

# **Отчёт о лабораторной работе**

## **Лабораторная работа 3**

Андрюшин Никита Сергеевич

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Выполнение лабораторной работы</b>	<b>6</b>
<b>3 Выводы</b>	<b>19</b>

# Список иллюстраций

2.1	Запуск сервера . . . . .	6
2.2	Установка kea . . . . .	6
2.3	Бекап конфига . . . . .	7
2.4	Редактирование конфига . . . . .	7
2.5	Настройка подсети . . . . .	8
2.6	Установка интерфейса . . . . .	8
2.7	Загрузка конфига . . . . .	8
2.8	Перезагрузка дæмонов . . . . .	9
2.9	Редактирование fz . . . . .	9
2.10	rz . . . . .	9
2.11	Пинг dhcp . . . . .	9
2.12	firewall и selinux . . . . .	10
2.13	Логи сервера . . . . .	10
2.14	Запуск dhcp . . . . .	10
2.15	Сверка по логу . . . . .	10
2.16	Скрипт для клиента . . . . .	11
2.17	Vagrantfile . . . . .	11
2.18	Запуск клиента . . . . .	11
2.19	ifconfig . . . . .	12
2.20	Таблица с назначениями . . . . .	12
2.21	Ключ sha512 . . . . .	12
2.22	Добавление ключа . . . . .	13
2.23	Обновление файла . . . . .	13
2.24	Применение изменений . . . . .	14
2.25	Порядок монтирования . . . . .	14
2.26	Смена прав файла . . . . .	14
2.27	Изменение конфигурации . . . . .	14
2.28	Перезапуск ddns . . . . .	15
2.29	Добавление информации о ddns . . . . .	15
2.30	Перезапуск службы с применением изменений . . . . .	16
2.31	Обновление данных . . . . .	16
2.32	dig . . . . .	16
2.33	Перенос конфигурации . . . . .	17
2.34	Скрипт vagrant . . . . .	17
2.35	Vagtantfile . . . . .	18

# **Список таблиц**

# **1 Цель работы**

Приобретение практических навыков по установке и конфигурированию DHCP-сервера.

## 2 Выполнение лабораторной работы

Для начала запустим сервер через vagrant (рис. 2.1).

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

Рис. 2.1: Запуск сервера

Зайдём под рутом и установим пакет для настройки dhcp - kea (рис. 2.2).

```
[nsandryushin@server.nsandryushin.net ~]$ sudo -i
[sudo] password for nsandryushin:
[root@server.nsandryushin.net ~]# dnf -y install kea
Extra Packages for Enterprise Linux 10 - x86_64
Extra Packages for Enterprise Linux 10 - x86_64
Rocky Linux 10 - BaseOS
Rocky Linux 10 - BaseOS
Rocky Linux 10 - AppStream
Rocky Linux 10 - AppStream
Rocky Linux 10 - CRB
Rocky Linux 10 - CRB
Rocky Linux 10 - Extras
Rocky Linux 10 - Extras
Dependencies resolved.

=====
                         Package          Architecture      Version
=====
Installing:
  kea                           x86_64           2.6.3-1.el10_0
Installing dependencies:
  kea-libs                       x86_64           2.6.3-1.el10_0
  libpq                          x86_64           16.8-2.el10_0
  log4cplus                      x86_64           2.1.1-8.el10
  mariadb-connector-c            x86_64           3.4.4-1.el10
  mariadb-connector-c-config     noarch           3.4.4-1.el10

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
(2/6): kea-libs-2.6.3-1.el10_0.x86_64.rpm
(3/6): mariadb-connector-c-3.4.4-1.el10.x86_64.rpm
(4/6): mariadb-connector-c-config-3.4.4-1.el10.noarch.rpm
(5/6): log4cplus-2.1.1-8.el10.x86_64.rpm
(6/6): kea-2.6.3-1.el10_0.x86_64.rpm
```

Рис. 2.2: Установка kea

Перед изменением конфигурационного файла, на всякий случай сделаем бекап и отредактируем его (рис. 2.3).

