

Лабораторная работа № 3.

Настройка DHCP-сервера

Садова Д. А.

Российский университет дружбы народов, Москва, Россия

Информация

- Садова Диана Алексеевна
- студент бакалавриата
- Российский университет дружбы народов
- [113229118@pfur.ru]
- <https://DianaSadova.github.io/ru/>

Вводная часть

- Понимание как устроен DHCP-сервера
- Умение его подключать

- Приобретение практических навыков по установке и конфигурированию DHCP-сервера

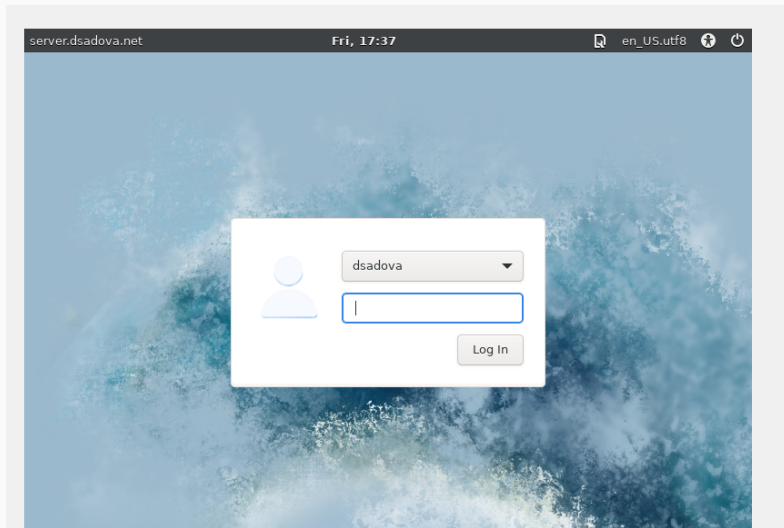
- Текст лабораторной работы № 3
- Интернет для исправления ошибок

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.

Установка DHCP-сервера

- Запустите виртуальную машину server:



- На виртуальной машине server войдите под вашим пользователем и откройте терминал. Перейдите в режим суперпользователя:

```
[dsadova@server.dsadova.net ~]$ sudo -i

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for dsadova:
[root@server.dsadova.net ~]#
```

Рис. 2: Заходим в режим суперпользователя

- Установите dhcpr:

```
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-2.6.3-1.el9.x86_64                     5/6
Running scriptlet: kea-2.6.3-1.el9.x86_64                       6/6
Installing     : kea-2.6.3-1.el9.x86_64                         6/6
Running scriptlet: kea-2.6.3-1.el9.x86_64                       6/6
Verifying      : kea-2.6.3-1.el9.x86_64                         1/6
Verifying      : kea-libs-2.6.3-1.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-2.6.3-1.el9.x86_64                kea-libs-2.6.3-1.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

Complete!
[root@server.dsadova.net ~]#
```

Рис. 3: Установливаем dhcpr

- Сохраните на всякий случай конфигурационный файл:

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

Рис. 4: Сохраняем конфигурационный файл

- Откройте файл `/etc/kea/kea-dhcrp4.conf` на редактирование. В этом файле: замените шаблон для `domain-name`

```
// "name": "domain-name" or "code": 15.  
{  
    "code": 15,  
    "data": "dsadova.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": "dsadova.net"  
},
```

Рис. 5: Заменяем шаблон

```
{  
  "name": "domain-name-servers",  
  "data": "192.168.1.1"  
},
```

Рис. 6: Заменяем шаблон

- на базе одного из приведённых в файле примеров конфигурирования подсети задайте собственную конфигурацию dhcp-сети, задав адрес подсети, диапазон адресов для распределения клиентам, адрес маршрутизатора и broadcast-адрес:

```
"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": "192.0.2.0/24",  
  
    // 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
```


- Настройте привязку dhcprd к интерфейсу eth1 виртуальной машины server:

```
"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"]  
}
```

