do |client|
  client.vm.box = "rocky9"
  client.vm.hostname = 'client'

  client.vm.boot_timeout = 1440

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

  client.vm.network :private_network,
    type: "dhcp",
    virtualbox_intnet: true

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

  client.vm.provision "client_routing",
    type: "shell",
    preserve_order: true,
    run: "always",
    path: "provision/client/01-routing.sh"

```

Рис. 3.16: Внесение изменений в Vagrantfile

После загрузки машины client в файле “**kea-leases4.csv**” появились записи, свидетельствующие о том, что трафик идет по тому интерфейсу, по которому было задумано (рис. [3.17]).

```

GNU nano 5.6.1
/var/lib/kea/kea-leases4.csv
address,hwaddr,client_id,valid_lifetime,expire,subnet_id,fqdn_fwd,fqdn_rev,hostname,state,user_context,pool_id
192.168.1.30,08:00:27:ad:80:1a,01:08:00:27:ad:80:1a,3600,1763258335,1,0,0,client,0,,0
192.168.1.30,08:00:27:ad:80:1a,01:08:00:27:ad:80:1a,3600,1763258350,1,0,0,client,0,,0
192.168.1.30,08:00:27:ad:80:1a,01:08:00:27:ad:80:1a,3600,1763258358,1,0,0,client,0,,0

```

Рис. 3.17: Информация в kea-leases4.csv

Далее вывожу информацию об имеющихся интерфейсах на клиенте и получаю всю информацию о eth0 и eth1 (рис. [3.18]).

```

user@client ~]$ 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:fe69:8d%eth0  prefixlen 64  scopeid 0x20<link>
                  ether 08:00:27:69:0a:8d  txqueuelen 1000  (Ethernet)
                    RX packets 918  bytes 107628 (105.1 Kib)
                    RX errors 0  dropped 0  overruns 0  frame 0
                    TX packets 791  bytes 126986 (124.0 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:feed:801a  prefixlen 64  scopeid 0x20<link>
                  ether 08:00:27:ad:80:1a  txqueuelen 1000  (Ethernet)
                    RX packets 200  bytes 29324 (28.6 Kib)
                    RX errors 0  dropped 0  overruns 0  frame 0
                    TX packets 1164  bytes 96353 (94.0 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

[user@client ~]$ 

```

Рис. 3.18: Информация о сетевых интерфейсах client

- **eth0** – основной интерфейс, который находится в активном статусе, он запущен и поддерживает broadcast и multicast. Также мы имеем его адрес(10.0.2.15);
- **eth1** – доп. сетевой интерфейс с теми же характеристиками, но другим адресом(192.168.1.30);
- **lo** – виртуальный интерфейс обратной связи, использующийся для внутренней связи в системе.

3.4 Настройка обновления DNS-зоны

Создаю ключ на сервере Bind9 машины server (рис. [3.19]).

```

[root@server ~]# mkdir -p /etc/named/keys
[root@server ~]# tsig-keygen -a HMAC-SHA512 DHCP_UPDATER > /etc/named/keys/dhcp_updater.key
[root@server ~]# 

```

Рис. 3.19: Создание ключа

Просматриваю файл dhcp_updater.key (рис. [3.20]).

```
[root@server ~]# cat /etc/named/keys/dhcp_updater.key
key "DHCP_UPDATER" {
    algorithm hmac-sha512;
    secret "04ywtrL72FkmB+2luY+zyu5HIAxUbUCVxqiiIUU+gu+9UUfvrIur50hzMAIS
UiTwaoFC+yDh55Pxp9U3Y0tox4g==";
};

[root@server ~]#
```

Рис. 3.20: dhcp_updater.key

Меняю права доступа (рис. [3.21]).

```
[root@server ~]# chown -R named:named /etc/named/keys
[root@server ~]#
```

Рис. 3.21: Изменения прав доступа

В конце файла named.conf добавляю строку для подключения ключа (рис. [3.22]).

