

Лабораторная работа №1

Администрирование сетевых подсистем

Машков Илья Евгеньевич

Содержание

1 Цель работы	5
2 Задание	6
3 Выполнение лабораторной работы	7
3.1 Установка DNS-сервера	7
3.2 Конфигурирование кэширующего DNS-сервера	8
3.2.1 Конфигурирование кэширующего DNS-сервера при отсутствии фильтрации DNS-запросов маршрутизаторами	8
3.2.2 Конфигурирование кэширующего DNS-сервера при наличии фильтрации DNS-запросов маршрутизаторами	13
3.2.3 Конфигурирование первичного DNS-сервера	14
3.2.4 Анализ работы DNS-сервера	19
3.3 Внесение изменений в настройки внутреннего окружения виртуальной машины	21
4 Выводы	24
Список литературы	25

Список иллюстраций

3.1 Установка bind и bind-utils	7
3.2 Обращение к dns-адресу Яндекса	8
3.3 resolv.conf	9
3.4 named.conf	9
3.5 named.ca	9
3.6 named.localhost и named.loopback	10
3.7 Запуск сервера и два запроса к адресу Яндекса	11
3.8 Установка dns-сервера, как основного для хоста	12
3.9 Перезапуск сети и проверка изменений	12
3.10 Настройка направления для DNS-запросов	13
3.11 Внесение изменений в настройки межсетевого экрана	13
3.12 Добавление перенаправления для DNS-запросов	14
3.13 Отключение DNSSEC	14
3.14 Копирование шаблона DNS-зон	14
3.15 Внесение изменений в файл named.conf	15
3.16 Замена зон в user.net	15
3.17 Создание директорий fz и rz	15
3.18 Копирование и перенос прмых и обратной зоны	16
3.19 Измнения в файле user.net	16
3.20 Изменения в файле обратной зоны	17
3.21 Изменения прав доступа	17
3.22 Восстановление меток и проверка корректности	17
3.23 Выдача разрешения на запись в файлы dns-зоны	17
3.24 Запуск лога	18
3.25 Лог после перезапуска DNS-сервера	19
3.26 Описание DNS-зоны ns.user.net	20
3.27 Использование утилиты Host	20
3.28 Внесение изменений в настройки внутреннего окружения виртуальной машины	21
3.29 dns.sh	22
3.30 Внесение изменений в Vagrantfile	23

Список таблиц

1 Цель работы

Приобретение практических навыков по установке и конфигурированию DNS-сервера, усвоение принципов работы системы доменных имён.

2 Задание

1. Установите на виртуальной машине server DNS-сервер bind и bind-utils.
2. Сконфигурируйте на виртуальной машине server кэширующий DNS-сервер.
3. Сконфигурируйте на виртуальной машине server первичный DNS-сервер.
4. При помощи утилит dig и host проанализируйте работу DNS-сервера.
5. Напишите скрипт для Vagrant, фиксирующий действия по установке и конфигурированию DNS-сервера во внутреннем окружении виртуальной машины server. Соответствующим образом внесите изменения в Vagrantfile.

3 Выполнение лабораторной работы

3.1 Установка DNS-сервера

Для начала перехожу в режим суперпользователя, а затем устанавливаю bind и bind-utils (рис. [3.1]).

```
[user@server ~]$ 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 user:  
[root@server ~]# dnf -y install bind bind-utils  
Extra Packages for Enterprise Linux 9 - x86_64 119 kB/s | 34 kB     00:00
```

Рис. 3.1: Установка bind и bind-utils

Затем осуществляю запрос к DNS-адресу Яндекса (рис. [3.2]).

```
[root@server ~]# dig www.yandex.ru

; <>> DiG 9.16.23-RH <>> www.yandex.ru
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 61969
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;www.yandex.ru.           IN      A

;; ANSWER SECTION:
www.yandex.ru.        3600    IN      A      77.88.55.88
www.yandex.ru.        3600    IN      A      5.255.255.77
www.yandex.ru.        3600    IN      A      77.88.44.55

;; Query time: 12 msec
;; SERVER: 10.0.2.3#53(10.0.2.3)
;; WHEN: Sat Nov 08 13:15:55 UTC 2025
;; MSG SIZE  rcvd: 79
```

Рис. 3.2: Обращение к dns-адресу Яндекса

Получаем вывод, который я сейчас поясню по секциям:

- **HEADER** – показывает версию dig и опции, которые используются с командой;
- **QUESTION SECTION** – выводит наш запрос (A - запрос адреса);
- **ANSWER SECTION** – 3600 - время жизни запроса и адреса, с которых пришёл ответ;
- В конце нам выводят общую информацию: время выполнения запроса, имя запрашиваемого DNS-сервера, дата и размер сообщения.

