

Отчёт по лабораторной работе 3

Настройка DHCP-сервера

Чилеше Лупупа

Содержание

1	Цель работы	5
2	Выполнение работы	6
2.1	Конфигурирование DHCP-сервера	6
2.1.1	Сохранение конфигурационного файла	6
2.1.2	Редактирование файла kea-dhcp4.conf	6
2.1.3	Проверка конфигурации	9
2.1.4	Перезагрузка конфигурации и включение службы	9
2.2	Настройка DNS-зон	9
2.2.1	Прямая зона	9
2.2.2	Обратная зона	10
2.2.3	Перезапуск named и проверка работы DNS	11
2.2.4	Настройка межсетевого экрана и SELinux	11
2.2.5	Запуск DHCP-сервера	12
2.3	Анализ работы DHCP-сервера	12
2.3.1	Настройка маршрутизации клиента	12
2.3.2	Анализ сетевых интерфейсов на клиентской машине	13
2.3.3	Анализ файла /var/lib/kea/kea-leases4.csv	14
2.4	Настройка обновления DNS-зоны	16
2.4.1	Создание и подключение TSIG-ключа для динамического обновления зон	16
2.4.2	Генерация ключа	16
2.4.3	Подключение ключа в конфигурацию named	16
2.4.4	Разрешение обновления прямой и обратной зон	17
2.4.5	Создание ключа для Kea DHCP	19
2.4.6	Настройка Kea DHCP-DDNS	19
2.4.7	Разрешение DHCP-сервису обновлять DNS-записи	21
3	Анализ работы DHCP-сервера после настройки обновления DNS-зоны	23
3.1	Внесение изменений в настройки внутреннего окружения виртуальной машины	24
4	Вывод	26
5	Контрольные вопросы	27

Список иллюстраций

2.1 Резервное копирование конфигурации	6
2.2 Редактирование domain-name и domain-search	7
2.3 Настройка подсети DHCP	8
2.4 Проверка конфигурации kea	9
2.5 Редактирование прямой зоны	10
2.6 Редактирование обратной зоны	10
2.7 Проверка разрешения имени DHCP	11
2.8 Firewall и restorecon	11
2.9 Скрипт маршрутизации клиента	12
2.10 Информация об интерфейсах клиента	13
2.11 Содержимое файла kea-leases4.csv	14
2.12 Генерация TSIG-ключа	16
2.13 Вставка ключа в named.conf	17
2.14 Настройка update-policy для прямой зоны	18
2.15 Настройка update-policy для обратной зоны	18
2.16 Конфигурация tsig-keys.json	19
2.17 Конфигурация kea-dhcp-ddns.conf	20
2.18 Статус службы Kea DDNS	20
2.19 dhcp4 конфигурация с DDNS параметрами	21
2.20 Статус Kea DHCP после перезапуска	22
3.1 Результат dig	23
3.2 dhcp.sh	25

Список таблиц

1 Цель работы

Приобретение практических навыков по установке и конфигурированию DHCP-сервера.

2 Выполнение работы

2.1 Конфигурирование DHCP-сервера

2.1.1 Сохранение конфигурационного файла

Перед внесением изменений была выполнена резервная копия основного конфигурационного файла DHCP-сервера:

```
cp /etc/kea/kea-dhcp4.conf /etc/kea/kea-dhcp4.conf__$(date -I)
```

На скриншоте показан результат выполнения команды:

```
Installed:
  kea-2.6.3-1.el10_0.x86_64          kea-libs-2.6.3-1.el10_0.x86_64
  libpq-16.8-2.el10_0.x86_64        log4cplus-2.1.1-8.el10.x86_64
  mariadb-connector-c-3.4.4-1.el10.x86_64  mariadb-connector-c-config-3.4.4-1.el10.noarch

Complete!
[root@server.chileshe.net ~]#
[root@server.chileshe.net ~]# cp /etc/kea/kea-dhcp4.conf /etc/kea/kea-dhcp4.conf__$(date -I)
[root@server.chileshe.net ~]# gedit /etc/kea/kea-dhcp4.conf
[root@server.chileshe.net ~]#
```

Рис. 2.1: Резервное копирование конфигурации

2.1.2 Редактирование файла kea-dhcp4.conf

Файл `/etc/kea/kea-dhcp4.conf` был открыт для редактирования. В блоке параметров были внесены следующие изменения.

2.1.2.1 Замена значений `domain-name` и `domain-search`

Исходные записи:

```
{ "code": 15, "data": "example.org" }, { "name": "domain-search", "data": "srv.world" }
```

были заменены на:

```
{ "code": 15, "data": "chileshe.net" }, { "name": "domain-search", "data":  
"chileshe.net" }
```

Фрагмент редактируемого файла представлен на изображении:

