

Лабораторная работа №3

Дисциплина: Администрирование сетевых подсистем

Комягин Андрей Николаевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Контрольные вопросы	18
4	Выводы	20
	Список литературы	21

Список иллюстраций

2.1	Запуск ОС	6
2.2	Установка Kea	6
2.3	Настройка файла конфигурации. Domain-name	7
2.4	Domain-name-servers	7
2.5	Subnet4	7
2.6	Перезапуск dhcp	7
2.7	Файл прямой DNS-зоны	8
2.8	Файл обратной DNS-зоны	8
2.9	Обращение к DHCP-серверу по имени	8
2.10	firewall-cmd –get-services	9
2.11	Добавление dhcp	9
2.12	Восстановление контекста безопасности	9
2.13	Запуск dhcp	10
2.14	Файл 01-routing.sh	10
2.15	Vagrantfile	10
2.16	Запуск client	11
2.17	Интерфейсы	11
2.18	Выданные адреса	12
2.19	Создание ключа	12
2.20	Права доступа	12
2.21	Подключение в файле	12
2.22	Разрешение обновления	13
2.23	Перезапуск DNS-сервера	13
2.24	Формирование ключа	13
2.25	kea-dhcp-ddns.conf	14
2.26	Запуск dhcp-ddns	15
2.27	kea-dhcp4.conf	15
2.28	Запуск dhcp	15
2.29	Переполучение адреса	16
2.30	edzhibitskaya.net.jnl	16
2.31	Запись о клиенте	16
2.32	Каталог DHCP	16
2.33	Замена файлов	16
2.34	Создание файла	17
2.35	Файл dhcp.sh	17
2.36	Выключение	17

Список таблиц

1 Цель работы

Изучение принципов работы DHCP, приобретение навыков по установке и конфигурированию DHCP-сервера.

2 Выполнение лабораторной работы

Загрузим операционную систему с помощью Vagrant. После загрузки устанавливаем Kea (рис. 2.1 и рис. 2.2).

```
C:\work\ankomyagin\vagrant>vagrant up server
```

Рис. 2.1: Запуск ОС

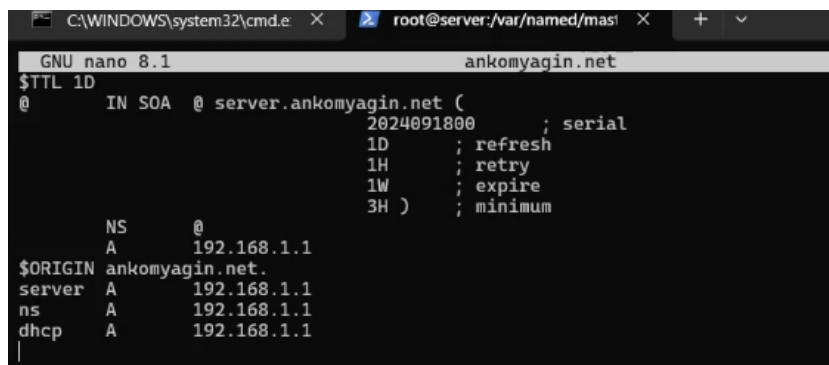
```
[root@server etc]# dnf -y install kea
Rocky Linux 10 - BaseOS                               4.9 kB/s | 3.9 kB   00:00
Rocky Linux 10 - AppStream                             6.4 kB/s | 3.9 kB   00:00
Rocky Linux 10 - CRB                                   5.6 kB/s | 3.9 kB   00:00
Rocky Linux 10 - Extras                               1.9 kB/s | 3.1 kB   00:01
Dependencies resolved.
=====
Package                                Architecture Version      Repository    Size
=====
Installing:
kea                                     x86_64      2.6.3-1.el10_0 baseos        1.3 M
Installing dependencies:
kea-libs                               x86_64      2.6.3-1.el10_0 baseos        3.1 M
libpq                                   x86_64      16.8-2.el10_0 baseos        255 k
log4cplus                              x86_64      2.1.1-8.el10  baseos        351 k
mariadb-connector-c                     x86_64      3.4.4-1.el10  baseos        206 k
mariadb-connector-c-config              noarch      3.4.4-1.el10  baseos         8.9 k
Transaction Summary
=====
Install 6 Packages

Total download size: 5.3 M
Installed size: 19 M
Downloading Packages:
(1/6): libpq-16.8-2.el10_0.x86_64.rpm                359 kB/s | 255 kB   00:00
(2/6): kea-libs-2.6.3-1.el10_0.x86_64.rpm             505 kB/s | 3.1 MB   00:06
(3/6): kea-2.6.3-1.el10_0.x86_64.rpm                  146 kB/s | 1.3 MB   00:09
```