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

Рис. 2.3: Бекап конфига

Мы поменяем изначальные данные на свои - изменим доменное имя на собственное, а также поставим ip на ip нашей машины - 192.168.1.1 (рис. 2.4).

```
GNU nano 8.1                                     /etc/kea/kea-dhcp4.conf
// domain-name-servers you could do this:
// {
//   "name": "domain-name-servers",
//   "code": 6,
//   "csv-format": "true",
//   "space": "dhcp4",
//   "data": "192.0.2.1, 192.0.2.2"
// }
// but it's a lot of writing, so it's easier to do this instead:
{
  "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": "nsandryushin.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": "nsandryushin.net"
},
```

Рис. 2.4: Редактирование конфига

Спустимся ниже и настроим свою подсеть следующим образом (рис. 2.5).

```

"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. 2.5: Настройка подсети

И установим интерфейс для dhcp как eth1 (рис. 2.6).

```

"DHcp4": {
    // Add names of your network interfaces to listen on.
    "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 name (e.g. "eth0/192.0.2.1").
        "interfaces": [ "eth1" ]

        // Kea DHCIPv4 server by default listens using raw sockets. This ensures
        // all packets, including those sent by directly connected clients
        // that don't have IPv4 address yet, are received. However, if your
        // traffic is always relayed, it is often better to use regular
        // UDP sockets. If you want to do that, uncomment this line:
        // "dhcp-socket-type": "udp"
    },
}

```

Рис. 2.6: Установка интерфейса

Загрузим конфиг и убедимся, что нигде нет критических ошибок (рис. 2.7).

```

[root@server manufyashin]# ./kea-dhcp4 -I /etc/kea/kea-dhcp4.conf
2025-09-20 12:08:22.895 INFO [kea-dhcp4 hosts/19080.139798808010624] HOSTS_BACKENDS_REGISTERED the following host backend types are available: mysql postgresql
2025-09-20 12:08:22.897 WARN [kea-dhcp4 dhcpsrv/19080.139798808010624] DHCPPSRV_MT_DISABLED_QUEUE_CONTROL disabling dhcp queue control when multi-threading is enabled.
2025-09-20 12:08:22.897 WARN [kea-dhcp4 dhcpsrv/19080.139798808010624] DHCP4_RESERVATIONS_LOOKUP_FIRST_ENABLED Multi-threading is enabled and host reservations lookup is always performed first.
2025-09-20 12:08:22.897 INFO [kea-dhcp4.dhcpsrv/19080.139798808010624] DHOPSRV_CFGMGR_NEW_SUBNET4 a new subnet has been added to configuration: 192.168.1.0/24 with params: t1=900,
t2=18000, valid-lifetime=3600
2025-09-20 12:08:22.898 INFO [kea-dhcp4.dhcpsrv/19080.139798808010624] DHOPSRV_CFGMGR_SOCKET_TYPE_SELECT using socket type raw
2025-09-20 12:08:22.898 INFO [kea-dhcp4.dhcpsrv/19080.139798808010624] DHOPSRV_CFGMGR_ADD_INTERFACE listening on interface eth1
2025-09-20 12:08:22.898 INFO [kea-dhcp4.dhcpsrv/19080.139798808010624] DHOPSRV_CFGMGR_SOCKET_TYPE_DEFAULT 'dhcp-socket-type' not specified , using default socket type raw

```

Рис. 2.7: Загрузка конфига

Перезагрузим системные дæмоны (рис. 2.8).

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

Рис. 2.8: Перезагрузка даемонов

И слегка отредактируем наш файл с прошлой лабораторной работы в папке fz, добавив запись о dhcp (рис. 2.9).

```
GNU nano 8.1
/var/named/master/fz/nsandryushin.net.

$TTL 1D
@ IN SOA @ server.nsandryushin.net. (
    2025092001; serial
        1D ; refresh
        1H ; retry
        1W ; expire
        3H ) ; minimum
    NS   @
    A   192.168.1.1
    AAAA ::1
$ORIGIN nsandryushin.net.
server A 192.168.1.1
ns A 192.168.1.1
dhcp A 192.168.1.1
```

Рис. 2.9: Редактирование fz

То же самое сделаем с rz (рис. 2.10).

```
GNU nano 8.1
/var/named/master/rz/192.168.1.

$TTL 1D
@ IN SOA @ server.nsandryushin.net. (
    2025092001; serial
        1D ; refresh
        1H ; retry
        1W ; expire
        3H ) ; minimum
    NS   @
    A   192.168.1.1
    AAAA ::1
    PTR  server.nsandryushin.net.
$ORIGIN 1.168.192.in-addr.arpa.
1 PTR server.nsandryushin.net.
1 PTR ns.nsandryushin.net.
1 PTR dhcp.nsandryushin.net.
```

Рис. 2.10: rz

Перезагрузим сервер ДНС и убедимся, что мы можем пингануть dhcp сервер (рис. 2.11).