3.2 Конфигурирование кэширующего DNS-сервера

3.2.1 Конфигурирование кэширующего DNS-сервера при отсутствии фильтрации DNS-запросов маршрутизаторами

Затем мы обращаемся к таким конфигурационным файлам, как: resolv.conf (рис. [3.3]), named.conf (рис. [3.4]), named.ca (рис. [3.5]), named.localhost и named.loopback (рис. [3.6]).

```
[root@server ~]# cat /etc/resolv.conf
# Generated by NetworkManager
nameserver 10.0.2.3
```

Рис. 3.3: resolv.conf

```
[root@server ~]# cat /etc/named.conf
//
// named.conf
//
// Provided by Red Hat bind package to configure the ISC BIND named(8) DNS
// server as a caching only nameserver (as a localhost DNS resolver only).
//
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//

options {
    listen-on port 53 { 127.0.0.1; };
    listen-on-v6 port 53 { ::1; };
    directory      "/var/named";
    dump-file      "/var/named/data/cache_dump.db";
    // ...
}
```

Рис. 3.4: named.conf

```
[root@server ~]# cat /var/named/named.ca
; <>> DiG 9.18.20 <>> -4 +tcp +nored +nostats @d.root-servers.net
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 47286
;; flags: qr aa; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 27

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1450
;; QUESTION SECTION:
;.           IN      NS

;; ANSWER SECTION:
.          518400  IN      NS      a.root-servers.net.
.          518400  IN      NS      b.root-servers.net.
.          518400  IN      NS      c.root-servers.net.
.          518400  IN      NS      d.root-servers.net.
.          518400  IN      NS      e.root-servers.net.
```

Рис. 3.5: named.ca

```
[root@server ~]# cat /var/named/named.localhost
$TTL 1D
@ IN SOA @ rname.invalid. (
          0 ; serial
          1D ; refresh
          1H ; retry
          1W ; expire
          3H ) ; minimum
NS      @
A       127.0.0.1
AAAA    ::1
[root@server ~]# cat /var/named/named.loopback
$TTL 1D
@ IN SOA @ rname.invalid. (
          0 ; serial
          1D ; refresh
          1H ; retry
          1W ; expire
          3H ) ; minimum
NS      @
A       127.0.0.1
AAAA    ::1
PTR    localhost.
```

Рис. 3.6: named.localhost и named.loopback

И если в выводе (рис. [3.3]), (рис. [3.5]), (рис. [3.6]) мы видим всё то, что видели при обращении к dns-адресу Яндекса, то вывод (рис. [3.4]) стоит рассмотреть более подробно.

named.conf является главным конфигурационным файлом для DNS сервера bind и определяет его глобальное поведение и структуру зон. Также файл указывает каталоги (по типу /var/named) для хранения файлов зон и кэша, включает механизмы безопасности **dnssec**, а также определяет или подключает все необходимые зоны (корневую и пользовательскую) для обработки прямых и обратных DNS-запросов.

-**OPTIONS** - отвечает за настройку кэширующего рекурсивного сервера и разрешает принимать запросы от локальной сети, выполнять рекурсию и пересыпать неразрешённые запросы вышестоящему серверу.