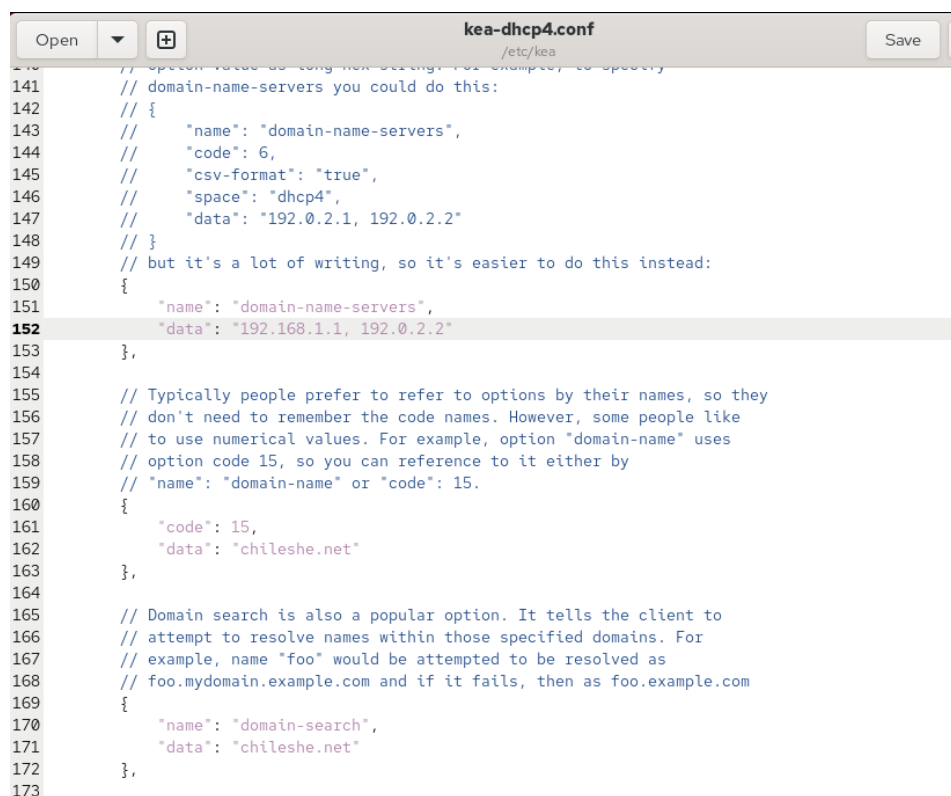


Рис. 2.2: Редактирование domain-name и domain-search

2.1.2.2 Измена параметра domain-name-servers

Значения по умолчанию:

```
{ "name": "domain-name-servers", "data": "192.0.2.1, 192.0.2.2" }
```

были заменены на:

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

2.1.2.3 Настройка подсети

На основе примера в конфигурационном файле добавлена собственная DHCP-сеть:

- подсеть: 192.168.1.0/24
- диапазон выдаваемых адресов: 192.168.1.30–192.168.1.199
- шлюз: 192.168.1.1
- broadcast-адрес выбирается автоматически на основе подсети

Использованный блок:

```
"subnet4": [ { "id": 1, "subnet": "192.168.1.0/24", "pools": [ { "pool": "192.168.1.30 - 192.168.1.199" } ], "option-data": [ { "name": "routers", "data": "192.168.1.1" } ] },
```

Оставшиеся шаблонные подсети были удалены.

Фрагмент конфигурации показан на скриншоте:



```
279 // }
280 // ],
281
282 // Below an example of a simple IPv4 subnet declaration. Uncomment to enable
283 // it. This is a list, denoted with [ ], of structures, each denoted with
284 // { }. Each structure describes a single subnet and may have several
285 // parameters. One of those parameters is "pools" that is also a list of
286 // structures.
287 "subnet4": [
288     {
289         "id": 1,
290         "subnet": "192.168.1.0/24",
291         "pools": [ { "pool": "192.168.1.30 - 192.168.1.199" } ],
292         "option-data": [
293             {
294                 // For each IPv4 subnet you most likely need to specify at
295                 // least one router.
296                 "name": "routers",
297                 "data": "192.168.1.1"
298             }
299         ]
300     }
301 ]
302 // You can add more subnets there.
303
304 ]
305
306 }
307
308 ]
```

Рис. 2.3: Настройка подсети DHCP

2.1.2.4 Привязка DHCP к интерфейсу eth1

Добавлен блок:

```
“interfaces-config”: { “interfaces”: [ “eth1” ] },
```

2.1.3 Проверка конфигурации

Валидность файла проверена командой:

```
kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
```

На изображении показан результат успешной проверки:

```
[root@server.chileshe.net ~]#
[root@server.chileshe.net ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-11-16 10:43:11.605 INFO [kea-dhcp4.hosts/16551.140228964001984] HOSTS_BACKENDS_REGISTERED the following host bac
kend types are available: mysql postgresql
2025-11-16 10:43:11.606 WARN [kea-dhcp4.dhcp4/16551.140228964001984] DHCP4_MT_DISABLED_QUEUE_CONTROL disabling dh
cp queue control when multi-threading is enabled.
2025-11-16 10:43:11.606 WARN [kea-dhcp4.dhcp4/16551.140228964001984] DHCP4_RESERVATIONS_LOOKUP_FIRST_ENABLED Multi-th
reading is enabled and host reservations lookup is always performed first.
2025-11-16 10:43:11.606 INFO [kea-dhcp4.dhcp4/16551.140228964001984] DHCP4_CFGMGR_NEW_SUBNET4 a new subnet has be
en added to configuration: 192.168.1.0/24 with params: t1=900, t2=1800, valid-lifetime=3600
2025-11-16 10:43:11.606 INFO [kea-dhcp4.dhcp4/16551.140228964001984] DHCP4_CFGMGR_SOCKET_TYPE_SELECT using socket
type raw
2025-11-16 10:43:11.606 INFO [kea-dhcp4.dhcp4/16551.140228964001984] DHCP4_CFGMGR_ADD_IFACE listening on interfac
e eth1
2025-11-16 10:43:11.606 INFO [kea-dhcp4.dhcp4/16551.140228964001984] DHCP4_CFGMGR_SOCKET_TYPE_DEFAULT "dhcp-socket-
type" not specified, using default socket type raw
[root@server.chileshe.net ~]#
[root@server.chileshe.net ~]# systemctl --system daemon-reload
[root@server.chileshe.net ~]# systemctl enable kea-dhcp4.service
Created symlink '/etc/systemd/system/multi-user.target.wants/kea-dhcp4.service' -> '/usr/lib/systemd/system/kea-dhcp4.s
ervice'.
[root@server.chileshe.net ~]#
```

Рис. 2.4: Проверка конфигурации kea

2.1.4 Перезагрузка конфигурации и включение службы

```
systemctl --system daemon-reload
```

```
systemctl enable kea-dhcp4.service
```