```
[root@server.nsandryushin.net ~]# systemctl restart named
[root@server.nsandryushin.net ~]# ping dhcp.nsandryushin.net
PING dhcp.nsandryushin.net (192.168.1.1) 56(84) bytes of data.
64 bytes from dhcp.nsandryushin.net (192.168.1.1): icmp_seq=1 ttl=64 time=0.113 ms
64 bytes from dhcp.nsandryushin.net (192.168.1.1): icmp_seq=2 ttl=64 time=0.084 ms
64 bytes from dhcp.nsandryushin.net (192.168.1.1): icmp_seq=3 ttl=64 time=0.320 ms
```

Рис. 2.11: Пинг dhcp

Теперь настроим firewall и обновим метки selinux (рис. 2.12).

```
[root@server.nsandryushin.net ~]# firewall-cmd --list-services
cockpit dhcpcv6-client dns ssh
[root@server.nsandryushin.net ~]# firewall-cmd --get-services
0:AD RH-Satellite-6 capsule afp alvr amanda-client k5-client ampg amps anno-1602 anno-1800 cupcsd aseqnet audit ausweiler bareos-filedaemon bareso-storage bb bgp bitcoin bitcoin-testnet_bitcointestnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mor civilization-v cockpit collectd condor-collector cratedb ctdb dds multicast dds-unicast dhcp dhcpcv6-client distcc dns dns-over-quic dropbox-lansync elastisearch etcd-client etcd-server factorio finger foreman foreman-proxy freeipa-4 freeipa-ldap freeipa-ldaps freeipa-nigella-client ganglia-master git gpm grafana gre high-availability http http3 https ident imap imaps iperf2 iperf3 lpfpp ipp ipp-client ipsec in kdeconnect kerberos kibana klogin kpasswd kprop kshell kube-api kube-apiserver kube-control-plane kube-control-plane-secure kube-controller-kube-nodeport-service kube-scheduler kube-worker kubelet kubelet-readyonly kubelet-worker ldap ldaps libvirt libvirt-tls lmnr-tcp llmnr-udp managehive matrix mdns memcache minecraft minidlna mndp mongod mosquitto mqtt mqtt-tls ms-wbt mssql murmur mysql nt bios-ns netdata-dashboard nfs nfs3 ntp ntp-openelementry openvswitch ovirt-imageio ovirt-storageconsole ovirt-vmconsole plex pncd p stgresql privoxy prometheus prometheus-node-exporter proxy-dhcp ps2link ps3netsrv puppetmaster quassel radius radsec rdp redis sh rsyncd rtspl salt-master samba samba-client samba-dc sane settlers-history-collection sip sipx slmdevr slp smtp smtp-submission smtsp snmp s lansync spotify-sync squid ssh sshfs steam steam-lan-transfer steam-streaming stellaris stronghold-crusader stun stuns submission supertuxkarting-relay synergy syscoman syslog syslog-tls telnet tentacle terraria tftp tftp-title38 tinc tor-socks transmission-client turn turns upnp-cliper httpd httpd-https wireguard ws-discovery ws-discovery-client ws-discovery-host ws-discovery-tcp ws-discovery-udp wsdd wsdd-http wsmans xdr p-server zabbix-agent zabbix-java-gateway zabbix-server zabbix-trapper zabbix-web-service zero-k zerotier
[root@server.nsandryushin.net ~]# firewall-cmd --add-service=dhcp
success
[root@server.nsandryushin.net ~]# firewall-cmd --add-service=dhcp --permanent
success
[root@server.nsandryushin.net ~]# 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_t:s0
[root@server.nsandryushin.net ~]# restorecon -vR /var/named
[root@server.nsandryushin.net ~]# restorecon -vR /var/lib/kea/
[root@server.nsandryushin.net ~]#
```

Рис. 2.12: firewall и selinux

Убедимся по логам, что сервер ДНС работает и не выдаёт ошибок (рис. 2.13).