Затем включаю автозапуск DNS-сервера при запуске системы и запускаю сам сервер. Также произвожу два запроса по адресам Яндеса: первый ровно такой же, что и до этого, а второй - @127.0.0.1 www.yandex.ru (рис. [3.7]).

```

[root@server ~]# systemctl enable named
Created symlink /etc/systemd/system/multi-user.target.wants/named.service → /usr
/lib/systemd/system/named.service.
[root@server ~]# dig www.yandex.ru

; <>> DiG 9.16.23-RH <>> www.yandex.ru
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 50173
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;www.yandex.ru.           IN      A

;; ANSWER SECTION:
www.yandex.ru.        3600    IN      A      5.255.255.77
www.yandex.ru.        3600    IN      A      77.88.44.55
www.yandex.ru.        3600    IN      A      77.88.55.88

;; Query time: 16 msec
;; SERVER: 10.0.2.3#53(10.0.2.3)
;; WHEN: Sat Nov  8 13:26:38 UTC 2025
;; MSG SIZE  rcvd: 79

[root@server ~]# dig @127.0.0.1 www.yandex.ru

; <>> DiG 9.16.23-RH <>> @127.0.0.1 www.yandex.ru
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 14426
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 57939df448686d4a01000000690f452d564972b6c0eafbb7 (good)
;; QUESTION SECTION:
;www.yandex.ru.           IN      A

;; ANSWER SECTION:
www.yandex.ru.        600    IN      A      5.255.255.77
www.yandex.ru.        600    IN      A      77.88.44.55
www.yandex.ru.        600    IN      A      77.88.55.88

;; Query time: 737 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sat Nov  8 13:27:09 UTC 2025
;; MSG SIZE  rcvd: 118

[root@server ~]#

```

Рис. 3.7: Запуск сервера и два запроса к адресу Яндекса

Главное отличие этих запросов состоит в следующем:

- Во втором случае был показан udf, информация о куки;
- Время жизни запроса стало меньше(3600 в первом случае и 600 во втором);
- Время выполнения запроса(16 мсек в первом случае и 737 мсек во втором), имя сервера(127.0.0.1) и размер сообщения(79 в первом случае и 118 во втором)

Делаю dns-сервер сервером по умолчанию для хоста, для этого меняю адрес на 127.0.0.1(рис. [3.8]).

```
[root@server ~]# nmcli connection edit eth0
===[ nmcli interactive connection editor ]==

Editing existing '802-3-ethernet' connection: 'eth0'

Type 'help' or '?' for available commands.
Type 'print' to show all the connection properties.
Type 'describe [<setting>.<prop>]' for detailed property description.

You may edit the following settings: connection, 802-3-ethernet (ethernet), 802-1x, dcb, sriov,
ethtool, match, ipv4, ipv6, hostname, link, tc, proxy
nmcli> remove ipv4.dns
nmcli> set ipv4.ignore-auto-dns yes
nmcli> set ipv4.dns 127.0.0.1
nmcli> save
Connection 'eth0' (52b757e4-3833-4753-b0eb-de0af7f26f57) successfully updated.
nmcli> quit
[root@server ~]# nmcli connection edit System\ eth0
===[ nmcli interactive connection editor ]==

Editing existing '802-3-ethernet' connection: 'System eth0'

Type 'help' or '?' for available commands.
Type 'print' to show all the connection properties.
Type 'describe [<setting>.<prop>]' for detailed property description.

You may edit the following settings: connection, 802-3-ethernet (ethernet), 802-1x, dcb, sriov,
ethtool, match, ipv4, ipv6, hostname, link, tc, proxy
nmcli> ^C
nmcli> remove ipv4.dns
nmcli> set ipv4.ignore-auto-dns yes
nmcli> set ipv4.dns 127.0.0.1
nmcli> save
Connection 'System eth0' (5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03) successfully updated.
nmcli> quit
[root@server ~]#
```

Рис. 3.8: Установка dns-сервера, как основного для хоста

Затем перезапускаю NetworkManager и проверяю изменения в файле resolv.conf (рис. [3.9]).

```
[root@server ~]# systemctl restart NetworkManager
[root@server ~]# cat /etc/resolv.conf
# Generated by NetworkManager
nameserver 127.0.0.1
[root@server ~]#
```

Рис. 3.9: Перезапуск сети и проверка изменений

Настраиваю направление DNS-запросов от всех узлов внутренней сети и запросы узла server через узел server. Делаю я это посредством изменений в файле named.conf (рис. [3.10]).

```

options {
    listen-on port 53 { 127.0.0.1; any; };
    listen-on-v6 port 53 { ::1; };
    directory "/var/named";
    dump-file "/var/named/data/cache_dump.db";
    statistics-file "/var/named/data/named_stats.txt";
    memstatistics-file "/var/named/data/named_mem_stats.txt";
    secroots-file "/var/named/data/named.secroots";
    recursing-file "/var/named/data/named.recurising";
    allow-query { localhost; 192.168.0.0/16; };
}