```

GNU nano 5.6.1                               /etc/named.conf
recursion yes;

dnssec-enable no;
dnssec-validation no;

managed-keys-directory "/var/named/dynamic";
geoip-directory "/usr/share/GeoIP";

pid-file "/run/named/named.pid";
session-keyfile "/run/named/session.key";

/* https://fedoraproject.org/wiki/Changes/CryptoPolicy */
include "/etc/crypto-policies/back-ends/bind.config";
};

logging {
    channel default_debug {
        file "data/named.run";
        severity dynamic;
    };
};

zone "." IN {
    type hint;
    file "named.ca";
};

include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
include "/etc/named/user.net";
include "/etc/named/keys/dhcp_updater.key"; █

```

Рис. 3.22: Подключение ключа

В файле user.net вношу изменения, позволяющие зоне обновляться (рис. [3.23]).

```

.6
.7 zone "user.net" IN {
.8     type master;
.9     file "master/fz/user.net";
.0     update-policy {
.1         grant DHCP_UPDATER wildcard *.user.net A DHCID;
.2     };
.3 };
.4 zone "1.168.192.in-addr.arpa" IN {
.5     type master;
.6     file "master/rz/192.168.1";
.7     update-policy {
.8         grant DHCP_UPDATER wildcard *.1.168.192.in-
.9             addr.arpa PTR DHCID; █
.0     };
.1 };

```

Рис. 3.23: Изменения в user.net

Произвожу проверку конфиговского файла и перезапускаю DNS-сервер (рис. [3.24]).

```
[root@server ~]# named-checkconf  
/etc/named.conf:35: option 'dnssec-enable' is obsolete and should be removed  
d  
[root@server ~]# systemctl restart named  
[root@server ~]#
```

Рис. 3.24: Проверка и перезапуск DNS-сервера

Формирую ключ для Kea. Для этого создаю файл “**tsig-keys.json**” и переношу туда ранее созданный ключ (рис. [3.25]).

```
GNU nano 5.6.1          /etc/kea/tsig-keys.json  
"tsig-keys": [  
    {  
        "name": "DHCP_UPDATER",  
        "algorithm": "hmac-sha512",  
        "secret":  
            "04ywtrL72FkmB+2luY+zyu5HIAxUCVxqiiIUU+gu+9UUUfvrIur50h"  
    }  
,
```

Рис. 3.25: tsig-keys.json

Сменяю владельца и поправляю права доступа (рис. [3.26]).

```
[root@server ~]# chown kea:kea /etc/kea/tsig-keys.json  
[root@server ~]# chmod 640 /etc/kea/tsig-keys.json  
[root@server ~]#
```

Рис. 3.26: Смена владельца и прав доступа

Далее настройка будет производится в файле “**kea-dhcp-ddns.conf**” (рис. [3.27]).

```

    // See section 11 for examples and details description.
  "DhcpDdns": {
    {
      "ip-address": "127.0.0.1",
      "port": 53001,
      "control-socket": {
        "socket-type": "unix",
        "socket-name": "/run/kea/kea-ddns-ctrl-socket"
      },
      <?include "/etc/kea/tsig-keys.json"?>

      "forward-ddns" : {
        "ddns-domains" : [
          {
            "name": "user.net.",
            "key-name": "DHCP_UPDATER",
            "dns-servers": [
              { "ip-address": "192.168.1.1" }
            ]
          }
        ]
      },
      "reverse-ddns" : {
        "ddns-domains" : [
          {
            "name": "1.168.192.in-addr.arpa",
            "key-name": "DHCP_UPDATER",
            "dns-servers": [
              { "ip-address": "192.168.1.1" }
            ]
          }
        ]
      }
    }
  }
}

```

Рис. 3.27: Изменения в kea-dhcp-ddns.conf

Меняю владельца файла и проверяю файл на наличие синтаксических ошибок(их не обнаружилось) и запускаю службу ddns (рис. [3.28]).

```

[root@server ~]# chown kea:kea /etc/kea/kea-dhcp-ddns.conf
[root@server ~]# kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
2025-11-16 05:20:15.869 WARN  [kea-dhcp-ddns_d2-to-dns/45227.139798702647168] DHCP_DDNS_CONFIG_SYNTAX_WARNING DHCP-DDNS server configuration syntax warning: /etc/kea/tsig-keys.json@6.93: Extraneous comma. A piece of configuration may have been omitted.
2025-11-16 05:20:15.870 INFO  [kea-dhcp-ddns_d2-to-dns/45227.139798702647168] DCTL_CONFIG_CHECK_COMPLETE server has completed configuration check: listening on 127.0.0.1 port 53001, using UDP, result: success()
[root@server ~]# systemctl enable --now kea-dhcp-ddns.service
Created symlink /etc/systemd/system/multi-user.target.wants/kea-dhcp-ddns.service.
[root@server ~]#

```

Рис. 3.28: Смена владельца и проверка файла, запуск службы ddns

Проверяю статус работы службы (рис. [3.29]).