```
[nsandryushin@server.nsandryushin.net ~]$ sudo tail -f /var/log/messages
[sudo] password for nsandryushin:
Sep 20 12:01:29 server named[10320]: zone localhost.localdomain/IN: loaded serial 0
Sep 20 12:01:29 server named[10320]: all zones loaded
Sep 20 12:01:29 server named[10320]: running
Sep 20 12:01:29 server named[1]: Started named.service - Berkeley Internet Name Domain (DNS).
Sep 20 12:01:39 server named[10320]: resolver priming query complete: timed out
Sep 20 12:01:39 server named[10320]: managed-keys-zone: Unable to fetch DNSKEY set '.'; timed out
Sep 20 12:03:02 server systemd[1]: Starting fwupd-refresh.service - Refresh fwupd metadata and update motd...
Sep 20 12:03:02 server systemd[1]: fwupd-refresh.service: Deactivated successfully.
Sep 20 12:03:02 server systemd[1]: Finished fwupd-refresh.service - Refresh fwupd metadata and update motd.
Sep 20 12:03:17 server systemd[5566]: Started run-pi0384-110684.scope - [systemd-run] /usr/bin/bash.
Sep 20 12:03:37 server systemd-logind[918]: Existing logind session ID 5 used by new audit session, ignoring.
Sep 20 12:03:37 server systemd-logind[918]: New session c3 of user root.           I
Sep 20 12:03:37 server systemd[1]: Started session c3.scope - Session c3 of User root.
Sep 20 12:03:47 server systemd[1]: Started kea-dhcp4.service - Kea DHCPv4 Server.
Sep 20 12:03:47 server kea-dhcp4[10433]: 2025-09-20 12:03:47.896 INFO [kea-dhcp4.dhcp4/10433.139955159222464] DHCP4_STARTING Kea
Sep 20 12:03:47 server kea-dhcp4[10433]: 2025-09-20 12:03:47.903 INFO [kea-dhcp4.commands/10433.139955159222464] COMMAND_RECEIVE
Sep 20 12:03:47 server kea-dhcp4[10433]: 2025-09-20 12:03:47.914 ERROR [kea-dhcp4.dhcp4/10433.139955159222464] DHCP4_INIT_FAIL failing file '/etc/kea/kea-dhcp4.conf': specified reservation '192.0.2.201' is not within the IPv4 subnet '192.168.1.0/24'
Sep 20 12:03:47 server systemd[1]: kea-dhcp4.service: Main process exited, code=exited, status=1/FAILURE
Sep 20 12:03:47 server systemd[1]: kea-dhcp4.service: Failed with result 'exit-code'.

```

Рис. 2.13: Логи сервера

Теперь запускаем dhcp сервер (рис. 2.14).

```
[root@server.nsandryushin.net ~]# systemctl start kea-dhcp4.service
[root@server.nsandryushin.net ~]#
```

Рис. 2.14: Запуск dhcp

Посмотрим на лог и убедимся, что запуск был успешен (рис. 2.15).

```
[nsandryushin@server.nsandryushin.net ~]$ sudo tail -f /var/log/messages
[sudo] password for nsandryushin:
Sep 20 12:03:47 server systemd[1]: kea-dhcp4.service: Main process exited, code=exited, status=1/FAILURE
Sep 20 12:03:47 server systemd[1]: kea-dhcp4.service: Failed with result 'exit-code'.
Sep 20 12:04:05 server systemd[1]: session-c3.scope: Deactivated successfully.
Sep 20 12:04:05 server systemd-logind[918]: Session c3 logged out. Waiting for processes to exit.
Sep 20 12:04:05 server systemd-logind[918]: Removed session c3.
Sep 20 12:05:05 server named[10320]: shut down hung fetch while resolving 'contile.services.mozilla.com/A'
Sep 20 12:05:05 server named[10320]: shut down hung fetch while resolving 'contile.services.mozilla.com/AAA'
Sep 20 12:08:39 server systemd[1]: Started kea-dhcp4.service - Kea DHCPv4 Server.
Sep 20 12:08:39 server kea-dhcp4[10516]: 2025-09-20 12:08:40.096 INFO [kea-dhcp4.dhcp4/10516.139974460152000] DHCP4_STARTING Kea DHCPv4 server
Sep 20 12:08:40 server kea-dhcp4[10516]: 2025-09-20 12:08:40.099 INFO [kea-dhcp4.commands/10516.139974460152000] COMMAND_RECEIVED Received com
Sep 20 12:08:51 server systemd-logind[918]: Existing logind session ID 5 used by new audit session, ignoring.
Sep 20 12:08:51 server systemd-logind[918]: New session c4 of user root.
Sep 20 12:08:51 server systemd[1]: Started session c4.scope - Session c4 of User root.
```

Рис. 2.15: Сверка по логу

Далее убедимся в том, что в нашей папке клиента в vagrant представлен скрипт следующего содержания для настройки сети, берущей свой ip по dhcp (рис. 2.16).