Рис. 8: Настраиваем привязку dhcprd к интерфейсу eth1

- Проверьте правильность конфигурационного файла:

```
[root@server.dsadova.net ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-09-20 07:50:13.512 INFO [kea-dhcp4.hosts/8258.140333635827840] HOSTS
_BACKENDS_REGISTERED the following host backend types are available: mysql
postgresql
2025-09-20 07:50:13.515 WARN [kea-dhcp4.dhcp4/8258.140333635827840] DHC
PSRV_MT_DISABLED_QUEUE_CONTROL disabling dhcp queue control when multi-thr
eading is enabled.
2025-09-20 07:50:13.515 WARN [kea-dhcp4.dhcp4/8258.140333635827840] DHCP4
_RESERVATIONS_LOOKUP_FIRST_ENABLED Multi-threading is enabled and host res
ervations lookup is always performed first.
2025-09-20 07:50:13.517 INFO [kea-dhcp4.dhcp4/8258.140333635827840] DHC
PSRV_CFGMGR_NEW_SUBNET4 a new subnet has been added to configuration: 192.
168.1.0/24 with params: t1=900, t2=1800, valid-lifetime=3600
2025-09-20 07:50:13.517 INFO [kea-dhcp4.dhcp4/8258.140333635827840] DHC
PSRV_CFGMGR_SOCKET_TYPE_SELECT using socket type raw
2025-09-20 07:50:13.518 INFO [kea-dhcp4.dhcp4/8258.140333635827840] DHC
PSRV_CFGMGR_ADD_IFACE listening on interface eth1
2025-09-20 07:50:13.518 INFO [kea-dhcp4.dhcp4/8258.140333635827840] DHC
PSRV_CFGMGR_SOCKET_TYPE_DEFAULT "dhcp-socket-type" not specified , using d
efault socket type raw
[root@server.dsadova.net ~]#
```

- Перезагрузите конфигурацию dhcpd и разрешите загрузку DHCP-сервера при запуске виртуальной машины server:

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

Рис. 10: Перезагружаем конфигурацию dhcpd

- Добавьте запись для DHCP-сервера в конце файла прямой DNS-зоны /var/named/master/fz/user.net:

```
$TTL 1D
@          IN SOA  @ server.dsadova.net. (
                                                2025091900      ; serial
                                                1D              ; refresh
                                                1H              ; retry
                                                1W              ; expire
                                                3H )            ; minimum
        NS       @
        A        192.168.1.1
        AAAA     ::1
$ORIGIN dsadova.net.
server      A 192.168.1.1
ns          A 192.168.1.1
dhcp        A 192.168.1.1
```

и в конце файла обратной зоны /var/named/master/rz/192.168.1:

```
$TTL 1D
@      IN SOA  @ server.dsadova.net. (
                                2025091900      ; serial
                                1D                ; refresh
                                1H                ; retry
                                1W                ; expire
                                3H )              ; minimum

      NS      @
      A       192.168.1.1
      AAAA    ::1
      PTR     server.dsadova.net.
$ORIGIN 1.168.192.in-addr.arpa.
1       PTR   server.dsadova.net.
1       PTR   ns.dsadova.net.
1       PTR   dhcp.dsadova.net
```

Рис. 12: Добавляем запись для DHCP-сервера

- Перезапустите named:

```
[root@server.dsadova.net ~]# systemctl restart named
```

Рис. 13: Перезапускаем named

- Проверьте, что можно обратиться к DHCP-серверу по имени:

```
[root@server.dsadova.net ~]# ping dhcp.dsadova.net
PING dhcp.dsadova.net (192.168.1.1) 56(84) bytes of data.
64 bytes from ns.dsadova.net (192.168.1.1): icmp_seq=1 ttl=64 time=0.077 m
s
64 bytes from server.dsadova.net (192.168.1.1): icmp_seq=2 ttl=64 time=0.0
48 ms
64 bytes from ns.dsadova.net (192.168.1.1): icmp_seq=3 ttl=64 time=0.053 m
s
64 bytes from server.dsadova.net (192.168.1.1): icmp_seq=4 ttl=64 time=0.0
87 ms
64 bytes from server.dsadova.net (192.168.1.1): icmp_seq=5 ttl=64 time=0.0
66 ms
64 bytes from server.dsadova.net (192.168.1.1): icmp_seq=6 ttl=64 time=0.0
58 ms
```

Рис. 14: Обращаемся к DHCP-серверу

- Внесите изменения в настройки межсетевого экрана узла server, разрешив работу с DHCP:

```
[root@server.dsadova.net ~]# irewall-cmd --list-services
firewall-cmd --get-services
firewall-cmd --add-service=dhcp
firewall-cmd --add-service=dhcp --permanent
bash: irewall-cmd: command not found...
```