```
[root@server ~]# systemctl status kea-dhcp-ddns.service
● kea-dhcp-ddns.service - Kea DHCP-DDNS Server
   Loaded: loaded (/usr/lib/systemd/system/kea-dhcp-ddns.service; enabled)
   Active: active (running) since Sun 2025-11-16 05:21:20 MSK; 1min
     Docs: man:kea-dhcp-ddns(8)
 Main PID: 45264 (kea-dhcp-ddns)
    Tasks: 5 (limit: 48821)
      Memory: 3.7M
        CPU: 21ms
       CGroup: /system.slice/kea-dhcp-ddns.service
           └─45264 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Nov 16 05:21:20 server systemd[1]: Started Kea DHCP-DDNS Server.
Nov 16 05:21:20 server kea-dhcp-ddns[45264]: 2025-11-16 05:21:20.225
Nov 16 05:21:20 server kea-dhcp-ddns[45264]: 2025-11-16 05:21:20.226
Nov 16 05:21:20 server kea-dhcp-ddns[45264]: INFO COMMAND_ACCEPTOR_START
Nov 16 05:21:20 server kea-dhcp-ddns[45264]: INFO DCTL_CONFIG_COMPLETE
Nov 16 05:21:20 server kea-dhcp-ddns[45264]: INFO DHCP_DDNS_STARTED
lines 1-17/17 (END)
```

Рис. 3.29: Статус службы

Вношу изменения в файл kea-dhcp4.conf, тем самым добавляя разрешение на динамическое обновление dns-записей с локального узла прямой и обратной зон (рис. [3.30]).

```
},
"dhcp-ddns": {
  "enable-updates": true
},
"ddns-qualifying-suffix": "user.net",
"ddns-override-client-update": true,
```

// Kea supports control channel, which is a way to receive management
// commands while the server is running. This is a Unix domain socket that
// receives commands formatted in JSON, e.g. config-set (which sets new
// configuration), config-reload (which tells Kea to reload its
// configuration from file), statistic-get (to retrieve statistics) and many

Рис. 3.30: Изменения в kea-dhcp4.conf

Проверяю файл на наличие синтаксических ошибок (рис. [3.31]).

```
[root@server ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-11-16 05:44:13.291 INFO [kea-dhcp4.hosts/45366.140354818984064] HOSTS_BACKENDS_REGISTERED the following host backend types are available: mysql postgresql
2025-11-16 05:44:13.291 WARN [kea-dhcp4.dhcpsrv/45366.140354818984064] DHCPSRV_MT_DISABLED_QUEUE_CONTROL disabling dhcp queue control when multi-threading is enabled.
2025-11-16 05:44:13.292 INFO [kea-dhcp4.dhcpsrv/45366.140354818984064] DHCP4_RESERVATIONS_LOOKUP_FIRST_ENABLED Multi-threading is enabled and host reservations are always performed first.
2025-11-16 05:44:13.292 INFO [kea-dhcp4.dhcpsrv/45366.140354818984064] DHCPSRV_CFGMGR_NEW_SUBNET4 a new subnet has been added to configuration: 192.168.1.0/24 with params: tl=988, t2=1800, valid-lifetime=3600
2025-11-16 05:44:13.292 INFO [kea-dhcp4.dhcpsrv/45366.140354818984064] DHCPSRV_CFGMGR_SOCKET_TYPE_SELECT using socket type raw
2025-11-16 05:44:13.292 INFO [kea-dhcp4.dhcpsrv/45366.140354818984064] DHCPSRV_CFGMGR_ADD_IFACE listening on interface eth1
2025-11-16 05:44:13.292 INFO [kea-dhcp4.dhcpsrv/45366.140354818984064] DHCPSRV_CFGMGR_SOCKET_TYPE_DEFAULT "dhcp-socket-type" not specified , using default socket type raw
```

Рис. 3.31: Проверка файла после изменений

Затем перезапускаю dhcp-сервер и проверяю его статус (рис. [3.32]).