```
#!/bin/bash
echo "Provisioning script $@"
nmcli connection modify "System eth0" ipv4.gateway "192.168.1.1"
nmcli connection up "System eth0"
nmcli connection modify eth0 ipv4.never-default true
nmcli connection down eth0
nmcli connection up eth0
# systemctl restart NetworkManager
```

Рис. 2.16: Скрипт для клиента

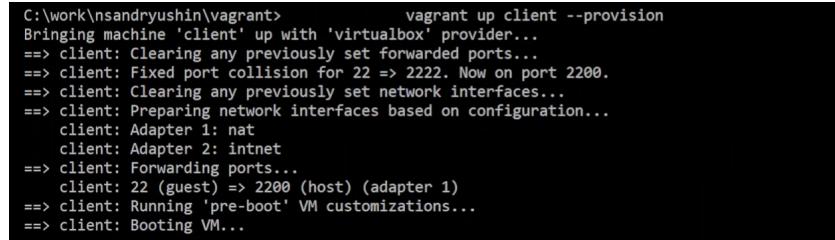
В Vagrantfile мы убедимся, что этот скрипт прописан для запуска (рис. 2.17).

```
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. 2.17: Vagrantfile

Когда приготовления завершены, мы можем запустить клиент (рис. 2.18).



```
C:\work\nsandryushin\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...
==> client: Booting VM...
```

Рис. 2.18: Запуск клиента

Зайдя в клиент, через ifconfig убедимся, что айпи был получен с сервера. Это так, айпи назначился как 192.168.1.30 (рис. 2.19).

```
[nsandryushin@client.nsandryushin.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 fd00::a00:27ff:fe5d:6d8e prefixlen 64 scopeid 0x0<global>
        inet6 fe80::a00:27ff:fe5d:6d8e prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:5d:6d:8e txqueuelen 1000 (Ethernet)
            RX packets 1654 bytes 195424 (190.8 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 1473 bytes 232829 (227.3 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::6d76:bb48:cd95:17b7 prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:8e:21:48 txqueuelen 1000 (Ethernet)
            RX packets 51 bytes 4918 (4.8 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 264 bytes 25460 (24.8 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)
```

Рис. 2.19: ifconfig

Информация о назначении айпи также хранится в файле /var/lib/kea/kea-leases4.csv на сервере (рис. 2.20).

```
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
192.168.1.30,08:00:27:8e:21:48,01:08:00:27:8e:21:48,3600,1758375351,1,0,0,client,,
```

Рис. 2.20: Таблица с назначениями

Теперь создадим ключ sha512 и убедимся в том, что он создался (рис. 2.21).

```
[root@server.nsandryushin.net ~]# mkdir -p /etc/named/keys
[root@server.nsandryushin.net ~]# tsig-keygen -a HMAC-SHA512 DHCP_UPDATER > /etc/named/keys/dhcp_updater.key
[root@server.nsandryushin.net ~]# cat /etc/named/keys/dhcp_updater.key
key "DHCP_UPDATER" {
    algorithm hmac-sha512;
    secret "QT/72lqh6PhaMuPkclNhNTaFYK8hTmQUgikvuMKj4Mb/grr1Jc1wIcJzJmyxb59Pu7SZU5ssHg6kj/c5lCWyQ==";
}
```

Рис. 2.21: Ключ sha512

Этот ключ добавим в /etc/named.conf (рис. 2.22).

```

GNU nano 8.1                               /etc/named.conf
    - If your recursive DNS server has a public IP address, you MUST enable access
      control to limit queries to your legitimate users. Failing to do so will
      cause your server to become part of large scale DNS amplification
      attacks. Implementing BCP38 within your network would greatly
      reduce such attack surface
*/
recursion yes;

dnssec-validation yes;

managed-keys-directory "/var/named/dynamic";
geolite-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/nsandryushin.net";
include "/etc/named/keys/dhcp_updater.key";