```

Рис. 3.10: Настройка направления для DNS-запросов

Вношу изменения в настройки межсетевого экрана узла server, и тем самым разрешаю ему работу с DNS. А также проверяю, что все запросы идут через заданный узел (рис. [3.11]).

```

[root@server etc]# nano /etc/named.conf
[root@server etc]# firewall-cmd --add-service=dns
success
[root@server etc]# firewall-cmd --add-service=dns --permanent
success
[root@server etc]# lsof | grep UDP
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1001/gvfs
      Output information may be incomplete.
avahi-dae 605          avahi1  12u    IPv4          19305    0t0      UDP *:mdns
avahi-dae 605          avahi1  13u    IPv6          19306    0t0      UDP *:mdns
avahi-dae 605          avahi1  14u    IPv4          19307    0t0      UDP *:56895
avahi-dae 605          avahi1  15u    IPv6          19308    0t0      UDP *:40579
chronyd  615           chrony   5u    IPv4          19161    0t0      UDP localhost:323
chronyd  615           chrony   6u    IPv6          19162    0t0      UDP localhost:323
named   6996          named   32u    IPv4          36777    0t0      UDP localhost:domain
named   6996          named   33u    IPv4          36778    0t0      UDP localhost:domain
named   6996          named   38u    IPv6          36781    0t0      UDP localhost:domain
named   6996          named   39u    IPv6          36782    0t0      UDP localhost:domain
named   6996 6997 isc-net-0  named   32u    IPv4          36777    0t0      UDP localhost:domain
named   6996 6997 isc-net-0  named   33u    IPv4          36778    0t0      UDP localhost:domain
named   6996 6997 isc-net-0  named   38u    IPv6          36781    0t0      UDP localhost:domain
named   6996 6998 isc-net-0  named   39u    IPv6          36782    0t0      UDP localhost:domain
named   6996 6998 isc-net-0  named   32u    IPv4          36777    0t0      UDP localhost:domain
named   6996 6998 isc-net-0  named   33u    IPv4          36778    0t0      UDP localhost:domain
named   6996 6998 isc-net-0  named   38u    IPv6          36781    0t0      UDP localhost:domain
named   6996 6999 isc-net-0  named   39u    IPv6          36782    0t0      UDP localhost:domain
named   6996 6999 isc-net-0  named   32u    IPv4          36777    0t0      UDP localhost:domain
named   6996 6999 isc-net-0  named   33u    IPv4          36778    0t0      UDP localhost:domain
named   6996 6999 isc-net-0  named   38u    IPv6          36781    0t0      UDP localhost:domain
named   6996 7000 isc-net-0  named   39u    IPv6          36782    0t0      UDP localhost:domain
named   6996 7000 isc-net-0  named   32u    IPv4          36777    0t0      UDP localhost:domain
named   6996 7000 isc-net-0  named   33u    IPv4          36778    0t0      UDP localhost:domain
named   6996 7000 isc-net-0  named   38u    IPv6          36781    0t0      UDP localhost:domain
named   6996 7001 isc-timer  named   32u    IPv4          36777    0t0      UDP localhost:domain
named   6996 7001 isc-timer  named   33u    IPv4          36778    0t0      UDP localhost:domain
named   6996 7001 isc-timer  named   38u    IPv6          36781    0t0      UDP localhost:domain
named   6996 7001 isc-timer  named   39u    IPv6          36782    0t0      UDP localhost:domain
named   6996 7002 isc-socke  named   32u    IPv4          36777    0t0      UDP localhost:domain
named   6996 7002 isc-socke  named   33u    IPv4          36778    0t0      UDP localhost:domain
named   6996 7002 isc-socke  named   38u    IPv6          36781    0t0      UDP localhost:domain
named   6996 7002 isc-socke  named   39u    IPv6          36782    0t0      UDP localhost:domain
named   6996 7003 isc-socke  named   32u    IPv4          36777    0t0      UDP localhost:domain
named   6996 7003 isc-socke  named   33u    IPv4          36778    0t0      UDP localhost:domain