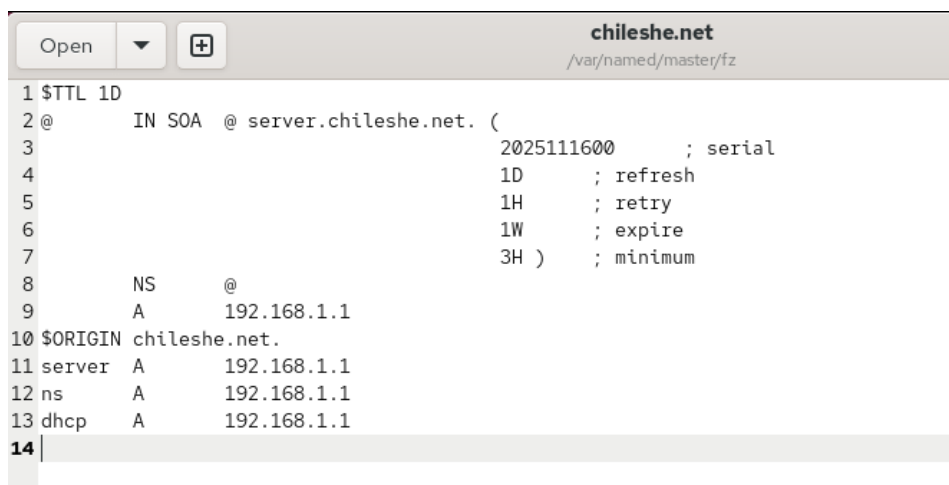
2.2 Настройка DNS-зон

2.2.1 Прямая зона

В конец файла /var/named/master/fz/chileshe.net добавлена запись:

```
dhcp A 192.168.1.1
```

Также обновлён серийный номер зоны.



```
1 $TTL 1D
2 @      IN SOA  @ server.chileshe.net. (
3                               2025111600      ; serial
4                               1D              ; refresh
5                               1H              ; retry
6                               1W              ; expire
7                               3H )            ; minimum
8      NS   @
9      A    192.168.1.1
10 $ORIGIN chileshe.net.
11 server A    192.168.1.1
12 ns     A    192.168.1.1
13 dhcp   A    192.168.1.1
14
```

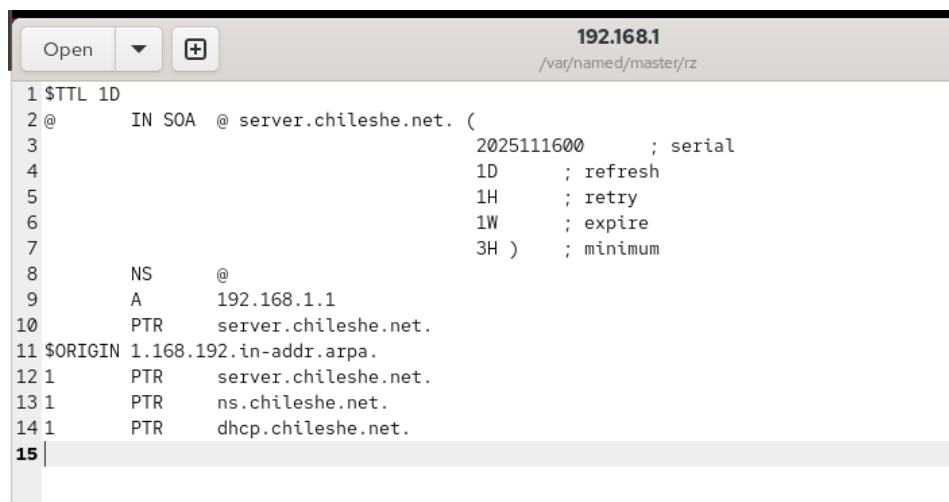
Рис. 2.5: Редактирование прямой зоны

2.2.2 Обратная зона

В файл `/var/named/master/rz/192.168.1` добавлена запись:

1 PTR dhcp.chileshe.net.

Серийный номер зоны также обновлён.



```
1 $TTL 1D
2 @      IN SOA  @ server.chileshe.net. (
3                               2025111600      ; serial
4                               1D              ; refresh
5                               1H              ; retry
6                               1W              ; expire
7                               3H )            ; minimum
8      NS   @
9      A    192.168.1.1
10     PTR   server.chileshe.net.
11 $ORIGIN 1.168.192.in-addr.arpa.
12 1       PTR   server.chileshe.net.
13 1       PTR   ns.chileshe.net.
14 1       PTR   dhcp.chileshe.net.
15
```

Рис. 2.6: Редактирование обратной зоны

2.2.3 Перезапуск named и проверка работы DNS

systemctl restart named

ping dhcp.chileshe.net

Результат пинга подтверждает успешное разрешение имени:

```
[root@server.chileshe.net ~]# systemctl restart named
[root@server.chileshe.net ~]# ping dhcp.chileshe.net
PING dhcp.chileshe.net (192.168.1.1) 56(84) bytes of data:
64 bytes from server.chileshe.net (192.168.1.1): icmp_seq=1 ttl=64 time=0.013 ms
64 bytes from server.chileshe.net (192.168.1.1): icmp_seq=2 ttl=64 time=0.049 ms
64 bytes from server.chileshe.net (192.168.1.1): icmp_seq=3 ttl=64 time=0.107 ms
64 bytes from server.chileshe.net (192.168.1.1): icmp_seq=4 ttl=64 time=0.045 ms
64 bytes from server.chileshe.net (192.168.1.1): icmp_seq=5 ttl=64 time=0.109 ms
64 bytes from server.chileshe.net (192.168.1.1): icmp_seq=6 ttl=64 time=0.024 ms
64 bytes from server.chileshe.net (192.168.1.1): icmp_seq=7 ttl=64 time=0.051 ms
```

Рис. 2.7: Проверка разрешения имени DHCP

2.2.4 Настройка межсетевого экрана и SELinux

Для разрешения работы DHCP были добавлены необходимые правила:

firewall-cmd --add-service=dhcp

firewall-cmd --add-service=dhcp --permanent

Далее восстановлены контексты SELinux:

restorecon -vR /etc

restorecon -vR /var/named

restorecon -vR /var/lib/kea/

Результат операций:

```
[root@server.chileshe.net ~]#
[root@server.chileshe.net ~]# firewall-cmd --add-service=dhcp
success
[root@server.chileshe.net ~]# firewall-cmd --add-service=dhcp --permanent
success
[root@server.chileshe.net ~]# restorecon -vR /etc
Relabeled /etc/NetworkManager/system-connections/eth1.nmconnection from unconfined_u:object_r:user_tmp_t:s0 to unconfi
ned_u:object_r:NetworkManager_etc_rw_t:s0
[root@server.chileshe.net ~]# restorecon -vR /var/named/
[root@server.chileshe.net ~]# restorecon -vR /var/lib/kea/
[root@server.chileshe.net ~]# systemctl start kea-dhcp4.service
[root@server.chileshe.net ~]#
```

Рис. 2.8: Firewall и restorecon

2.2.5 Запуск DHCP-сервера

Во вспомогательном терминале был включён мониторинг системных сообщений:

```
tail -f /var/log/messages
```

В основном терминале произведён запуск службы:

```
systemctl start kea-dhcp4.service
```

2.3 Анализ работы DHCP-сервера

2.3.1 Настройка маршрутизации клиента

Перед запуском виртуальной машины **client** был создан файл 01-routing.sh в каталоге vagrant/provision/client.

