

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2

дисциплина: Администрирование сетевых подсистем

Студент: Кармацкий Н.С.

Группа: НФИбд-01-21

МОСКВА

2023 г.

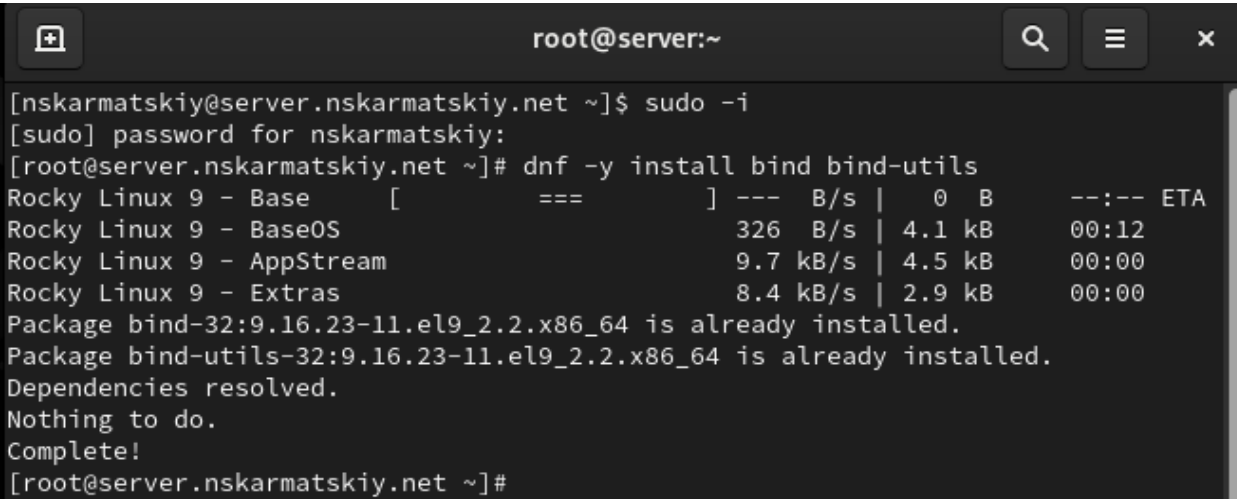
Постановка задачи

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

Выполнение работы

1. Установка на виртуальной машине server DNS-сервер bind и bind-utils

1. Запускаем виртуальную машину server. Переходим в режим суперпользователя и устанавливаем bind и bind-utils.



```
root@server:~  
[nskarmatskiy@server.nskarmatskiy.net ~]$ sudo -i  
[sudo] password for nskarmatskiy:  
[root@server.nskarmatskiy.net ~]# dnf -y install bind bind-utils  
Rocky Linux 9 - Base [====] --- B/s | 0 B --:-- ETA  
Rocky Linux 9 - BaseOS 326 B/s | 4.1 kB 00:12  
Rocky Linux 9 - AppStream 9.7 kB/s | 4.5 kB 00:00  
Rocky Linux 9 - Extras 8.4 kB/s | 2.9 kB 00:00  
Package bind-32:9.16.23-11.el9_2.2.x86_64 is already installed.  
Package bind-utils-32:9.16.23-11.el9_2.2.x86_64 is already installed.  
Dependencies resolved.  
Nothing to do.  
Complete!  
[root@server.nskarmatskiy.net ~]#
```

Рис.1.1: Режим суперпользователя и установка bind и bind-utils

2. В качестве упражнения с помощью утилиты dig сделаем запрос, например, к DNS-адресу www.yandex.ru

```
root@server:~  
[root@server.nskarmatskiy.net ~]# dig www.yandex.ru  
  
; <<>> DiG 9.16.23-RH <<>> www.yandex.ru  
;; global options: +cmd  
;; Got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2163  
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1  
  
;; OPT PSEUDOSECTION:  
; EDNS: version: 0, flags:; udp: 1232  
; COOKIE: 3d6614eaa25bb49a01000000654fb7d43388072602f1d708 (good)  
;; QUESTION SECTION:  
;www.yandex.ru.                IN      A  
  
;; ANSWER SECTION:  
www.yandex.ru.                300     IN      A      5.255.255.70  
www.yandex.ru.                300     IN      A      77.88.55.88  
www.yandex.ru.                300     IN      A      77.88.55.60  
www.yandex.ru.                300     IN      A      5.255.255.77  
  
;; Query time: 1214 msec  
;; SERVER: 127.0.0.1#53(127.0.0.1)  
;; WHEN: Sat Nov 11 17:20:20 UTC 2023  
;; MSG SIZE rcvd: 134
```

Рис.1.2: Запрос к яндексу

HEADER — отображает информацию о версии утилиты, ID запроса, полученных ошибках и использованных флагах вывода. Выводится и другая важная информация о количестве запросов, обращений к DNS-серверу и т. д.;

QUESTION SECTION — секция, которая отображает текущий запрос(www.yandex.ru);

ANSWER SECTION — секция, в которой отображается результат обработки созданного запроса (в данном случае это IP-адрес домена).