```

Рис. 3.11: Внесение изменений в настройки межсетевого экрана

3.2.2 Конфигурирование кэширующего DNS-сервера при наличии фильтрации DNS-запросов маршрутизаторами

Добавляю перенаправление DNS-запросов на конкретный вышестоящий сервер посредством изменений в файле named.conf. Если говорить конкретнее, то я добавляю строчки **forwardes** и **forward first**.(рис. [3.12]).

```
options {
    listen-on port 53 { 127.0.0.1; any; };
    listen-on-v6 port 53 { ::1; };
    directory    "/var/named";
    dump-file    "/var/named/data/cache_dump.db";
    statistics-file "/var/named/data/named_stats.txt";
    memstatistics-file "/var/named/data/named_mem_stats.txt";
    secroots-file  "/var/named/data/named.secroots";
    recursing-file "/var/named/data/named.recurse";
    allow-query   { localhost; 192.168.0.0/16; };
    forwardes { 192.168.1.1; };
    forward first;
```

Рис. 3.12: Добавление перенаправления для DNS-запросов

Также указываю настройки, касающиеся технологии DNSSEC. Мы их выключаем, т.к. вышестоящий сервер может их не поддерживать (рис. [3.13]).

```
    reduce local attack surface
*/
recursion yes;

dnssec-enable no;
dnssec-validation no;
```

Рис. 3.13: Отключение DNSSEC

3.2.3 Конфигурирование первичного DNS-сервера

Копирую шаблон описания DNS-зон в другой каталог и переименовываю его на **user.net** (рис. [3.14]).

```
[root@server etc]# nano /etc/named.conf
[root@server etc]# nano /etc/named.conf
[root@server etc]# nano /etc/named.conf
[root@server etc]# cp /etc/named.rfc1912.zones /etc/named/
[root@server etc]# cd /etc/named
[root@server named]# mv /etc/named/named.rfc1912.zones /etc/named/user.net
[root@server named]# ls
user.net
[root@server named]#
```

Рис. 3.14: Копирование шаблона DNS-зон

Затем включаю файл описания зоны user.net в файле named.conf (рис. [3.15]).

```
include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
include "/etc/named/user.net";
```

Рис. 3.15: Внесение изменений в файл named.conf

В файле user.net меняю зону **localhost.localdomain** на свою прямую зону и вместо зоны **1.0.0.127.in-addr.arpa** прописываю свою обратную зону (рис. [3.16]).

```
GNU nano 5.6.1                                     /etc/named/user.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 "user.net" IN {
    type master;
    file "master/fz/user.net";
    allow-update { none; };
};

zone "1.168.192.in-addr.arpa" IN {
    type master;
    file "master/rz/192.168.1";
    allow-update { none; };
};
```

Рис. 3.16: Замена зон в user.net

Затем создаю директории **fz** и **rz**, в которых будут храниться файлы прямой и обратной зоны (рис. [3.17]).

```
[root@server named]# cd /var/named
[root@server named]# mkdir -p /var/named/master/fz
[root@server named]# mkdir -p /var/named/master/rz
[root@server named]#
```

Рис. 3.17: Создание директорий fz и rz

Копирую шаблоны прямой и обратной зоны в соответствующие директории, а также переношу шаблон прямой зоны **named.localhost** и переименовываю его в **user.net** (рис. [3.18]).

```
[root@server named]# cp /var/named/named.localhost /var/named/master/fz/
[root@server named]# cd /var/named/master/fz/
[root@server fz]# mv named.localhost user.net
[root@server fz]#
```

Рис. 3.18: Копирование и перенос прмых и обратной зоны

Затем изменяю файл **user.net**, указав необходимые dns-записи для прямой зоны и пр. (рис. [3.19]).