В него был внесён скрипт, который перенастраивает NetworkManager так, чтобы весь исходящий трафик по умолчанию шёл через интерфейс eth1.

Этот скрипт выполняет следующие действия:

- назначает для соединения *eth1* шлюз 192.168.1.1;
- активирует соединение *eth1*;
- запрещает использовать *eth0* как маршрут по умолчанию для IPv4 и IPv6;
- перезапускает соединение *eth0*.

Фрагмент скрипта:

```
1  #!/bin/bash
2  echo "Provisioning script $0"
3  nmcli connection modify "eth1" ipv4.gateway "192.168.1.1"
4  nmcli connection up "eth1"
5  nmcli connection modify eth0 ipv4.never-default true
6  nmcli connection modify eth0 ipv6.never-default true
7  nmcli connection down eth0
8  nmcli connection up eth0
9  # systemctl restart NetworkManager
10
```

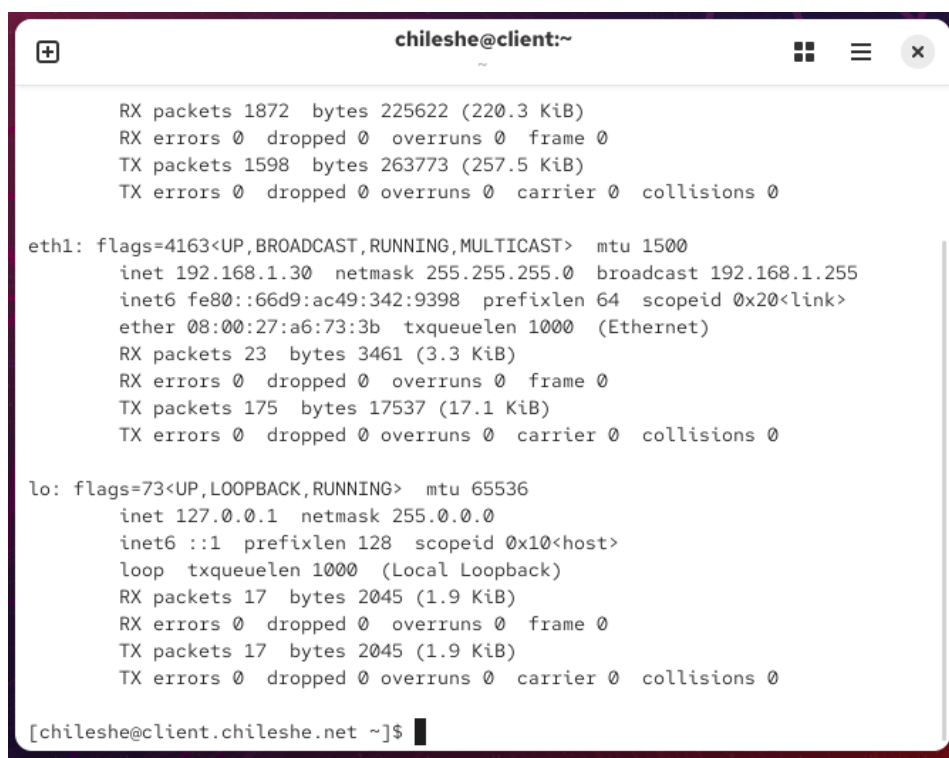
Рис. 2.9: Скрипт маршрутизации клиента

Скрипт был подключён в секции конфигурации клиента внутри Vagrantfile, после чего клиентская машина была поднята и провиженинг выполнен.

2.3.2 Анализ сетевых интерфейсов на клиентской машине

После загрузки клиентской виртуальной машины и применения сетевых настроек был выполнен просмотр интерфейсов командой `ifconfig`.

Результат:



```
chileshe@client:~  
  
RX packets 1872  bytes 225622 (220.3 KiB)  
RX errors 0  dropped 0  overruns 0  frame 0  
TX packets 1598  bytes 263773 (257.5 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::66d9:ac49:342:9398  prefixlen 64  scopeid 0x20<link>  
    ether 08:00:27:a6:73:3b  txqueuelen 1000  (Ethernet)  
    RX packets 23  bytes 3461 (3.3 KiB)  
    RX errors 0  dropped 0  overruns 0  frame 0  
    TX packets 175  bytes 17537 (17.1 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  
  
[chileshe@client.chileshe.net ~]$
```

Рис. 2.10: Информация об интерфейсах клиента

Разбор данных по интерфейсу `eth1`:

- интерфейс активирован, работает в режиме **UP, BROADCAST, RUNNING, MULTICAST**;
- IP-адрес: 192.168.1.30;
- маска сети: 255.255.255.0;
- broadcast-адрес: 192.168.1.255;

- MAC-адрес: 08:00:27:a6:73:3b;
- интерфейс передаёт и принимает пакеты без ошибок, что подтверждает корректную работу полученного от DHCP параметров.

Интерфейс lo представляет собой стандартный интерфейс loopback, используется для локальных обращений к машине.