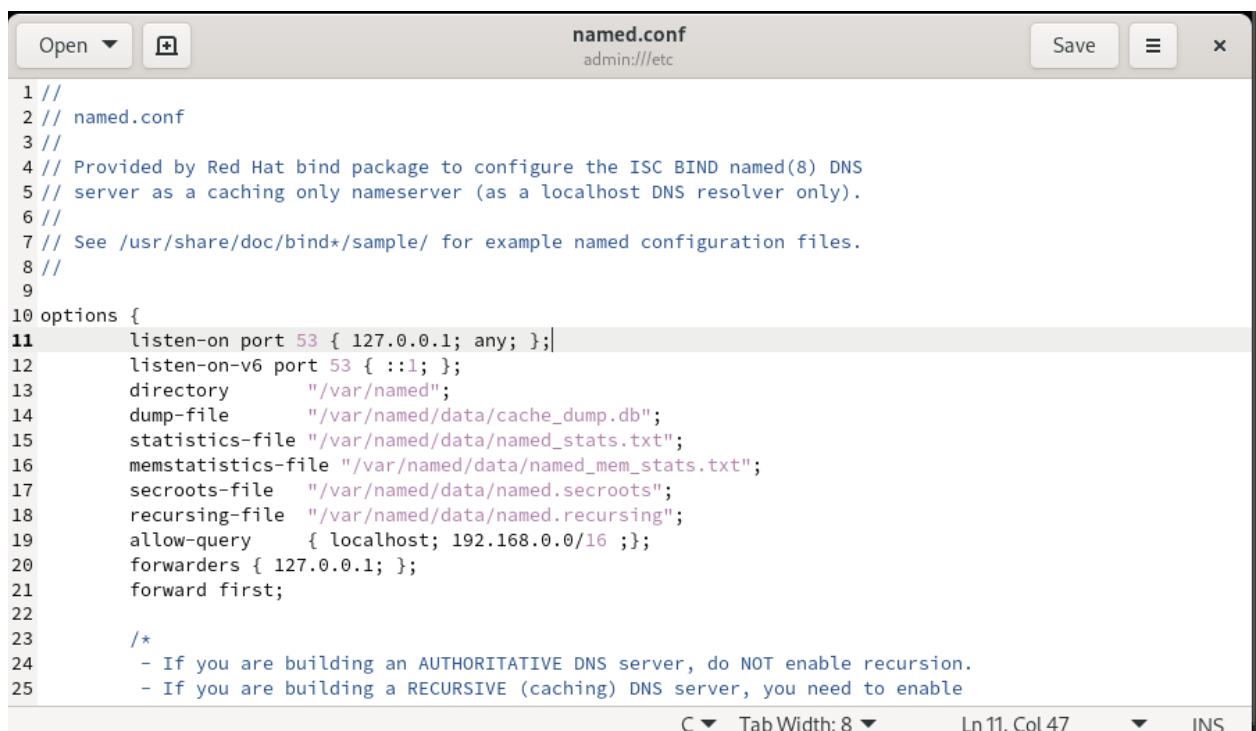
2. Конфигурирование кэширующего DNS-сервера

1. Проанализируем построчно содержание файлов /etc/resolv.conf, /etc/named.conf, /var/named/named.ca, /var/named/named.localhost, /var/named/named.loopback.

```
resolv.conf [Read-Only]  
/etc  
1 # Generated by NetworkManager  
2 search nskarmatskiy.net  
3 nameserver 127.0.0.1
```