Рис. 2.2: Установка Kea

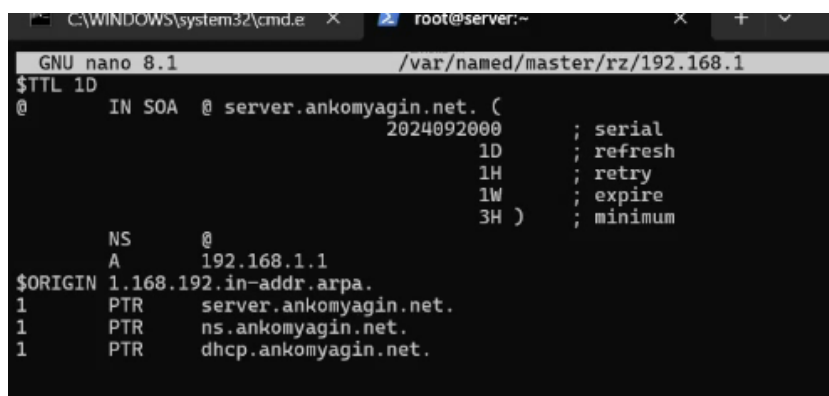
На всякий случай сохраняем файл конфигурации (копируем его), открываем на редактирование и меняем шаблон. Указываем имя, адрес подсети, диапазон адресов для распределения клиентам, адрес маршрутизатора и broadcast-адрес. Также настраиваем привязку dhcprd к интерфейсу eth1 (рис. 2.3, рис. 2.4 и рис. 2.5).

Редактируем файлы прямой DNS-зоны и обратной, добавляем запись для DHCP-сервер(рис. 2.7 и рис. 2.8).



```
C:\WINDOWS\system32\cmd.e X root@server:/var/named/master/ ankomyagin.net
GNU nano 8.1
$TTL 1D
@ IN SOA @ server.ankomyagin.net (
    2024091800 ; serial
    1D ; refresh
    1H ; retry
    1W ; expire
    3H ) ; minimum
NS @
A 192.168.1.1
$ORIGIN ankomyagin.net.
server A 192.168.1.1
ns A 192.168.1.1
dhcp A 192.168.1.1
```

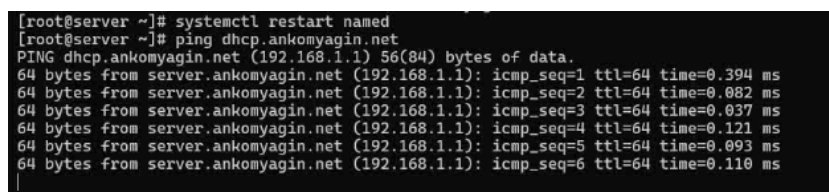
Рис. 2.7: Файл прямой DNS-зоны



```
C:\WINDOWS\system32\cmd.e X root@server:~/ /var/named/master/rz/192.168.1
GNU nano 8.1
$TTL 1D
@ IN SOA @ server.ankomyagin.net. (
    2024092000 ; serial
    1D ; refresh
    1H ; retry
    1W ; expire
    3H ) ; minimum
NS @
A 192.168.1.1
$ORIGIN 1.168.192.in-addr.arpa.
1 PTR server.ankomyagin.net.
1 PTR ns.ankomyagin.net.
1 PTR dhcp.ankomyagin.net.
```