2.3.3 Анализ файла /var/lib/kea/kea-leases4.csv

На сервере был просмотрен файл с информацией о выданных DHCP-адресах. Файл отражает каждую запись о выдаче адреса клиентам.

```
[root@server.chileshe.net ~]#
[root@server.chileshe.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:a6:73:3b,01:08:00:27:a6:73:3b,3600,1763293865,1,0,0,client,0,,0
192.168.1.30,08:00:27:a6:73:3b,01:08:00:27:a6:73:3b,3600,1763293865,1,0,0,client,0,,0
192.168.1.30,08:00:27:a6:73:3b,01:08:00:27:a6:73:3b,3600,1763293870,1,0,0,client,0,,0
[root@server.chileshe.net ~]#
```

Рис. 2.11: Содержимое файла kea-leases4.csv

Разбор строк файла:

1. **address** — выданный клиенту IPv4-адрес:

192.168.1.30

2. **hwaddr** — MAC-адрес клиента:

08:00:27:a6:73:3b

Этот адрес совпадает с MAC-адресом интерфейса eth1 на клиентской машине.

3. **client_id** — идентификатор клиента:

01:08:00:27:a6:73:3b

Здесь 01 означает идентификатор Ethernet, далее идёт MAC-адрес.

4. **valid_lifetime** — время жизни аренды в секундах:

3600 (1 час)

5. **expire** — время истечения аренды (UNIX timestamp):
значения отличаются для разных аренд.
6. **subnet_id** — идентификатор подсети, из которой выделен адрес:
1
Это соответствует идентификатору, заданному в конфигурации DHCP.
7. **fqdn_fwd** и **fqdn_rev** — управление прямой и обратной DNS-записями:
значение 1 означает включённое обновление.
8. **hostname** — имя клиента:
указывается как `client`
9. **state** — состояние аренды:
0 означает *active*.
10. **user_context** — дополнительные данные пользователя, отсутствуют (пусто).
11. **pool_id** — идентификатор пула, из которого выдан адрес:
0
Так как в конфигурации задан только один пул.

Все записи подтверждают, что DHCP-сервер корректно выдал адрес клиентской машине и обновил аренду несколько раз, что отражено в файле несколькими строками.

2.4 Настройка обновления DNS-зоны

2.4.1 Создание и подключение TSIG-ключа для динамического обновления зон

2.4.2 Генерация ключа

Для обеспечения безопасного обновления DNS-записей был создан каталог для хранения ключей и сгенерирован TSIG-ключ алгоритмом **HMAC-SHA512**.

Результат выполнения команд и содержимое ключа:

```
[root@server.chileshe.net ~]#  
[root@server.chileshe.net ~]# mkdir -p /etc/named/keys  
[root@server.chileshe.net ~]# tsig-keygen -a HMAC-SHA512 DHCP_UPDATER > /etc/named/keys/dhcp_updater.key  
[root@server.chileshe.net ~]# cat /etc/named/keys/dhcp_updater.key  
key "DHCP_UPDATER" {  
    algorithm hmac-sha512;  
    secret "g3bICzGG3iM4vLjJhh1E8XsAlUHwrM5rVzi93JYc0B7dnzpnUFg0GqeWSsT0Gt2ju1vswV9ZrrKL6qPeJbtYEA=";  
};  
[root@server.chileshe.net ~]# chown -R named:named /etc/named/keys/  
[root@server.chileshe.net ~]#
```

Рис. 2.12: Генерация TSIG-ключа

После генерации ключ был помещён в файл `/etc/named/keys/dhcp_updater.key` и получил корректные права доступа, принадлежащие пользователю и группе **named**.

2.4.3 Подключение ключа в конфигурацию named

В файл `/etc/named.conf` был добавлен `include`-блок: `include "/etc/named/keys/dhcp_updater`
Это позволяет серверу BIND9 использовать TSIG-ключ для проверки авторизованных обновлений зон.

Фрагмент конфигурации:


```

44
45 logging {
46     channel default_debug {
47         file "data/named.run";
48         severity dynamic;
49     };
50 };
51
52 zone "." IN {
53     type hint;
54     file "named.ca";
55 };
56
57 include "/etc/named.rfc1912.zones";
58 include "/etc/named.root.key";
59 include "/etc/named/chileshe.net";
60 include "/etc/named/keys/dhcp_updater.key";

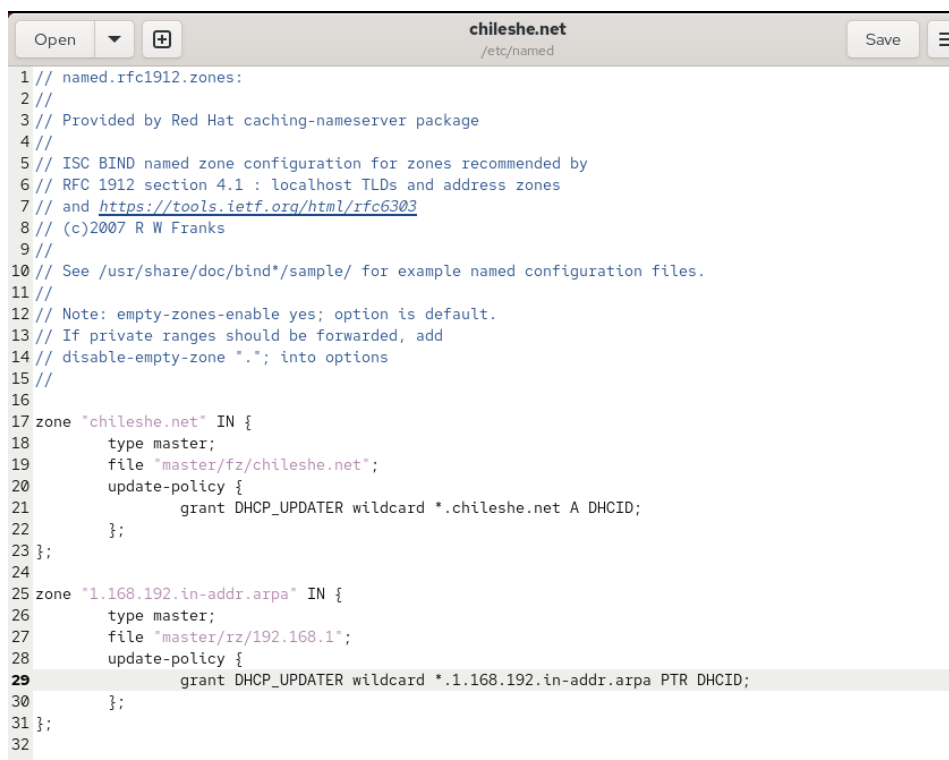
```

Рис. 2.13: Вставка ключа в named.conf

2.4.4 Разрешение обновления прямой и обратной зон

Файлы зон были скорректированы так, чтобы DHCP-сервер мог автоматически добавлять и изменять записи A/PTR.