```
[root@server ~]# systemctl restart kea-dhcp4.service
[root@server ~]# systemctl status kea-dhcp4.service
● kea-dhcp4.service - Kea DHCPv4 Server
    Loaded: loaded (/usr/lib/systemd/system/kea-dhcp4.service; enabled; preset: disabled)
      Active: active (running) since Sun 2025-11-16 05:45:01 MSK; 7s ago
        Docs: man:kea-dhcp4(8)
     Main PID: 45374 (kea-dhcp4)
        Tasks: 7 (limit: 48821)
       Memory: 2.5M
          CPU: 29ms
         CGroup: /system.slice/kea-dhcp4.service
                   └─45374 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf

Nov 16 05:45:01 server systemd[1]: Started Kea DHCPv4 Server.
Nov 16 05:45:02 server kea-dhcp4[45374]: 2025-11-16 05:45:02.017 INFO [kea-dhcp4.dhcp4]
Nov 16 05:45:02 server kea-dhcp4[45374]: 2025-11-16 05:45:02.018 INFO [kea-dhcp4.commu...
lines 1-14/14 (END)
```

Рис. 3.32: Статус dhcp-сервера

На машине client переполучаю адрес (рис. [3.33]).

```
[root@client ~]# nmcli connection show
NAME           UUID                                  TYPE      DEVICE
System eth0    5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03  ethernet  eth0
System eth1    9c92fad9-6ecb-3e6c-eb4d-8a47cf50c04  ethernet  eth1
lo             ed18e23a-468a-41a2-ac02-0214c377b570  loopback  lo
eth0          52b757e4-3833-4753-b0eb-de0af7f26f57  ethernet  --
[root@client ~]# nmcli connection down "System eth1"
Connection 'System eth1' successfully deactivated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/3)
[root@client ~]# nmcli connection up "System eth1"
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/4)
```

Рис. 3.33: Переполучение адреса на клиенте

3.5 Анализ работы DHCP-сервера после настройки обновления DNS-зоны

Обращаюсь к client.user.net, получаю ответ похожий на те, что были в прошлой лабе (рис. [3.34]).

```
[root@client ~]# dig @192.168.1.1 client.user.net
; <>> DiG 9.16.23-RH <>> @192.168.1.1 client.user.net
; (1 server found)
; global options: +cmd
; Got answer:
; ->>HEADER<- opcode: QUERY, status: NXDOMAIN, id: 44211
; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1
;
; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 789512cfec0857ba010000006925b7833ec148d0b3e19146 (good)
; QUESTION SECTION:
;client.user.net.           IN      A
;
; AUTHORITY SECTION:
user.net.          10800   IN      SOA     user.net. server.user.net. 20251
11603 86400 3600 604800 10800
;
; Query time: 1 msec
; SERVER: 192.168.1.1#53(192.168.1.1)
; WHEN: Tue Nov 25 17:04:51 MSK 2025
; MSG SIZE rcvd: 115
```

Рис. 3.34: Обращение client.user.net

Видим, что запрос ушёл на server, а вернулся на тот самый адрес(192.168.1.1), который мы и ставили выше.

3.6 Внесение изменений в настройки внутреннего окружения виртуальной машины

Вношу изменения в настройки внутреннего окружения машины server, внося туда конфигурационные файлы dhcp (рис. [3.35]).