Рис. 2.8: Файл обратной DNS-зоны

Перезапускаем named, проверяем, что обращение по имени возможно(рис. 2.9).



```
[root@server ~]# systemctl restart named
[root@server ~]# ping dhcp.ankomyagin.net
PING dhcp.ankomyagin.net (192.168.1.1) 56(84) bytes of data:
64 bytes from server.ankomyagin.net (192.168.1.1): icmp_seq=1 ttl=64 time=0.394 ms
64 bytes from server.ankomyagin.net (192.168.1.1): icmp_seq=2 ttl=64 time=0.082 ms
64 bytes from server.ankomyagin.net (192.168.1.1): icmp_seq=3 ttl=64 time=0.037 ms
64 bytes from server.ankomyagin.net (192.168.1.1): icmp_seq=4 ttl=64 time=0.121 ms
64 bytes from server.ankomyagin.net (192.168.1.1): icmp_seq=5 ttl=64 time=0.093 ms
64 bytes from server.ankomyagin.net (192.168.1.1): icmp_seq=6 ttl=64 time=0.110 ms
```

Рис. 2.9: Обращение к DHCP-серверу по имени

Затем вносим изменения в настройки межсетевого экрана узла server, разрешив работу с DHCP(рис. 2.10 и рис. 2.11) и восстанавливаем контекст безопасности в SELinux(рис. 2.12)


```

--- dhcp.ankomyagin.net ping statistics ---
154 packets transmitted, 154 received, 0% packet loss, time 158090ms
rtt min/avg/max/mdev = 0.037/0.132/1.980/0.161 ms
[root@server ~]# firewall-cmd --list-services

cockpit dhcpv6-client dns ssh
[root@server ~]#
[root@server ~]# firewall-cmd --get-services
0-AD RH-Satellite-6 RH-Satellite-6-capsule afp alvr amanda-client amanda-k5-client amqp amqps
anno-1602 anno-1800 apcupsd aseqnet audit ausweisapp2 bacula bacula-client bareos-director b
areos-filedaemon bareos-storage bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rp
c bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent civilization-iv civilizati
ion-v cockpit collectd condor-collector cratedb ctdb dds dds-multicast dds-unicast dhcp dhcpv
6 dhcpv6-client distcc dns dns-over-quic dns-over-tls docker-registry docker-swarm dropbox-la
nsync elasticsearch etcd-client etcd-server factorio finger foreman foreman-proxy freeipa-4 f
reeipa-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 iperf2 iperf
3 ipfs ipp ipp-client ipsec irc ircs iscsi-target isns jenkins kadmin kdeconnect kerberos kib
ana klogin kpasswd kprop kshell kube-api kube-apiserver kube-control-plane kube-control-plane
-secure kube-controller-manager kube-controller-manager-secure kube-nodeport-services kube-sc
heduler 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 matr
ix mdns memcache minecraft minidlna mndp mongodb mosh mountd mpd mqtt mqtt-tls ms-wbt mssql m
urmur mysql nbd nebula need-for-speed-most-wanted netbios-ns netdata-dashboard nfs nfs3 nmea-
0183 nrpe ntp nut opentelemetry openvpn ovirt-imageio ovirt-storageconsole ovirt-vmconsole pl
ex pmcd pmpoxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy prometheus prometheus-node-e
xporter proxy-dhcp ps2link ps3netrv ptp pulseaudio puppetmaster quassel radius radsec rdp re
dis redis-sentinel rootd rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba-client samb
a-dc sane settlers-history-collection sip sips slimevr slp smtp smtp-submission smtps snmp sn
mp-tls snmp-tls-trap snmptrap spideroak-lansync spotify-sync squid ssdp ssh statsrv steam-lan-t
ransfer steam-streaming stellaris stronghold-crusader stun stuns submission supertuxkart svdr
p svn syncthing syncthing-gui syncthing-relay synergy syscomlan syslog syslog-tls telnet tent
acle terraria tftp tile38 tinc tor-socks transmission-client turn turns upnp-client vdsd vnc-
server vrrp warpinator wbem-http wbem-https wireguard ws-discovery ws-discovery-client ws-dis
covery-host ws-discovery-tcp ws-discovery-udp wsdd wsdd-http wsman wsmans xdmcp xmpp-bosh xmp
p-client xmpp-local xmpp-server zabbix-agent zabbix-java-gateway zabbix-server zabbix-trapper
zabbix-web-service zero-k zerotier
[root@server ~]#

```

Рис. 2.10: firewall-cmd --get-services

```

[root@server ~]# firewall-cmd --add-service=dhcp
success
[root@server ~]# firewall-cmd --add-service=dhcp --permanent
success
[root@server ~]#