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

Рис. 3.19: Измнения в файле user.net

Также изменяю файл шаблон обратной зоны похожим образом, переименовываю его на “**192.168.1**” и переношу в соответствующую директорию (рис. [3.20]).

```

GNU nano 5.6.1
$TTL 1D
@       IN SOA  @ server.user.net. (
                                192.168.1
                                2023072700      ; serial
                                1D              ; refresh
                                1H              ; retry
                                1W              ; expire
                                3H )            ; minimum
NS      @
A       192.168.1.1
PTR    server.user.net.
$ORIGIN 1.168.192.in-addr.arpa.
1      PTR    server.user.net.
1      PTR    ns.user.net

```

Рис. 3.20: Изменения в файле обратной зоны

Далее меняю права доступа так, чтобы named мог с ними работать (рис. [3.21]).

```

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

```

Рис. 3.21: Изменения прав доступа

Т.к. мы внесли изменения доступа к конфигурационным файлам, нам нужно корректно восстановить метки к файлам named в SELinux. И проверяем их состояние (рис. [3.22]).

```

[root@server ~]# restorecon -vR /etc
Relabeled '/etc/sysconfig/network-scripts/ifcfg-eth1' from unconfined_u:object_r:user_t:s0 to unconfined_u:object_r:net_
conf_t:s0
[root@server ~]# restorecon -vR /var/named
[root@server ~]# getsebool -a | grep named
named_tcp_bind_http_port --> off
named_write_master_zones --> on
[root@server ~]#

```

Рис. 3.22: Восстановление меток и проверка корректности

Также даю разрешение на запись в файлы dns-зоны (рис. [3.23]).

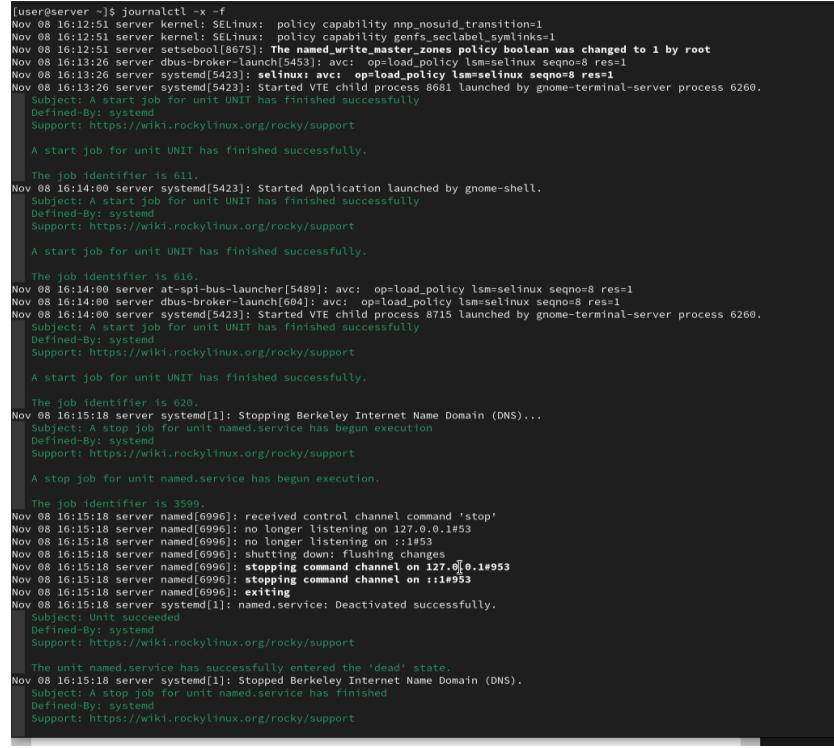
```

[root@server ~]# setsebool named_write_master_zones 1
[root@server ~]# setsebool -P named_write_master_zones 1
[root@server ~]#