```
[user@server ~]$ sudo -
[sudo] password for user:
[root@server ~]# cd /vagrant/provision/server
[root@server server]# mkdir -p /vagrant/provision/server/dhcp/etc/kea
[root@server server]# cp -R /etc/kea/* /vagrant/provision/server/dhcp
/etc/kea/
[root@server server]#
```

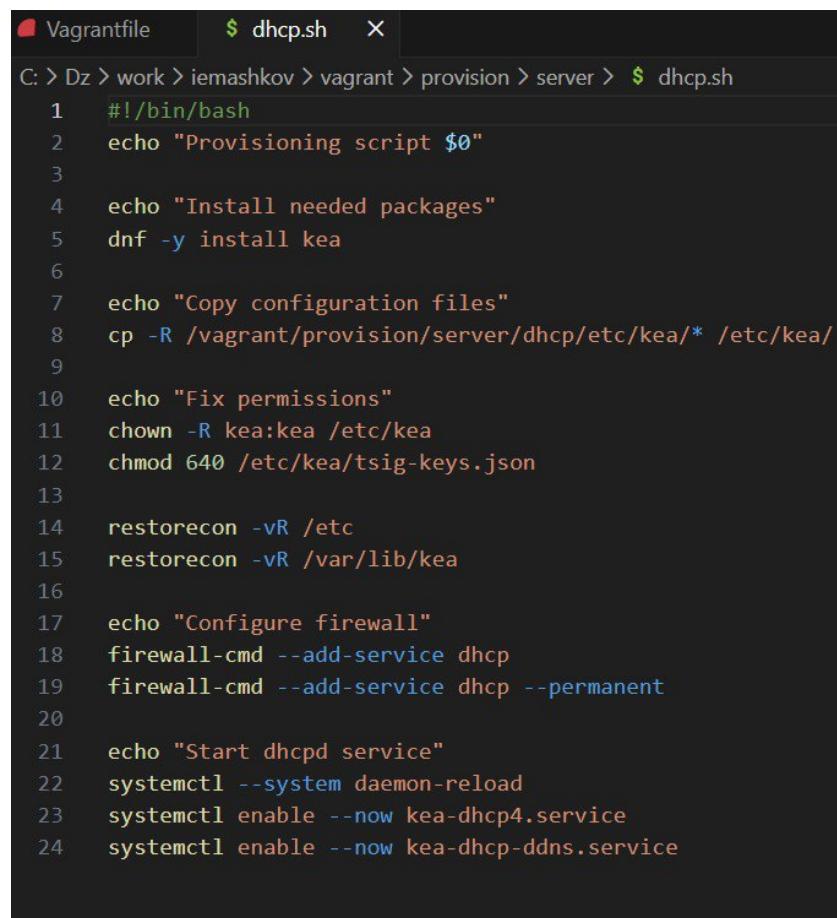
Рис. 3.35: Внесение изменений в настройки внутреннего окружения

Также заменяю конфигурационные файлы dns-сервера (рис. [3.36]).

```
[root@server server]# cd /vagrant/provision/server/dns/
[root@server dns]# cp -R /var/named/* /vagrant/provision/server/dns/var/named/
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/192.168.1'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/user.net'? yes
[root@server dns]# cp -R /etc/named/* /vagrant/provision/server/dns/etc/named/
cp: overwrite '/vagrant/provision/server/dns/etc/named/user.net'? yes
[root@server dns]#
```

Рис. 3.36: Замена конфигов dns-сервера

В каталоге vagrant/provision/server создаю скрипт dhcp.sh (рис. [3.37]).



```
Vagrantfile    $ dhcp.sh    X
C: > Dz > work > iemashkov > vagrant > provision > server > $ dhcp.sh
1  #!/bin/bash
2  echo "Provisioning script $0"
3
4  echo "Install needed packages"
5  dnf -y install kea
6
7  echo "Copy configuration files"
8  cp -R /vagrant/provision/server/dhcp/etc/kea/* /etc/kea/
9
10 echo "Fix permissions"
11 chown -R kea:kea /etc/kea
12 chmod 640 /etc/kea/tsig-keys.json
13
14 restorecon -vR /etc
15 restorecon -vR /var/lib/kea
16
17 echo "Configure firewall"
18 firewall-cmd --add-service dhcp
19 firewall-cmd --add-service dhcp --permanent
20
21 echo "Start dhcpcd service"
22 systemctl --system daemon-reload
23 systemctl enable --now kea-dhcp4.service
24 systemctl enable --now kea-dhcp-ddns.service
```

Рис. 3.37: dhcp.sh

Настройка автоматизации отработки этого скрипта при запуске машины server производится путём внесения изменений в файл Vagrantfile (рис. [3.38]).

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

Рис. 3.38: Изменения в файле Vagrantfile

4 Выводы

Во время выполнения лабораторной работы я приобрёл практические навыки по установке и конфигурированию DHCP-сервера.

Список литературы

Администрирование сетевых подсистем