В прямой зоне:



```
1 // named.rfc1912.zones:
2 //
3 // Provided by Red Hat caching-nameserver package
4 //
5 // ISC BIND named zone configuration for zones recommended by
6 // RFC 1912 section 4.1 : localhost TLDs and address zones
7 // and https://tools.ietf.org/html/rfc6303
8 // (c)2007 R W Franks
9 //
10 // See /usr/share/doc/bind*/sample/ for example named configuration files.
11 //
12 // Note: empty-zones-enable yes; option is default.
13 // If private ranges should be forwarded, add
14 // disable-empty-zone "."; into options
15 //
16
17 zone "chileshe.net" IN {
18     type master;
19     file "master/fz/chileshe.net";
20     update-policy {
21         grant DHCP_UPDATER wildcard *.chileshe.net A DHCID;
22     };
23 };
24
25 zone "1.168.192.in-addr.arpa" IN {
26     type master;
27     file "master/rz/192.168.1";
28     update-policy {
29         grant DHCP_UPDATER wildcard *.1.168.192.in-addr.arpa PTR DHCID;
30     };
31 };
32
```

Рис. 2.14: Настройка update-policy для прямой зоны

В обратной зоне:



```
1 "tsig-keys" [
2     {
3         "name": "DHCP_UPDATER",
4         "algorithm": "hmac-sha512",
5         "secret":
6         "g3bICzGG3iM4vLjJhhLE8XsA1UHwrM5rVzi93JYcOB7dnzpnUFg0GqeWSsTOgt2ju1vswV9ZrrKL6qPeJbtYEA=="
7     }
8 ]
```

Рис. 2.15: Настройка update-policy для обратной зоны

Использована директива: `grant DHCP_UPDATER wildcard *.chileshe.net A DHCID;`

а также аналогичная для PTR-записей в обратной зоне.

После внесения изменений конфигурация прошла успешную проверку и DNS-сервер был перезапущен.

2.4.5 Создание ключа для Kea DHCP

Для взаимодействия DHCP-сервера Kea с BIND была создана JSON-версия TSIG-ключа:

```
21 {
22   "ip-address": "127.0.0.1",
23   "port": 53001,
24   "control-socket": {
25     "socket-type": "unix",
26     "socket-name": "/run/kea/kea-ddns-ctrl-socket"
27   },
28   <?include "/etc/kea/tsig-keys.json" ?>
29
30   "forward-ddns" : {
31     "ddns-domains" : [
32       {
33         "name": "chileshe.net.",
34         "key-name": "DHCP_UPDATER",
35         "dns-servers": [
36           { "ip-address": "192.168.1.1" }
37         ]
38       }
39     ]
40   },
41
42   "reverse-ddns" : {
43     "ddns-domains" : [
44       {
45         "name": "1.168.192.in-addr.arpa.",
46         "key-name": "DHCP_UPDATER",
47         "dns-servers": [
48           { "ip-address": "192.168.1.1" }
49         ]
50       }
51     ]
52   },
```

Рис. 2.16: Конфигурация tsig-keys.json

Файл получил корректные права и владельца (kea:kea).

2.4.6 Настройка Kea DHCP-DDNS

В файл /etc/kea/kea-dhcp-ddns.conf были внесены параметры:

- IP-адрес и порт локального DDNS-сервиса Kea;
- подключение файла ключей;
- настройки **forward-ddns** и **reverse-ddns**, указывающие имя зоны, TSIG-ключ и DNS-сервер BIND;
- логирование.

Фрагмент конфигурации:

```
53
54 // Logging configuration starts here. Kea uses different loggers to log various
55 // activities. For details (e.g. names of loggers), see Chapter 18.
56 "loggers": [
57   {
58     "name": "kea-dhcp-ddns",
59     "output-options": [
60       {
61         "output": "stdout",
62         "pattern": "%-5p %m\n"
63       }
64     ]
65   }
66 ],
67 // This specifies the severity of log messages to keep. Supported values
68 // are: FATAL, ERROR, WARN, INFO, DEBUG
69 "severity": "INFO",
70 // If DEBUG level is specified, this value is used. 0 is least verbose,
71 // 99 is most verbose. Be cautious, Kea can generate lots and lots
72 // of logs if told to do so.
73 "debuglevel": 0
74 }
75 ]
76 }
```

Рис. 2.17: Конфигурация kea-dhcp-ddns.conf

После проверки синтаксиса служба была включена и успешно запущена:

```
[root@server.chileshe.net ~]# chown kea:kea /etc/kea/kea-dhcp-ddns.conf
[root@server.chileshe.net ~]# kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
2025-11-16 11:11:57.243 INFO [kea-dhcp-ddns.dctl/20846.139770689831232] DCTL_CONFIG_CHECK_COMPLETE server has completed configuration check: listening on 127.0.0.1, port 53001, using UDP, result: success(0), text=Configuration check successful
[root@server.chileshe.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.chileshe.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 Sun 2025-11-16 11:12:16 UTC; 8s ago
 Invocation: 975d7e85b87145789bb7754902f4a873
    Docs: man:kea-dhcp-ddns(8)
   Main PID: 21065 (kea-dhcp-ddns)
      Tasks: 5 (limit: 10381)
     Memory: 1.7M (peak: 6.1M)
        CPU: 10ms
    CGroup: /system.slice/kea-dhcp-ddns.service
            └─21065 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Nov 16 11:12:16 server.chileshe.net systemd[1]: Started kea-dhcp-ddns.service - Kea DHCP-DDNS Server.
Nov 16 11:12:16 server.chileshe.net kea-dhcp-ddns[21065]: 2025-11-16 11:12:16.089 INFO [kea-dhcp-ddns.dctl/21065.1406] COMMAND_ACCEPTOR_START Starting to accept connections
Nov 16 11:12:16 server.chileshe.net kea-dhcp-ddns[21065]: INFO DCTL_CONFIG_COMPLETE server has completed configuration
Nov 16 11:12:16 server.chileshe.net kea-dhcp-ddns[21065]: INFO DHCP_DDNS_STARTED Kea DHCP-DDNS server version 2.6.3
lines 1-17/17 (END)
```

Рис. 2.18: Статус службы Кеа DDNS

2.4.7 Разрешение DHCP-сервису обновлять DNS-записи

В основной конфигурационный файл DHCP /etc/kea/kea-dhcp4.conf были внесены параметры:

- включение DDNS-обновлений;
- указание доменного суффикса (chileshe.net);
- принудительное переопределение данных от клиента.