```

Рис. 3.23: Выдача разрешения на запись в файлы dns-зоны

Запускаю расширенный лог системных сообщений в реальном времени, чтобы проверить корректность работы системы (рис. [3.24]).



```
[user@server ~]$ journalctl -x -f
Nov 08 16:12:51 server kernel: SELinux: policy capability np_nosuid_transition=1
Nov 08 16:12:51 server kernel: SELinux: policy capability genfs_seclabel_symlinks=1
Nov 08 16:12:51 server setselinux[8675]: The named_write_master_zones policy boolean was changed to 1 by root
Nov 08 16:13:26 server dbus-broker-launch[5453]: avc: op=load_policy lsm=selinux seqno=8 res=1
Nov 08 16:13:26 server systemd[5423]: selinux avc: op=load_policy lsm=selinux seqno=8 res=1
Nov 08 16:13:26 server systemd[5423]: Started VTE child process 8681 launched by gnome-terminal-server process 6260.
        Subject: A start job for unit UNIT has finished successfully
        Defined-By: systemd
        Support: https://wiki.rockylinux.org/rocky/support
        A start job for unit UNIT has finished successfully.

        The job identifier is 611.
Nov 08 16:14:00 server systemd[5423]: Started Application launched by gnome-shell.
        Subject: A start job for unit UNIT has finished successfully
        Defined-By: systemd
        Support: https://wiki.rockylinux.org/rocky/support
        A start job for unit UNIT has finished successfully.

        The job identifier is 612.
Nov 08 16:14:00 server at-spi-bus-launcher[5489]: avc: op=load_policy lsm=selinux seqno=8 res=1
Nov 08 16:14:00 server dbus-broker-launch[604]: avc: op=load_policy lsm=selinux seqno=8 res=1
Nov 08 16:14:00 server systemd[5423]: Started VTE child process 8715 launched by gnome-terminal-server process 6260.
        Subject: A start job for unit UNIT has finished successfully
        Defined-By: systemd
        Support: https://wiki.rockylinux.org/rocky/support
        A start job for unit UNIT has finished successfully.

        The job identifier is 620.
Nov 08 16:15:18 server systemd[1]: Stopping Berkeley Internet Name Domain (DNS)...
        Subject: A stop job for unit named.service has begun execution
        Defined-By: systemd
        Support: https://wiki.rockylinux.org/rocky/support
        A stop job for unit named.service has begun execution.

        The job identifier is 3590.
Nov 08 16:15:18 server named[6996]: received control channel command 'stop'
Nov 08 16:15:18 server named[6996]: no longer listening on 127.0.0.1#953
Nov 08 16:15:18 server named[6996]: no longer listening on ::1#953
Nov 08 16:15:18 server named[6996]: shutting down: flushing changes
Nov 08 16:15:18 server named[6996]: stopping command channel on 127.0.0.1#953
Nov 08 16:15:18 server named[6996]: stopping command channel on ::1#953
Nov 08 16:15:18 server named[6996]: exiting
Nov 08 16:15:18 server systemd[1]: named.service: Deactivated successfully.
        Subject: Unit succeeded
        Defined-By: systemd
        Support: https://wiki.rockylinux.org/rocky/support
        The unit named.service has successfully entered the 'dead' state.
Nov 08 16:15:18 server systemd[1]: Stopped Berkeley Internet Name Domain (DNS).
        Subject: A stop job for unit named.service has finished
        Defined-By: systemd
        Support: https://wiki.rockylinux.org/rocky/support
```

Рис. 3.24: Запуск лога

Затем перезапускаю DNS-сервер и смотрю, что в логе не выскоцило ни одной ошибки, а значит, что всё работает корректно (рис. [3.25]).

Рис. 3.25: Лог после перезапуска DNS-сервера

3.2.4 Анализ работы DNS-сервера

Отправляю запрос на описание DNS-зоны с сервера ns.user.net (рис. [3.26]).

```
[root@server rz]# dig ns.user.net
; <>> DiG 9.16.23-RH <>> ns.user.net
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 61919
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;;
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1232
;; COOKIE: a72f90288f3a5de601000000690f71125f2140c51514e9b9 (good)
;; QUESTION SECTION:
;ns.user.net.           IN      A
;;
;; ANSWER SECTION:
ns.user.net.        86400   IN      A      192.168.1.1
;;
;; Query time: 1 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sat Nov 08 16:34:26 UTC 2025
;; MSG SIZE  rcvd: 84
[root@server rz]# █
```

Рис. 3.26: Описание DNS-зоны ns.user.net

В Выводе видим всё то же самое, что и при запросах к адресам Яндекса. Также мы видим, что обращение приходит на сервер 127.0.0.1, а ответ приходит на 192.168.1.1, что подтверждает корректность всех проделанных действий.