Рис. 15: Вносим изменения

- Восстановите контекст безопасности в SELinux:

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

Рис. 16: Восстанавливаем контекст безопасности в SELinux

- В дополнительном терминале запустите мониторинг происходящих в системе процессов в реальном времени:

```
[root@server.dsadova.net ~]# tail -f /var/log/messages
Sep 20 07:56:46 server named[8366]: no valid RRSIG resolving './NS/IN': 19
2.203.230.10#53
Sep 20 07:56:46 server named[8366]: validating ./NS: no valid signature fo
und
Sep 20 07:56:46 server named[8366]: no valid RRSIG resolving './NS/IN': 19
2.33.4.12#53
Sep 20 07:56:46 server named[8366]: validating ./NS: no valid signature fo
und
Sep 20 07:56:46 server named[8366]: no valid RRSIG resolving './NS/IN': 19
2.5.5.241#53
Sep 20 07:56:46 server named[8366]: validating ./NS: no valid signature fo
und
Sep 20 07:56:46 server named[8366]: no valid RRSIG resolving './NS/IN': 19
2.112.36.4#53
Sep 20 07:56:50 server named[8366]: resolver priming query complete
Sep 20 07:57:31 server systemd[1]: Starting PackageKit Daemon...
Sep 20 07:57:31 server systemd[1]: Started PackageKit Daemon.
```

- В основном рабочем терминале запустите DHCP-сервер:

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

Рис. 18: Запускаем DHCP-сервер

Анализ работы DHCP-сервера

1. Перед запуском виртуальной машины client в каталоге с проектом в вашей операционной системе в подкаталоге vagrant/provision/client создайте файл 01-routing.sh. Открыв его на редактирование, пропишите в нём следующий скрипт:

```
echo "Provisioning script $0"
nmcli connection modify "eth1" ipv4.gateway "192.168.1.1"
nmcli connection up "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. 19: Переписываем 01-routing.sh

2. В Vagrantfile подключите этот скрипт в разделе конфигурации для клиента:

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

Рис. 20: Переписываем Vagrantfile

3. Зафиксируйте внесённые изменения для внутренних настроек виртуальной машины `client` и запустите её, введя в терминале: `vagrant up client --provision`.
4. После загрузки виртуальной машины `client` вы можете увидеть на виртуальной машине `server` на терминале с мониторингом происходящих в системе процессов записи о подключении к виртуальной внутренней сети узла `client` и выдачи ему IP-адреса из соответствующего диапазона адресов. Также информацию о работе DHCP-сервера можно наблюдать в файле `/var/lib/kea/kea-leases4.csv`.

5. Войдите в систему виртуальной машины client под вашим пользователем и откройте терминал. В терминале введите