```

Рис. 2.22: Добавление ключа

Обновим файл /etc/named/nsandryushin.net, добавив туда dhcp (рис. 2.23).

```

GNU nano 8.1                               /etc/named/nsandryushin.net
// named.rfc1912.zones:
//
// Provided by Red Hat caching-nameserver package
//
// ISC BIND named zone configuration for zones recommended by
// RFC 1912 section 4.1 : localhost TLDs and address zones
// and https://tools.ietf.org/html/rfc6303
// (c)2007 R W Franks
//
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//
// Note: empty-zones-enable yes; option is default.
// If private ranges should be forwarded, add
// disable-empty-zone "."; into options
//

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

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 DHCID;
    };
};

```

Рис. 2.23: Обновление файла

Проверим корректность конфига на синтаксис и перезапустим DNS службу, а также создадим файл с ключом (рис. 2.24).

```
[root@server.nsandryushin.net ~]# named-checkconf
[root@server.nsandryushin.net ~]# systemctl restart named
[root@server.nsandryushin.net ~]# touch /etc/kea/tsig-keys.json
[root@server.nsandryushin.net ~]#
```

Рис. 2.24: Применение изменений

В созданный файл вставим ключ, который мы сгенерировали ранее (рис. 2.25).

```
GNU nano 8.1
/etc/kea/tsig-keys.json
{
  "tsig-keys": [
    {
      "name": "DHCP_UPDATER",
      "algorithm": "hmac-sha512",
      "secret": "QT72lh6PhaMuPkclkNhnTaFYK8hTaQUgikvuMKj4Mb/grr1Jc1wIcJzJmyxb59Pu7SZU5ssHg6kj/c5lCWyQ=="
    }
  ],
}
```

Рис. 2.25: Порядок монтирования

Поменяем права и владельца созданного файла, предоставив его системному пользователю службы (рис. 2.26).

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

Рис. 2.26: Смена прав файла

Теперь заполним файл конфигурации ddns, который перепишем с нуля согласно данному шаблону, поменяв имя на свой домен (рис. 2.27).

```
GNU nano 8.1
/etc/kea/kea-dhcp-ddns.conf
{
  "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": "nsandryushin.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"
        }
      ]
    }
  ]
}
```

Рис. 2.27: Изменение конфигурации

Предоставим этот файл во владение системному пользователю, а так же загрузим эту конфигурацию, и убедимся, что она загружена успешно. После этого перезапустим ddns службу (рис. 2.28).

```
[root@server.nsandryushin.net ~]# chown kea:kea /etc/kea/kea-dhcp-ddns.conf
[root@server.nsandryushin.net ~]# kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
2025-09-20 12:51:15.786 INFO [kea-dhcp-ddns.dct1/11282.139911991370048] DCT1_CONFIG_CHECK_COMPLETE server has completed configuration changing UDP, result: success(0), text:Configuration check successful
[root@server.nsandryushin.net ~]# systemctl enable --now kea-dhcp-ddns.service
Created symlink '/etc/systemd/system/multi-user.target.wants/kea-dhcp-ddns.service' → '/usr/lib/systemd/system/kea-dhcp-ddns.service'.
[root@server.nsandryushin.net ~]# 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; preset: disabled)
     Active: active (running) since Sat 2025-09-20 12:51:26 UTC; 5s ago
    Invocation: 86f5c278084943a7e1e90eb6015587ec
      Docs: man:kea-dhcp-ddns(8)
      Main PID: 11423 (kea-dhcp-ddns)
        Tasks: 5 (limit: 10407)
       Memory: 1.8M (peak: 5.9M)
         CPU: 87ms
        CGroup: /system.slice/kea-dhcp-ddns.service
                   └─11423 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Sep 20 12:51:26 server.nsandryushin.net systemd[1]: Started kea-dhcp-ddns.service - Kea DHCP-DDNS Server.
Sep 20 12:51:27 server.nsandryushin.net kea-dhcp-ddns[11423]: 2025-09-20 12:51:27.022 INFO [kea-dhcp-ddns.dct1/11423.13990289772992] DC
Sep 20 12:51:27 server.nsandryushin.net kea-dhcp-ddns[11423]: INFO COMMAND_ACCEPTOR_START Starting to accept connections via unix domain
Sep 20 12:51:27 server.nsandryushin.net kea-dhcp-ddns[11423]: INFO DCT1_CONFIG_COMPLETE server has completed configuration: listening on
Sep 20 12:51:27 server.nsandryushin.net kea-dhcp-ddns[11423]: INFO DHCP_DDNS_STARTED Kea DHCP-DDNS server version 2.6.3 started
```