```

Рис. 2.11: Добавление dhcp

```

[root@server ~]# restorecon -vR /etc
Relabeled /etc/NetworkManager/system-connections/eth1.nmconnection from unconfined_u:object_r
:user_tmp_t:s0 to unconfined_u:object_r:NetworkManager_etc_rw_t:s0
[root@server ~]# restorecon -vR /var/named
[root@server ~]# restorecon -vR /var/lib/kea/
[root@server ~]#

```

Рис. 2.12: Восстановление контекста безопасности

Наконец, в еще одном терминале запускаем просмотр лога ошибок, а в основном терминале запускаем сам сервис(рис. 2.13).

```
[root@server ~]# systemctl start kea-dhcp4.service
[root@server ~]# systemctl status kea-dhcp4
● kea-dhcp4.service - Kea DHCPv4 Server
   Loaded: loaded (/usr/lib/systemd/system/kea-dhcp4.service; enabled; preset: disabled)
   Active: active (running) since Sat 2025-09-20 13:23:42 UTC; 1min 20s ago
  Invocation: 733b792f7d2b4c0299fae96f5c706855
     Docs: man:kea-dhcp4(8)
    Main PID: 12401 (kea-dhcp4)
      Tasks: 7 (limit: 10397)
    Memory: 2.6M (peak: 6M)
       CPU: 160ms
    CGroup: /system.slice/kea-dhcp4.service
            └─12401 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf

Sep 20 13:23:42 server.ankomyagin.net systemd[1]: Started kea-dhcp4.service - Kea DHCPv4 Ser
Sep 20 13:23:43 server.ankomyagin.net kea-dhcp4[12401]: 2025-09-20 13:23:43.171 INFO [kea-d
Sep 20 13:23:43 server.ankomyagin.net kea-dhcp4[12401]: 2025-09-20 13:23:43.173 INFO [kea-d
```

Рис. 2.13: Запуск dhcp

Переходим к анализу работы сервера.

Перед запуском виртуальной машины client в каталоге с проектом подкаталоге client создаем файл 01-routing.sh, добавляем скрипт настройки NetworkManager, чтобы весь трафик client шёл по умолчанию через eth1(рис. 2.14). Добавляем соответствующий скрипт в Vagrantfile(рис. 2.15).

```
GNU nano 8.1 01-routing.sh
#!/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. 2.14: Файл 01-routing.sh

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

Рис. 2.15: Vagrantfile

Запускаем машину client с внесенными изменениями(рис. 2.16). На машине server на терминале с мониторингом можно увидеть записи о подключении

к виртуальной внутренней сети узла client и выдачи ему IP-адреса из соответствующего диапазона адресов.

```
PS C:\Users\Andrey> cd /
PS C:\> cd .\work\ankomyagin\vagrant\
PS C:\work\ankomyagin\vagrant> vagrant up client --provision
Bringing machine 'client' up with 'virtualbox' provider...
==> client: Clearing any previously set forwarded ports...
==> client: Fixed port collision for 22 => 2222. Now on port 2200.
==> client: Clearing any previously set network interfaces...
==> client: Preparing network interfaces based on configuration...
      client: Adapter 1: nat
      client: Adapter 2: intnet
==> client: Forwarding ports...
      client: 22 (guest) => 2200 (host) (adapter 1)
==> client: Running 'pre-boot' VM customizations...
```

Рис. 2.16: Запуск client

В терминале запущенной машины смотрим информацию об имеющихся интерфейсах(рис. 2.17), а на сервере смотрим список адресов(рис. 2.18). Файл хранит информацию о выделенных DHCP адресах. Записи включают в себя IP-адрес, который был выделен клиенту, информацию о том кому и на какой срок выдан адрес, дату начала и окончания, MAC-адрес сетевого интерфейса, который был использован при получении IP-адреса, идентификатор клиента и имя хоста.