```
[dsadova@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:a8d prefixlen 64 scopeid 0x20<link>
    inet6 fd17:625c:f037:2:a00:27ff:fe69:a8d prefixlen 64 scopeid 0x0<glob
al>
    ether 08:00:27:69:0a:8d txqueuelen 1000 (Ethernet)
    RX packets 1360 bytes 160412 (156.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1200 bytes 181626 (177.3 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::a00:27ff:fe22:9ad6 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:22:9a:d6 txqueuelen 1000 (Ethernet)
    RX packets 8 bytes 1126 (1.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 307 bytes 32757 (31.9 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
```

Система имеет три сетевых интерфейса - eth0, eth1 и loopback (lo).
Интерфейс eth1 соответствует записям из DHCP-сервера.

Интерфейс eth0:

- flags=4163<UP,BROADCAST,RUNNING,MULTICAST> - интерфейс активен, поддерживает широковещание и multicast
- inet 10.0.2.15 - IPv4 адрес (вероятно NAT-интерфейс VirtualBox)
- netmask 255.255.255.0 - маска подсети класса C
- broadcast 10.0.2.255 - широковещательный адрес
- inet6 fe80::a00:27ff:fe69:a8d - link-local IPv6 адрес
- inet6 fd17:625c:f037:2:a00:27ff:fe69:a8d - глобальный IPv6 адрес (с ошибкой в записи)
- ether 08:00:27:69:0a:8d - MAC-адрес интерфейса

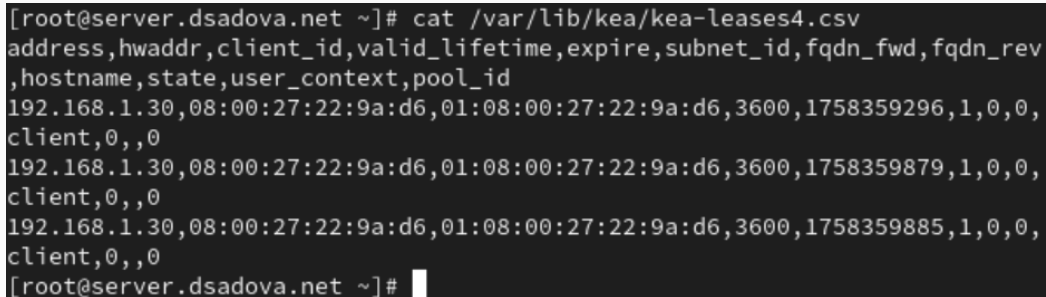
Интерфейс eth1 (основной интерес):

- flags=4163<UP,BROADCAST,RUNNING,MULTICAST> - интерфейс активен
- inet 192.168.1.30 - IPv4 адрес, совпадает с арендой из DHCP-сервера
- netmask 255.255.255.0 - маска подсети
- broadcast 192.168.1.255 - широковещательный адрес
- inet6 fe80::a00:27ff:fe22:9ad6 - link-local IPv6 адрес
- ether 08:00:27:22:9a:d6 - MAC-адрес, полностью совпадает с записью в DHCP: 08:00:27:22:9a:d6

Интерфейс lo (loopback):

- flags=73<UP,LOOPBACK,RUNNING> - loopback интерфейс активен
- inet 127.0.0.1 - стандартный loopback адрес
- netmask 255.0.0.0 - маска класса A
- inet6 ::1 - IPv6 loopback адрес

6. На машине server посмотрите список выданных адресов:

A terminal window with a dark background and light gray text. The prompt is [root@server.dsadova.net ~]#. The command cat /var/lib/kea/kea-leases4.csv has been executed. The output is a CSV file with 11 columns: address, hwaddr, client_id, valid_lifetime, expire, subnet_id, fqdn_fwd, fqdn_rev, hostname, state, user_context, pool_id. There are three data rows, all for the address 192.168.1.30. The last line shows the prompt again with a cursor.

```
[root@server.dsadova.net ~]# cat /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
192.168.1.30,08:00:27:22:9a:d6,01:08:00:27:22:9a:d6,3600,1758359296,1,0,0,
client,0,,0
192.168.1.30,08:00:27:22:9a:d6,01:08:00:27:22:9a:d6,3600,1758359879,1,0,0,
client,0,,0
192.168.1.30,08:00:27:22:9a:d6,01:08:00:27:22:9a:d6,3600,1758359885,1,0,0,
client,0,,0
[root@server.dsadova.net ~]#
```

Рис. 22: Просматриваем список выданных адресов

Файл `/var/lib/kea/kea-leases4.csv` содержит историю аренды IPv4 адресов сервером Kea DHCP.

Строка 1-2: Заголовок с полями:

- `address` - IP-адрес
- `hwaddr` - MAC-адрес клиента
- `client_id` - идентификатор клиента
- `valid_lifetime` - время жизни аренды (секунды)
- `expire` - timestamp истечения аренды
- `subnet_id` - ID подсети
- `fqdn_fwd`, `fqdn_rev` - флаги DNS-записей
- `hostname` - имя хоста

Строка 3:

192.168.1.30,08:00:27:22:9a:d6,01:08:00:27:22:9a:d6,3600,1758359296,1,0,0,,client,0,,0

Адрес: 192.168.1.30. MAC: 08:00:27:22:9a:d6 (вероятно виртуальная машина VirtualBox). Время жизни: 3600 секунд (1 час). Истекает: 1758359296 (Unix timestamp). Подсеть: ID 1. Имя хоста: client. Состояние: 0 (активная/нормальная аренда)

Строка 4:

192.168.1.30,08:00:27:22:9a:d6,01:08:00:27:22:9a:d6,3600,1758359879,1,0,0,,client,0,,0

Обновление аренды того же клиента. Новое время истечения: 1758359879.
Увеличение времени аренды

Строка 5:

192.168.1.30,08:00:27:22:9a:d6,01:08:00:27:22:9a:d6,3600,1758359885,1,0,0,,client,0,,0

1. Создадим ключ на сервере с Bind9 (на виртуальной машине server):

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

Рис. 23: Создаем ключ

2. Файл /etc/named/keys/dhcp_updater.key будет иметь следующий вид:

```
[root@server.dsadova.net ~]# cat /etc/named/keys/dhcp_updater.key
key "DHCP_UPDATER" {
    algorithm hmac-sha512;
    secret "Zq1aMBJEiVooJ5UZhvE5VPgK6a6GYhtuttMx6yNLJC+c8qOj+bc6uIjbIl
haznR8bjrT9yNCfZ0JmzuNnaK6WQ==";
};
```

Рис. 24: Просматриваем ключ

3. Поправим права доступа:

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

Рис. 25: Меняем права доступа

4. Подключим ключ в файле /etc/named.conf:

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

Рис. 26: Подключаем ключ

5. На виртуальной машине server под пользователем с правами суперпользователя отредактируйте файл /etc/named/user.net (вместо user укажите свой логин), разрешив обновление зоны:

```
zone "dsadova.net" IN {  
    type master;  
    file "master/fz/dsadova.net";  
    update-policy {  
        grant DHCP_UPDATER wildcard *.dsadova.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;  
    };  
};
```

6. Сделаем проверку конфигурационного файла:

```
[root@server.dsadova.net ~]# named-checkconf
/etc/named/dsadova.net:29: zone 'dsadova.net': already exists previous def
inition: /etc/named/dsadova.net:17
/etc/named/dsadova.net:31: writeable file 'master/fz/dsadova.net': already
in use: /etc/named/dsadova.net:19
/etc/named/dsadova.net:36: zone '1.168.192.in-addr.arpa': already exists p
revious definition: /etc/named/dsadova.net:23
/etc/named/dsadova.net:38: writeable file 'master/rz/192.168.1': already i
n use: /etc/named/dsadova.net:25
[root@server.dsadova.net ~]#
```

Рис. 28: Проверка конфигурационного файла

7. Перезапустите DNS-сервер:

```
[root@server.dsadova.net ~]# nano /etc/named/dsadova.net  
[root@server.dsadova.net ~]# systemctl restart named  
[root@server.dsadova.net ~]#
```

Рис. 29: Перезагружаем DNS-сервер

8. Сформируем ключ для Кеа. Файл ключа назовём /etc/kea/tsig-keys.json:

```
[root@server.dsadova.net ~]# touch /etc/kea/tsig-keys.json  
[root@server.dsadova.net ~]#
```

Рис. 30: Формируем ключ

9. Перенесём ключ на сервер Кеа DHCP и перепишем его в формате json:

```
[root@server.dsadova.net ~]# # Создаем файл с правильным содержимым
cat > /etc/kea/tsig-keys.json << 'EOF'
{
  "tsig-keys": [
    {
      "name": "DHCP_UPDATER",
      "algorithm": "hmac-sha512",
      "secret": "Zq1aMBJEiVooJ5UZhvE5VPgK6a6GYhtuttMx6yNLJC+c8qOj+bc
6uIjbIlhaznR8bjrT9yNCfZ0JmzuNnaK6WQ=="
    }
  ]
}
EOF
```

Рис. 31: Переносим ключ на сервер

10. Сменим владельца:

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

Рис. 32: Меняем владельца

11. Поправим права доступа:

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

Рис. 33: Меняем права доступа

12. Настройка происходит в файле /etc/kea/kea-dhcp-ddns.conf:

```
"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": "dsadova.net.",
        "key-name": "DHCP_UPDATER",
        "dns-servers": [
          { "ip-address": "192.168.1.1" }
        ]
      }
    ]
  },

  "reverse-ddns" : {
    "ddns-domains" : [
      {
```

13. Изменим владельца файла:

```
[root@server.dsadova.net ~]# chown kea:kea /etc/kea/kea-dhcp-ddns.conf  
[root@server.dsadova.net ~]#
```

Рис. 35: Меняем владельца

14. Проверим файл на наличие возможных синтаксических ошибок:

```
[root@server.dsadova.net ~]# kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
2025-09-20 08:58:43.821 INFO [kea-dhcp-ddns.dctl/9175.140649280432000] DC
TL_CONFIG_CHECK_COMPLETE server has completed configuration check: listeni
ng on 127.0.0.1, port 53001, using UDP, result: success(0), text=Configura
tion check successful
[root@server.dsadova.net ~]# nano /etc/kea/kea-dhcp-ddns.conf
[root@server.dsadova.net ~]#
```

Рис. 36: Проверяем файл

15. Запустим службу ddns:

```
[root@server.dsadova.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.dsadova.net ~]#
```

Рис. 37: Запускаем службу

16. Проверим статус работы службы:

```
[root@server.dsadova.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.dsadova.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)
   Active: active (running) since Sat 2025-09-20 09:00:22 UTC; 14s ago
     Docs: man:kea-dhcp-ddns(8)
  Main PID: 9219 (kea-dhcp-ddns)
    Tasks: 5 (limit: 4554)
   Memory: 6.2M
      CPU: 24ms
   CGroup: /system.slice/kea-dhcp-ddns.service
           └─9219 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Sep 20 09:00:22 server.dsadova.net systemd[1]: Started Kea DHCP-DDNS Serv
Sep 20 09:00:22 server.dsadova.net kea-dhcp-ddns[9219]: 2025-09-20 09:00:
Sep 20 09:00:22 server.dsadova.net kea-dhcp-ddns[9219]: INFO  COMMAND_ACC
Sep 20 09:00:22 server.dsadova.net kea-dhcp-ddns[9219]: INFO  DCTL_CONFIG
Sep 20 09:00:22 server.dsadova.net kea-dhcp-ddns[9219]: INFO  DHCP_DDNS_S
lines 1-16/16 (END)
```

17. Внесите изменения в конфигурационный файл `/etc/kea/kea-dhcp4.conf`, добавив в него разрешение на динамическое обновление DNS-записей с локального узла прямой и обратной зон:

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

Рис. 39: Вносим изменения в файл

18. Проверим файл на наличие возможных синтаксических ошибок:

```
[root@server.dsadova.net ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-09-20 09:05:12.282 INFO [kea-dhcp4.hosts/9249.140408992798848] HOSTS
_BACKENDS_REGISTERED the following host backend types are available: mysql
postgresql
2025-09-20 09:05:12.289 WARN [kea-dhcp4.dhcp4srv/9249.140408992798848] DHC
PSRV_MT_DISABLED_QUEUE_CONTROL disabling dhcp queue control when multi-thr
eading is enabled.
2025-09-20 09:05:12.289 WARN [kea-dhcp4.dhcp4/9249.140408992798848] DHCP4
_RESERVATIONS_LOOKUP_FIRST_ENABLED Multi-threading is enabled and host res
ervations lookup is always performed first.
2025-09-20 09:05:12.289 INFO [kea-dhcp4.dhcp4srv/9249.140408992798848] DHC
PSRV_CFGMGR_NEW_SUBNET4 a new subnet has been added to configuration: 192.
168.1.0/24 with params: t1=900, t2=1800, valid-lifetime=3600
2025-09-20 09:05:12.289 INFO [kea-dhcp4.dhcp4srv/9249.140408992798848] DHC
PSRV_CFGMGR_SOCKET_TYPE_SELECT using socket type raw
2025-09-20 09:05:12.289 INFO [kea-dhcp4.dhcp4srv/9249.140408992798848] DHC
PSRV_CFGMGR_ADD_IFACE listening on interface eth1
2025-09-20 09:05:12.289 INFO [kea-dhcp4.dhcp4srv/9249.140408992798848] DHC
PSRV_CFGMGR_SOCKET_TYPE_DEFAULT "dhcp-socket-type" not specified , using d
efault socket type raw
[root@server.dsadova.net ~]#
```

19. Перезапустите DHCP-сервер:

```
[root@server.dsadova.net ~]# systemctl restart kea-dhcp4.service  
[root@server.dsadova.net ~]#
```

Рис. 41: Перезапускаем сервер

20. Проверим статус:

```
[root@server.dsadova.net ~]# systemctl status kea-dhcp4.service
● kea-dhcp4.service - Kea DHCPv4 Server
   Loaded: loaded (/usr/lib/systemd/system/kea-dhcp4.service; enabled; >
   Active: active (running) since Sat 2025-09-20 09:06:20 UTC; 11s ago
     Docs: man:kea-dhcp4(8)
  Main PID: 9258 (kea-dhcp4)
    Tasks: 6 (limit: 4554)
   Memory: 6.9M
      CPU: 37ms
   CGroup: /system.slice/kea-dhcp4.service
           └─9258 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf

Sep 20 09:06:20 server.dsadova.net systemd[1]: Started Kea DHCPv4 Server.
Sep 20 09:06:20 server.dsadova.net kea-dhcp4[9258]: 2025-09-20 09:06:20.6>
Sep 20 09:06:20 server.dsadova.net kea-dhcp4[9258]: 2025-09-20 09:06:20.6>
lines 1-14/14 (END)
```

Рис. 42: Проверяем статус

21. На машине client переполучите адрес:

```
[dsadova@client ~]$ nmcli connection up eth1  
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkMa  
nager/ActiveConnection/7)  
[dsadova@client ~]$
```

Рис. 43: На client получаем адрес

22. В каталоге прямой DNS-зоны /var/named/master/fz должен появиться файл user.net.jnl, в котором в бинарном файле автоматически вносятся изменения записей зоны.

```
[root@server.dsadova.net ~]# ls /var/named/master/fz
dsadova.net  dsadova.net.jnl
[root@server.dsadova.net ~]#
```

Рис. 44: Проверяем наличие файла

Анализ работы DHCP-сервера после настройки обновления DNS-зоны

На виртуальной машине client под вашим пользователем откройте терминал и с помощью утилиты dig убедитесь в наличии DNS-записи о клиенте в прямой DNS-зоне:

```
[dsadova@client ~]$ dig @192.168.1.1 client.dsadova.net

; <>> DiG 9.16.23-RH <>> @192.168.1.1 client.dsadova.net
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60274
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 985e85db29ba4bb90100000068ce7272bc3dd87896359493 (good)
;; QUESTION SECTION:
;client.dsadova.net.                IN      A

;; ANSWER SECTION:
client.dsadova.net.                1200    IN      A      192.168.1.30
```

- opcode: QUERY - тип операции: запрос
- status: NOERROR - успешное выполнение
- id: 60274 - идентификатор запроса
- flags: qr aa rd ra - флаги: qr - это ответ (query response), aa - авторитетный ответ (authoritative answer), rd - рекурсия запрошена (recursion desired), ra - рекурсия доступна (recursion available)

Секция QUESTION:

- client.dsadova.net. IN A - запрос IPv4 адреса для client.dsadova.net

Секция ANSWER:

- client.dsadova.net. 1200 IN A 192.168.1.30 - ключевой результат!
- Время жизни записи: 1200 секунд (20 минут)
- Класс: IN (Internet)
- Тип: A (IPv4 адрес)
- Адрес: 192.168.1.30 - полностью совпадает с DHCP-арендой

Внесение изменений в настройки внутреннего окружения виртуальной машины

1. На виртуальной машине server перейдите в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создайте в нём каталог dhcp, в который поместите в соответствующие подкаталоги конфигурационные файлы DHCP:

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

Рис. 46: Создаем каталог dhcp

2. Замените конфигурационные файлы DNS-сервера:

```
[root@server.dsadova.net server]# cd /vagrant/provision/server/dns/
cp -R /var/named/* /vagrant/provision/server/dns/var/named/
cp -R /etc/named/* /vagrant/provision/server/dns/etc/named/
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/dsadova.net'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/192.168.1'?
'?'
cp: overwrite '/vagrant/provision/server/dns/etc/named/dsadova.net'? y
[root@server.dsadova.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-key
s.bind'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-key
s.bind.jnl'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/dsadova.net'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/dsadova.net.jnl'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/192.168.1.jnl'? 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.dsadova.net dns]# y
bash: y: command not found...
[root@server.dsadova.net dns]#
```


3. В каталоге `/vagrant/provision/server` создайте исполняемый файл `dhcp.sh`. Открыв его на редактирование, пропишите в нём следующий скрипт:

```
GNU nano 5.6.1 dhcp.sh
#!/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
systemctl enable --now kea-dhcp4.service
systemctl enable --now kea-dhcp-ddns.service
```

4. Для отработки созданного скрипта во время загрузки виртуальной машины server в конфигурационном файле Vagrantfile необходимо добавить в разделе конфигурации для сервера:

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

Рис. 49: Переписываем файл Vagrantfile

- Приобрели практические навыки по установке и конфигурированию DHCP-сервера. Решили проблемы и неисправности при настройке и конфигурированию DHCP-сервера.