Фрагмент файла:

```
28 "Dhcp4": {
29     // Add names of your network interfaces to listen on.
30     "interfaces-config": {
31         // See section 8.2.4 for more details. You probably want to add just
32         // interface name (e.g. "eth0" or specific IPv4 address on that
33         // interface name (e.g. "eth0/192.0.2.1").
34         "interfaces": [ "eth1" ]
35
36         // Kea DHCPv4 server by default listens using raw sockets. This ensures
37         // all packets, including those sent by directly connected clients
38         // that don't have IPv4 address yet, are received. However, if your
39         // traffic is always relayed, it is often better to use regular
40         // UDP sockets. If you want to do that, uncomment this line:
41         // "dhcp-socket-type": "udp"
42     },
43
44     "dhcp-ddns": {
45         "enable-updates": true
46     },
47     "ddns-qualifying-suffix": "chileshe.net",
48     "ddns-override-client-update": true,
49     // Kea supports control channel, which is a way to receive management
50     // commands while the server is running. This is a Unix domain socket that
51     // receives commands formatted in JSON, e.g. config-set (which sets new
```

Рис. 2.19: dhcp4 конфигурация с DDNS параметрами

Файл успешно прошёл проверку, и служба DHCP была перезапущена:

```

[root@server.chileshe.net ~]#
[root@server.chileshe.net ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-11-16 11:14:52.274 INFO [kea-dhcp4.hosts/21438.140623153055936] HOSTS_BACKENDS_REGISTERED the following host back
end types are available: mysql postgresql
2025-11-16 11:14:52.274 WARN [kea-dhcp4.dhcp4/21438.140623153055936] DHCP4_RESERVED_QUEUE_CONTROL disabling dh
cp queue control when multi-threading is enabled.
2025-11-16 11:14:52.274 WARN [kea-dhcp4.dhcp4/21438.140623153055936] DHCP4_RESERVED_QUEUE_LOOKUP_FIRST_ENABLED Multi-th
reading is enabled and host reservations lookup is always performed first.
2025-11-16 11:14:52.274 INFO [kea-dhcp4.dhcp4/21438.140623153055936] DHCP4_RESERVED_QUEUE_NEW_SUBNET4 a new subnet has be
en added to configuration: 192.168.1.0/24 with params: t1=900, t2=1800, valid-lifetime=3600
2025-11-16 11:14:52.274 INFO [kea-dhcp4.dhcp4/21438.140623153055936] DHCP4_RESERVED_QUEUE_SOCKET_TYPE_SELECT using socket
type raw
2025-11-16 11:14:52.274 INFO [kea-dhcp4.dhcp4/21438.140623153055936] DHCP4_RESERVED_QUEUE_ADD_IFACE listening on interfac
e eth1
2025-11-16 11:14:52.274 INFO [kea-dhcp4.dhcp4/21438.140623153055936] DHCP4_RESERVED_QUEUE_SOCKET_TYPE_DEFAULT "dhcp-socket-
type" not specified, using default socket type raw
[root@server.chileshe.net ~]#
[root@server.chileshe.net ~]# systemctl restart kea-dhcp4.service
[root@server.chileshe.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 Sun 2025-11-16 11:15:08 UTC; 6s ago
  Invocation: de75f0d0f27f43bda07c31097ff84384
     Docs: man:kea-dhcp4(8)
    Main PID: 21534 (kea-dhcp4)
       Tasks: 7 (limit: 10381)
      Memory: 2.5M (peak: 6M)
         CPU: 13ms
    CGroup: /system.slice/kea-dhcp4.service
            └─21534 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf

Nov 16 11:15:08 server.chileshe.net systemd[1]: Started kea-dhcp4.service - Kea DHCPv4 Server.
Nov 16 11:15:08 server.chileshe.net kea-dhcp4[21534]: 2025-11-16 11:15:08.196 INFO [kea-dhcp4.dhcp4/21534.1396269388]
Nov 16 11:15:08 server.chileshe.net kea-dhcp4[21534]: 2025-11-16 11:15:08.196 INFO [kea-dhcp4.commands/21534.1396269388]
[root@server.chileshe.net ~]#

```

Рис. 2.20: Статус Kea DHCP после перезапуска

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

После настройки DDNS на виртуальной машине **client** была выполнена команда: `dig @192.168.1.1 client.chileshe.net`

Результат:

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

; <<>> DiG 9.18.33 <<>> @192.168.1.1 client.chileshe.net
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 11008
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: afc53246b429f390010000006919b2f6f69971e0db8f348b (good)
;; QUESTION SECTION:
;client.chileshe.net.          IN      A

;; ANSWER SECTION:
client.chileshe.net.  1200    IN      A      192.168.1.30

;; Query time: 1 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sun Nov 16 11:18:14 UTC 2025
;; MSG SIZE rcvd: 92