```
vagrant@client:~$ su - ankomyagin
Password:
Last login: Thu Sep 18 08:43:52 UTC 2025 on pts/1
[ankomyagin@client.ankomyagin.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 fd17:625c:f037:2:a00:27ff:feaa:ce23 prefixlen 64 scopeid 0x0<
global>
    inet6 fe80::a00:27ff:feaa:ce23 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:aa:ce:23 txqueuelen 1000 (Ethernet)
    RX packets 3354 bytes 308015 (300.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2213 bytes 273292 (266.8 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::fcab:a8ef:8ae8:66c9 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:de:98:49 txqueuelen 1000 (Ethernet)
    RX packets 34 bytes 3752 (3.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 149 bytes 15075 (14.7 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 2039 (1.9 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 17 bytes 2039 (1.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Рис. 2.17: Интерфейсы

```
ankomyagin@server:~$ cat /var/lib/kea/kea-leases4.csv
GNU nano 8.1 /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:de:98:49,01:08:00:27:de:98:49,3600,1753822609,1,0,0,client,0,,0
```

Рис. 2.18: Выданные адреса

Перейдем к настройке обновления DNS-зоны.

Создаем ключ на сервере с Bind9(рис. 2.19). Поправим права доступа и подключим ключ в файле(рис. 2.20 и рис. 2.21).

```
root@serv:~$ mkdir -p /etc/named/keys
mkdir: cannot create directory '/etc/named': Permission denied
[ankomyagin@server.ankomyagin.net ~]$ sudo -i
[root@server.ankomyagin.net ~]$ mkdir -p /etc/named/keys
[root@server.ankomyagin.net ~]$ tsig-keygen -a HMAC-SHA512 DHCP_UPDATER > /etc/named/keys/dhcp_updater.key
[root@server.ankomyagin.net ~]$ nani |
```

Рис. 2.19: Создание ключа

```
p_updater.key
[root@server.ankomyagin.net ~]$ chown -R named:named /etc/named/keys
[root@server.ankomyagin.net ~]$ nano /etc/named/keys/dhcp_updater.key
[root@server.ankomyagin.net ~]$ nano |
```

Рис. 2.20: Права доступа

```
};

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

include "/etc/named/ankomyagin.net";

^G Help      ^O Write Out  ^F Where Is   ^K Cut        ^T Execut
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justif
```

Рис. 2.21: Подключение в файле

Также разрешим обновление в файле /etc/named/edzhibitskaya.net (рис. 2.22).

```

zone "ankomyagin.net" IN {
    type master;
    file "master/fz/ankomyagin.net";
    update-policy {
        grant DHCP_UPDATER wildcard *.user.net A DHCPID;
    };
};

zone "1.168.192.in-addr.arpa" IN {
    type master;
    file "master/rz/192.168.1";
    update-policy {
        grant DHCP_UPDATER wildcard *.1.168.192.in-addr.arpa PTR DHCPID;
    };
};

```

Рис. 2.22: Разрешение обновления

Проверяем на наличие опечаток, исправляем и перезапускаем named (рис. 2.23).

```

[root@server.ankomyagin.net ~]# nano /etc/named/ankomyagin.net
/etc/named.conf:58: missing ';' before 'include'
[root@server.ankomyagin.net ~]# nano /etc/named.conf
[root@server.ankomyagin.net ~]# named-checkconf
[root@server.ankomyagin.net ~]# systemctl restart named

```

Рис. 2.23: Перезапуск DNS-сервера

Далее формируем ключ(рис. 2.24). Меняем владельца и поправляем права доступа.

```

GNU nano 8.1 /etc/kea/tsig-keys.json Modified
"tsig-keys": [
  {
    "name": "DHCP",
    "algorithm": "hmac-sha512",
    "secret": "dZMZHYSVS6GtgxtYZYHXtmiUG08oYL0iho+6V9Wj6dvRM9nq0p/2g6QoYWi9sxqp4yQ95DdUGYS0"
  }
],

```

Рис. 2.24: Формирование ключа

В файле /etc/kea/kea-dhcp-ddns.conf прописываем все настройки(рис. 2.25).

```

GNU nano 8.1 /etc/kea/kea-dhcp-ddns.co
{
  "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": "ankomyagin.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" }
          ]
        }
      ]
    },
    "loggers": [
      {
        "name": "kea-dhcp-ddns",
        "output_options": [
          {
            "output": "stdout",
            "pattern": "%-5p %m\n"
          }
        ],
        "severity": "INFO",
        "debuglevel": 0
      }
    ]
  }
}