Рис. 2.28: Перезапуск ddns

Теперь добавим информацию о ddns в наш файл с конфигурацией dhcp (рис. 2.29).

```
GNU nano 8.1                                     /etc/kea/kea-dhcp4.conf
// seconds reclaimed leases, which have expired more than 3600
// seconds ago, will be removed. The limits for leases reclamation
// are 100 leases or 250 ms for a single cycle. A warning message
// will be logged if there are still expired leases in the
// database after 5 consecutive reclamation cycles.
// If both "flush-reclaimed-timer-wait-time" and "hold-reclaimed-time"
// are not 0, when the client sends a release message the lease is expired
// instead of being deleted from the lease storage.
"expired-leases-processing": {
    "reclaim-timer-wait-time": 10,
    "flush-reclaimed-timer-wait-time": 25,
    "hold-reclaimed-time": 3600,
    "max-reclaim-leases": 100,
    "max-reclaim-time": 250,
    "unwarned-reclaim-cycles": 5
},
// Global timers specified here apply to all subnets, unless there are
// subnet specific values defined in particular subnets.
"renew-timer": 9000,
"rebind-timer": 18000,
"valid-lifetime": 36000,
"dhcp-ddns": {
    "enable-updates": true
},
"ddns-qualifying-suffix": "nsandryushin.net",
"ddns-override-client-update": true,
// Many additional parameters can be specified here:
// - option definitions (if you want to define vendor options, your own
//   custom options or perhaps handle standard options
//   that Kea does not support out of the box yet)
// - client classes
// - hooks
// - ddns information (how the DHCPv4 component can reach a DDNS daemon)
//
```

Рис. 2.29: Добавление информации о ddns

Вновь загрузим конфигурацию и перезапустим службу dhcp (рис. 2.30).

```
[root@server.nsandryushin.net ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-09-20 13:04:10.447 INFO [kea-dhcp4.hosts/11662.139791145076928] HOSTS_BACKENDS_REGISTERED the following host backend types are available
2025-09-20 13:04:10.452 WARN [kea-dhcp4.dhcpsrv/11662.139791145076928] DHCPSRV_MT_DISABLED.QUEUE_CONTROL disabling dhcp queue control when
2025-09-20 13:04:10.452 WARN [kea-dhcp4.dhcp4/11662.139791145076928] DHCP4_RESERVATIONS_LOOKUP_FIRST_ENABLED Multi-threading is enabled in
2025-09-20 13:04:10.453 INFO [kea-dhcp4.dhcpsrv/11662.139791145076928] DHCPSRV_CFMGR_NEW_SUBNET4 a new subnet has been added to configuration
t2=1800, valid-lifetime=3600
2025-09-20 13:04:10.453 INFO [kea-dhcp4.dhcpsrv/11662.139791145076928] DHCPSRV_CFMGR_SOCKET_TYPE_SELECT using socket type raw
2025-09-20 13:04:10.454 INFO [kea-dhcp4.dhcpsrv/11662.139791145076928] DHCPSRV_CFMGR_ADD_IFACE listening on interface eth1
2025-09-20 13:04:10.455 INFO [kea-dhcp4.dhcpsrv/11662.139791145076928] DHCPSRV_CFMGR_SOCKET_TYPE_DEFAULT "dhcp-socket-type" not specified
[root@server.nsandryushin.net ~]# systemctl restart 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 13:04:22 UTC; 6s ago
       Docs: man:kea-dhcp4(8)
             Main PID: 11677 (kea-dhcp4)
               Tasks: 7 (limit: 10407)
             Memory: 3.5M (peak: 7.2M)
                CPU: 124ms
              CGroup: /system.slice/kea-dhcp4.service
                  └─11677 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf

Sep 20 13:04:22 server.nsandryushin.net systemd[1]: Started kea-dhcp4.service - Kea DHCPv4 Server.
Sep 20 13:04:22 server.nsandryushin.net kea-dhcp4[11677]: 2025-09-20 13:04:22.941 INFO [kea-dhcp4.dhcp4/11677.139782083418304] DHCP4_STARTED
```