С помощью утилиты host проверяем корректность работы DNS-сервера и видим, что все корректно работает (рис. [3.27]).

```
[root@server rz]# host -l user.net
user.net name server user.net.
user.net has address 192.168.1.1
ns.user.net has address 192.168.1.1
server.user.net has address 192.168.1.1
[root@server rz]# host -a user.net
Trying "user.net"
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 8364
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0
;;
;; QUESTION SECTION:
;user.net.           IN      ANY
;;
;; ANSWER SECTION:
user.net.        86400   IN      SOA     user.net. server.user.net. 2024072700 86400 3600 604800 10800
user.net.        86400   IN      NS      user.net.
user.net.        86400   IN      A       192.168.1.1
Received 99 bytes from 127.0.0.1#53 in 6 ms
[root@server rz]# host -t A user.net
user.net has address 192.168.1.1
[root@server rz]# host -t PTR 192.168.1.1
1.1.168.192.in-addr.arpa domain name pointer server.user.net.
1.1.168.192.in-addr.arpa domain name pointer ns.user.net.1.168.192.in-addr.arpa.
[root@server rz]# █
```

Рис. 3.27: Использование утилиты Host

3.3 Внесение изменений в настройки внутреннего окружения виртуальной машины

Создаю каталог **dns** во внутреннем окружении виртуальной машины (рис. [3.28]).

```
[root@server rz]# cd .vagrant
[root@server vagrant]# ls
[root@server vagrant]# ls -l
total 0
[root@server vagrant]# mkdir -p /vagrant/provision/server/dns/etc/named
[root@server vagrant]# mkdir -p /vagrant/provision/server/dns/var/named/master/
[root@server vagrant]# cp -R /etc/named.conf /vagrant/provision/server/dns/etc/
[root@server vagrant]# cp -R /etc/named/* /vagrant/provision/server/dns/etc/named/
[root@server vagrant]# cp -R /var/named/master/* /vagrant/provision/server/dns/var/named/master/
[root@server vagrant]#
```

Рис. 3.28: Внесение изменений в настройки внутреннего окружения виртуальной машины

Создаю скрипт “**dns.sh**”, который будет выполнять все действия выполненные в этой лабораторной работе при запуске системы (рис. [3.29]).

```
GNU nano 5.6.1                                         dns.sh
#!/bin/bash

echo "Provisioning script $0"

echo "Install needed packages"
dnf -y install bind bind-utils

echo "Copy configuration files"
cp -R /vagrant/provision/server/dns/etc/* /etc
cp -R /vagrant/provision/server/dns/var/named/* /var/named

chown -R named:named /etc/named
chown -R named:named /var/named

restorecon -vR /etc
restorecon -vR /var/named

echo "Configure firewall"
firewall-cmd --add-service=dns
firewall-cmd --add-service=dns --permanent

echo "Tuning SELinux"
setsebool named_write_master_zones 1
setsebool -P named_write_master_zones 1

echo "Change dns server address"
nmcli connection edit "System eth0" <<EOF
remove ipv4.dns
set ipv4.ignore-auto-dns yes
set ipv4.dns 127.0.0.1
save
quit
EOF
systemctl restart NetworkManager

echo "Start named service"
systemctl enable named
systemctl start named
```

Рис. 3.29: dns.sh

Затем вношу изменения в файл Vagrantfile, чтобы скрипт отрабатывался при каждом запуске системы (рис. [3.30]).

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

  server.vm.boot_timeout = 1440

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

  server.vm.network :private_network,
    ip: "192.168.1.1",
    virtualbox_intnet: true

  server.vm.provision "server dummy",
    type: "shell",
    preserve_order: true,
    path: "provision/server/01-dummy.sh"
  server.vm.provision "server dns",
    type: "shell",
    preserve_order: true,
    path: "provision/server/dns.sh"
  server.vm.provider :virtualbox do |v|
    v.linked_clone = true
  # Customize the amount of memory on the VM
  # v.memory = 8192
  # v.cpus = 2
  # v.name = "server"
# Display the VirtualBox GUI when booting the machine
```

Рис. 3.30: Внесение изменений в Vagrantfile

4 Выводы

Во время выполнения данной лабораторной работы я приобрёл практические навыки по установке и конфигурированию DNS-сервера и усвоил принципы работы системы доменных имён.

Список литературы

Администрирование сетевых подсистем