```

Рис. 2.25: kea-dhcp-ddns.conf

Проверяем на наличие ошибок, меняем владельца “chown kea:kea /etc/kea/kea-dhcp-ddns.conf” и запускаем службу(рис. 2.26).

```
lines 1-18/18 (END)...skipping...
● kea-dhcp-ddns.service - Kea DHCP-DDNS Server
   Loaded: loaded (/usr/lib/systemd/system/kea-dhcp-ddns.service; enabled; preset: disabled)
   Active: active (running) since Sat 2025-09-20 15:09:34 UTC; 6s ago
     Invocation: 88c23c3e3416441287f8d925898999c4
       Docs: man:kea-dhcp-ddns(8)
    Main PID: 12950 (kea-dhcp-ddns)
      Tasks: 5 (limit: 10397)
     Memory: 1.7M (peak: 6M)
        CPU: 127ms
    CGroup: /system.slice/kea-dhcp-ddns.service
            └─12950 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Sep 20 15:09:34 server.ankomyagin.net systemd[1]: Started kea-dhcp-ddns.service - Kea DHCP-DDNS Serv
Sep 20 15:09:34 server.ankomyagin.net kea-dhcp-ddns[12950]: 2025-09-20 15:09:34.519 INFO [kea-dhcp-
12950, version: 2.6.3 (stable)
Sep 20 15:09:34 server.ankomyagin.net kea-dhcp-ddns[12950]: 2025-09-20 15:09:34.520 WARN [kea-dhcp-
DHCP-DDNS server configuration syntax warning: /etc/kea/tsig-keys.json:5.110: Extraneous comma, AP
Sep 20 15:09:34 server.ankomyagin.net kea-dhcp-ddns[12950]: INFO COMMAND_ACCEPTOR_START Starting to
ctrl-socket
Sep 20 15:09:34 server.ankomyagin.net kea-dhcp-ddns[12950]: INFO DCTL_CONFIG_COMPLETE server has co
Sep 20 15:09:34 server.ankomyagin.net kea-dhcp-ddns[12950]: INFO DHCP_DDNS_STARTED Kea DHCP-DDNS se
```

Рис. 2.26: Запуск dhcp-ddns

Кроме того добавляем изменения в конфигурационный файл /etc/kea/kea-dhcp4.conf(рис. 2.27). Проверяем на наличие ошибок и запускаем сервер(рис. 2.28).

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

Рис. 2.27: kea-dhcp4.conf

```
[root@server.ankomyagin.net ~]# systemctl restart kea-dhcp4.service
[root@server.ankomyagin.net ~]# 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 Sat 2025-09-20 15:12:57 UTC; 4s ago
     Invocation: 07222d9823604c4386e1cfedee32909
       Docs: man:kea-dhcp4(8)
    Main PID: 12974 (kea-dhcp4)
      Tasks: 7 (limit: 10397)
     Memory: 2.5M (peak: 5.9M)
        CPU: 197ms
    CGroup: /system.slice/kea-dhcp4.service
            └─12974 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf
```

Рис. 2.28: Запуск dhcp

На машине client переполучаем адрес, в каталоге прямой DNS-зоны появляется файл edzhbitskaya.net.jnl, в котором автоматически вносятся изменения записей зоны(рис. 2.29 и рис. 2.30).

```
[sudo] password for ankomyagin:
[root@client.ankomyagin.net ~]# nmcli connection down eth1
Connection 'eth1' successfully deactivated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/4)
[root@client.ankomyagin.net ~]# nmcli connection up eth1
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/6)
[root@client.ankomyagin.net ~]#
```

Рис. 2.29: Переполучение адреса

edzhibitskaya.net.jnl

Рис. 2.30: edzhibitskaya.net.jnl

Анализируем работу DHCP-сервера после настройки обновлений.

На машине client с помощью утилиты dig убедимся в наличии DNS-записи о клиенте в прямой DNS-зоне(рис. 2.31).

Запись о клиенте

Рис. 2.31: Запись о клиенте

Наконец внесем изменения в настройки окружения.