Рис. 2.30: Перезапуск службы с применением изменений

Теперь на клиенте перезапустим интернет, чтобы обновить данные (рис. 2.31).

```
[nsandryushin@client.nsandryushin.net ~]$ nmcli connection down eth1
Connection 'eth1' successfully deactivated (D-Bus active path: /org/freedesktop/
NetworkManager/ActiveConnection/4)
[nsandryushin@client.nsandryushin.net ~]$ nmcli connection up eth1
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkMa
nager/ActiveConnection/6)
[nsandryushin@client.nsandryushin.net ~]$
```

Рис. 2.31: Обновление данных

Теперь через dig получим информацию о нашем сервере (рис. 2.32).

```
[nsandryushin@client.nsandryushin.net ~]$ dig @192.168.1.1 client.nsandryushin.n
et
; <>> DiG 9.18.33 <>> @192.168.1.1 client.nsandryushin.net
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NXDOMAIN, id: 51786
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1232
;; COOKIE: d25edd1c8c4fa8870100000068cea78418f68153507351ac (good)
;; QUESTION SECTION:
;client.nsandryushin.net.      IN      A

;; AUTHORITY SECTION:
nsandryushin.net.      10800   IN      SOA      nsandryushin.net. server.nsandry
ushin.net. 2025092001 86400 3600 604800 10800
;; Query time: 3 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sat Sep 20 13:09:24 UTC 2025
;; MSG SIZE  sent: 139
```

Рис. 2.32: dig

Теперь переместим данные созданных ранее конфигураций в вагрант, после

чего создадим скрипт (рис. 2.33).

```
[root@server.nsandryushin.net ~]# cd /vagrant/provision/server/
[root@server.nsandryushin.net server]# mkdir -p /vagrant/provision/server/dhcp/etc/kea
[root@server.nsandryushin.net server]# cp -R /etc/kea/* /vagrant/provision/server/dhcp/etc/kea/
[root@server.nsandryushin.net server]# cd /vagrant/provision/server/dns/           I
[root@server.nsandryushin.net dns]# cp -R /var/named/* /vagrant/provision/server/dns/var/named/
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/nsandryushin.net'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/192.168.1'? y
[root@server.nsandryushin.net dns]# cp -R /etc/named/* /vagrant/provision/server/dns/etc/named/
cp: overwrite '/vagrant/provision/server/dns/etc/named/nsandryushin.net'? y
[root@server.nsandryushin.net dns]# cd /vagrant/provision/server
[root@server.nsandryushin.net server]# touch dhcp.sh
[root@server.nsandryushin.net server]# chmod +x dhcp.sh
[root@server.nsandryushin.net server]#
```

Рис. 2.33: Перенос конфигурации

В скрипте напишем алгоритм настройки dhcp (рис. 2.34).

```
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 dhcpcd service"
systemctl --system daemon-reload
systemctl enable --now kea-dhcp4.service
systemctl enable --now kea-dhcp-ddns.service
```

Рис. 2.34: Скрипт vagrant

И добавим запуск скрипта в Vagrantfile (рис. 2.35).

```
59 server.vm.hostname = 'server'
60
61 server.vm.boot_timeout = 1440
62
63 server.ssh.insert_key = false
64 server.ssh.username = 'vagrant'
65 server.ssh.password = 'vagrant'
66
67 server.vm.network :private_network,
68   ip: "192.168.1.1",
69   virtualbox_intnet: true
70
71 server.vm.provider :virtualbox do |virtualbox|
72   virtualbox.customize ["modifyvm", :id, "--vrde", "on"]
73   virtualbox.customize ["modifyvm", :id, "--vrdeport", "3391"]
74 end
75
76 server.vm.provision "server dummy",
77   type: "shell",
78   preserve_order: true,
79   path: "provision/server/01-dummy.sh"
80
81 server.vm.provision "server dns",
82   type: "shell",
83   preserve_order: true,
84   path: "provision/server/dns.sh"
85 server.vm.provision "server dhcp",
86   type: "shell",
87   preserve_order: true,
88   path: "provision/server/dhcp.sh"
89
90 end
91
```

Рис. 2.35: Vagrantfile

## **3 Выводы**

в результате выполнения работы были получены навыки настройки dhcp