[chileshe@client.chileshe.net ~]$
```

Рис. 3.1: Результат dig

Построчный разбор:

- **DiG 9.18.33 [192.168.1.1?] client.chileshe.net** — используется утилита dig, запрос отправлен DNS-серверу по адресу 192.168.1.1, запрашивается запись

client.chileshe.net.

- **(1 server found)** — найден один DNS-сервер.
- **Got answer** — ответ получен успешно.
- **opcode: QUERY, status: NOERROR** — тип операции: запрос; ошибок нет.
- **QUERY: 1, ANSWER: 1** — один вопрос и один найденный ответ.
- **client.chileshe.net. IN A** — запрашивалась A-запись.
- **ANSWER SECTION** — секция с ответом:

– **client.chileshe.net. 1200 IN A 192.168.1.30**

DNS-сервер сообщает, что у узла client.chileshe.net IP-адрес:
192.168.1.30, выданный DHCP.

- **SERVER: 192.168.1.1** — отвечает локальный DNS-сервер.
- **QUERY TIME: 1 msec** — время ответа минимально.
- **MSG SIZE rcvd: 92** — размер полученного DNS-пакета.

Это подтверждает корректную работу DDNS-обновлений:

DHCP-сервер автоматически создал запись A в прямой зоне.

3.1 Внесение изменений в настройки внутреннего окружения виртуальной машины

В каталоге /vagrant/provision/server/ был создан сценарий:


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

Рис. 3.2: dhcp.sh

4 Вывод

В ходе выполнения работы была настроена комплексная система автоматической выдачи сетевых параметров и динамического обновления DNS-записей. DHCP-сервер Kea был интегрирован с DNS-сервером Bind9 посредством механизма TSIG-подписей, что позволило обеспечить безопасное и корректное обновление прямой и обратной DNS-зоны при подключении новых узлов. Конфигурационные файлы DHCP, DDNS и DNS были адаптированы под заданную сетевую архитектуру, а необходимые службы успешно прошли проверку и были запущены.

После настройки клиентская виртуальная машина автоматически получила IP-адрес от DHCP-сервера, а соответствующие A- и PTR-записи были добавлены в зоны chileshe.net и 1.168.192.in-addr.arpa. Проверка с помощью утилиты dig показала корректное разрешение имени клиента, что подтвердило работоспособность DDNS-механизма. Дополнительно была выполнена подготовка внутренних конфигураций Vagrant, позволяющая полностью автоматизировать развёртывание сервера DHCP и DNS в дальнейшем. Все цели лабораторной работы были достигнуты, а функциональность сетевой инфраструктуры подтверждена тестированием.

5 Контрольные вопросы

1. В каких файлах хранятся настройки сетевых подключений?

Настройки сетевых подключений в Linux обычно хранятся в каталоге `/etc/sysconfig/network-scripts/` (для систем семейства RHEL, Rocky, CentOS), где каждый интерфейс представлен отдельным файлом вида `ifcfg-eth0`, `ifcfg-eth1` и т. д.

В современных системах, использующих NetworkManager, конфигурации могут храниться в каталоге `/etc/NetworkManager/system-connections/` в формате `keyfile`. Эти файлы определяют параметры IP-адресации, шлюз, DNS-серверы, маршруты и тип соединения.

2. За что отвечает протокол DHCP?

DHCP (Dynamic Host Configuration Protocol) отвечает за автоматическую выдачу сетевых параметров клиентским устройствам. Он позволяет компьютерам получать IP-адрес, маску подсети, gateway, DNS-серверы и другие параметры без ручной настройки.

3. Поясните принцип работы протокола DHCP. Какими сообщениями обмениваются клиент и сервер?

Работа DHCP основана на обмене четырьмя основными сообщениями:

- **DHCPDISCOVER** — клиент ищет DHCP-сервер в сети и отправляет широковещательный запрос.
- **DHCPOFFER** — сервер откликается и предлагает свободный IP-адрес.
- **DHCPREQUEST** — клиент принимает одно из предложений и запрашивает конкретный адрес у сервера.

- **ДНСПАК** — сервер подтверждает выдачу адреса и передает клиенту полную конфигурацию.

Этот процесс обеспечивает автоматическую и согласованную настройку сетевых параметров клиентских устройств.

4. В каких файлах обычно находятся настройки DHCP-сервера? За что отвечает каждый?

В DHCP-сервере Kea настройки располагаются в каталоге `/etc/kea/`. Основные файлы: - **kea-dhcp4.conf** — основная конфигурация DHCPv4: параметры подсетей, диапазоны выдачи адресов, опции клиентов.

- **kea-dhcp-ddns.conf** — настройки динамического обновления DNS (DDNS).

- **tsig-keys.json** — хранит TSIG-ключи для авторизованных обновлений зон DNS.

- **lease-файлы (например, kea-leases4.csv)** — таблицы активных аренд (lease), содержащие информацию о выданных IP-адресах.

5. Что такое DDNS? Для чего применяется DDNS?

DDNS (Dynamic DNS) — механизм динамического обновления DNS-зон.

Он используется, чтобы автоматически регистрировать A- и PTR-записи для устройств, получающих адреса по DHCP. Например, когда клиент получил IP-адрес, сервер автоматически добавляет записи вида: - `client.example.net` → `192.168.1.10` (A-запись) - `10.1.168.192.in-addr.arpa` → `client.example.net` (PTR-запись)

DDNS обеспечивает актуальность DNS при динамической адресации устройств.

6. Какую информацию можно получить с помощью ifconfig? Приведите примеры с опциями.

Утилита `ifconfig` позволяет просматривать и настраивать сетевые интерфейсы. Она показывает: - IP-адрес интерфейса, - маску подсети, - MAC-адрес, - статистику переданных и полученных пакетов, - состояние интерфейса (UP/DOWN).

Примеры: - `ifconfig` — вывести сведения обо всех активных интерфейсах.

- `ifconfig eth1` — показать информацию только об интерфейсе `eth1`.

- `ifconfig eth1 down` — отключить интерфейс.

- `ifconfig eth1 mtu 1400` — изменить MTU интерфейса.

7. Какую информацию можно получить, используя `ping`? Приведите примеры.

Утилита `ping` используется для проверки доступности узла и диагностики сетевой задержки. Она показывает: - время отклика до узла (latency), - потерю пакетов, - IP-адрес удалённого хоста, - работу маршрутизации.

Примеры: - `ping 8.8.8.8` — проверить доступность DNS Google.

- `ping -c 4 192.168.1.1` — отправить 4 ICMP-пакета.

- `ping -s 200 192.168.1.1` — отправить пакеты размером 200 байт.

- `ping -i 0.2 host` — посылать пакеты каждые 0.2 секунды.