На виртуальной машине server в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создаем каталог dhcp, в который помещаем соответствующие подкаталоги конфигурационные файлы DHCP(рис. 2.32).

```
[root@server.ankomyagin.net ~]# cd
[root@server.ankomyagin.net ~]# cd /vagrant/provision/server
[root@server.ankomyagin.net server]# mkdir -p /vagrant/provision/server/dhcp/etc/kea
[root@server.ankomyagin.net server]# cp -R /etc/kea/* /vagrant/provision/server/dhcp/etc/kea/
[root@server.ankomyagin.net server]# cd /vagrant/provision/server/dns/
[root@server.ankomyagin.net dns]# cp -R /var/named/* /vagrant/provision/server/dns/var/named/
```

Рис. 2.32: Каталог DHCP

Затем заменим файл сервера(рис. 2.33).

```
[root@server.ankomyagin.net dns]#
[root@server.ankomyagin.net dns]# cp -R /var/named/* /vagrant/provision/server/dns/var/named/
cp: overwrite '/vagrant/provision/server/dns/var/named/data/named.run'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind.jnl'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/ankomyagin.net'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/fz/ankomyagin.net'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/192.168.1'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.ca'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.empty'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.localhost'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/named.loopback'? y
[root@server.ankomyagin.net dns]# cp -R /etc/named/* /vagrant/provision/server/dns/etc/named/
cp: overwrite '/vagrant/provision/server/dns/etc/named/ankomyagin.net'? y
```

Рис. 2.33: Замена файлов

Далее создаем файл и добавляем туда скрипт(рис. 2.34 и рис. 2.35).

```
GNU nano 8.1
#!/bin/bash
echo "Provisioning script $0"
echo "Install needed packages"
dnf -y install kea
echo "Copy configuration files"
cp -R /vagrant/provision/server/dhcp/etc/kea/* /etc/kea
echo "Fix permissions"
chown -R kea:kea /etc/kea
chmod 640 /etc/kea/tsig-keys.json
restorecon -vR /etc
restorecon -vR /var/lib/kea
echo "Configure firewall"
firewall-cmd --add-service dhcp
firewall-cmd --add-service dhcp --permanent
echo "Start dhcpd service"
systemctl --system daemon-reload
```

Рис. 2.34: Создание файла

```
[root@server.ankomyagin.net server]# touch dhcp.sh
[root@server.ankomyagin.net server]# chmod +x dhcp.sh
[root@server.ankomyagin.net server]# nano dhcp.sh
```

Рис. 2.35: Файл dhcp.sh

Завершаем работу(рис. 2.36).

```
[root@server.ankomyagin.net ~]# exit
logout
[ankomyagin@server.ankomyagin.net ~]$ exit
logout
vagrant@server:~$ exit
logout
PS C:\work\ankomyagin\vagrant> |
```

Рис. 2.36: Выключение

3 Контрольные вопросы

- В каких файлах хранятся настройки сетевых подключений?

`/etc/NetworkManager/system-connections/` (управляется через NetworkManager)

- За что отвечает протокол DHCP?

DHCP (Dynamic Host Configuration Protocol) отвечает за автоматическую выдачу клиентам сетевых настроек: IP-адреса, маски подсети, шлюза по умолчанию и адресов DNS-серверов.

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

Он выделяет каждому компьютеру произвольный свободный IP-адрес из определённого администратором диапазона

- В каких файлах обычно находятся настройки DHCP-сервера? За что отвечает каждый из файлов?

`/etc/dhcp/dhcpd.conf`- содержит все настройки — объявление подсетей, пулы адресов, шлюзы, DNS-серверы, время аренды и т.д.

`/var/lib/dhcp/dhcpd.leases` - автоматически ведётся демоном `dhcpd`, хранит историю выданных адресов, кому и на какой срок.

- Что такое DDNS? Для чего применяется DDNS?

Это технология, позволяющая автоматически обновлять записи на DNS-сервере в реальном времени

- Какую информацию можно получить, используя утилиту `ifconfig`?
Приведите примеры с использованием различных опций.

Показывает конфигурацию сетевых интерфейсов (IP-адрес, маску, MAC-адрес), статистику по приему/передаче данных (RX/TX).

`ifconfig` – показать все активные интерфейсы.

`ifconfig eth0` – показать информацию только для интерфейса `eth0`.

`ifconfig eth0 up` – включить (up) интерфейс `eth0`.

- Какую информацию можно получить, используя утилиту `ping`?

Проверяет доступность узла в сети и качество соединения (время отклика, потерю пакетов).

4 Выводы

В ходе работы были изучены принципы работы DHCP и приобретены навыки по установке и конфигурированию DHCP-сервера.

Список литературы

[ТУИС] (https://esystem.rudn.ru/pluginfile.php/2854738/mod_resource/content/8/003-dhcp.pdf)