Рис.2.1: resolv.conf

Тут отображается имя нашего сервера и его ip



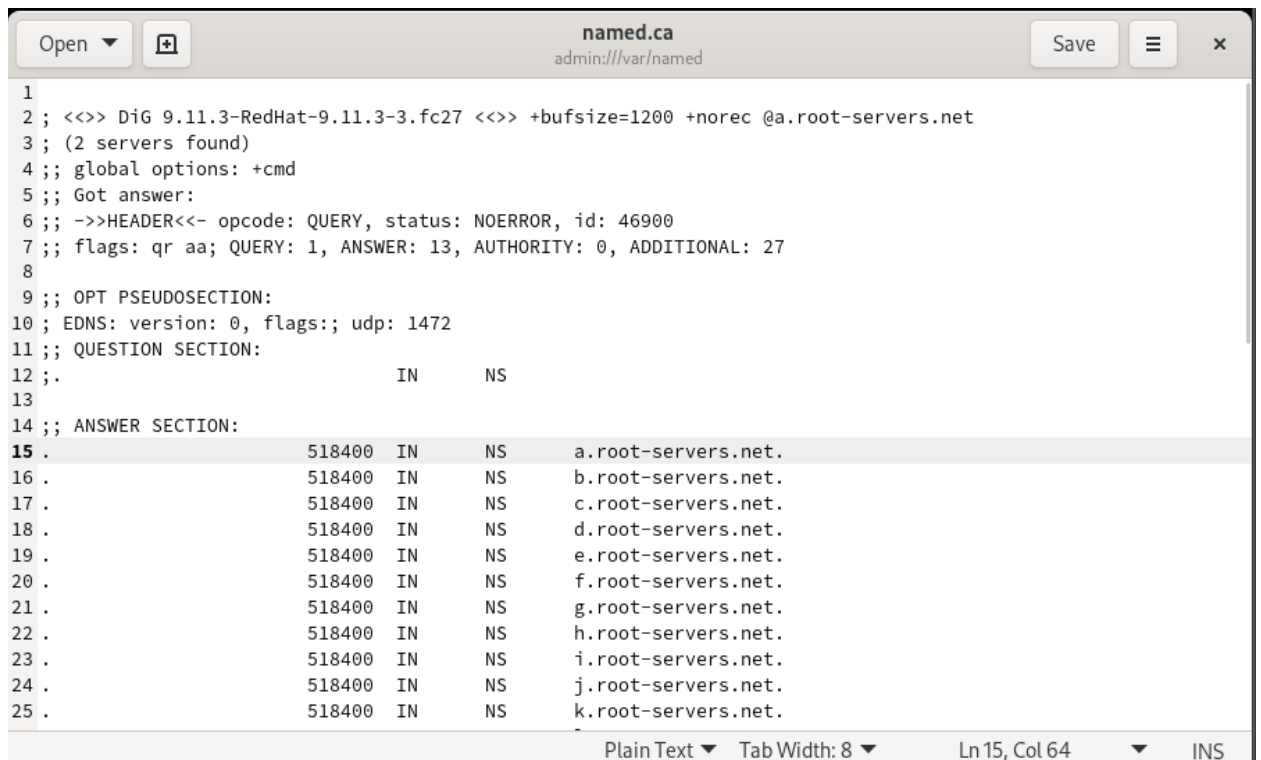
```
1 //
2 // named.conf
3 //
4 // Provided by Red Hat bind package to configure the ISC BIND named(8) DNS
5 // server as a caching only nameserver (as a localhost DNS resolver only).
6 //
7 // See /usr/share/doc/bind*/sample/ for example named configuration files.
8 //
9
10 options {
11     listen-on port 53 { 127.0.0.1; any; };
12     listen-on-v6 port 53 { ::1; };
13     directory "/var/named";
14     dump-file "/var/named/data/cache_dump.db";
15     statistics-file "/var/named/data/named_stats.txt";
16     memstatistics-file "/var/named/data/named_mem_stats.txt";
17     secroots-file "/var/named/data/named.secroots";
18     recursing-file "/var/named/data/named.recursing";
19     allow-query { localhost; 192.168.0.0/16; };
20     forwarders { 127.0.0.1; };
21     forward first;
22
23     /*
24      - If you are building an AUTHORITATIVE DNS server, do NOT enable recursion.
25      - If you are building a RECURSIVE (caching) DNS server, you need to enable
```

Рис.2.2: named.conf

Этот код является конфигурационным файлом для сервера DNS ISC BIND (named).

1. options {: Начало блока опций, где задаются настройки для сервера.
2. listen-on port 53 { 127.0.0.1; any; };; Указывает, на каких адресах и портах сервер будет слушать запросы. Здесь указано слушать на локальном адресе 127.0.0.1 и на любом доступном адресе.
3. listen-on-v6 port 53 { ::1; };; То же самое, но для IPv6, слушает на локальном адресе ::1.
4. directory "/var/named";: Задаёт директорию, в которой хранятся файлы зоны и другие данные сервера.
5. dump-file "/var/named/data/cache_dump.db";: Указывает путь к файлу, в который будет выполняться дамп (запись) данных кэша сервера.
6. statistics-file "/var/named/data/named_stats.txt";: Указывает путь к файлу, в который будут записываться статистика сервера.
7. memstatistics-file "/var/named/data/named_mem_stats.txt";: Указывает путь к файлу, в который будут записываться статистика использования памяти сервером.
8. secroots-file "/var/named/data/named.secroots";: Путь к файлу, в котором хранятся корневые ключи для проверки DNSSEC.

9. `recursing-file "/var/named/data/named.recursing";`:: Путь к файлу, в который будут записываться данные о рекурсивных запросах.
10. `allow-query { localhost; 192.168.0.0/16 ;}`:: Указывает, каким клиентам разрешено отправлять запросы. Здесь разрешены запросы только от локального хоста и от сети 192.168.0.0/16.
11. `forwarders { 127.0.0.1; }`:: Задаёт адреса, на которые будут направляться запросы, если они не могут быть удовлетворены локально. В данном случае, запросы будут перенаправляться на 127.0.0.1.
12. `forward first;`:: Указывает серверу сначала пытаться выполнить запрос через `forwarders`, и только в случае неудачи выполнять собственный поиск.
13. `recursion yes;`:: Включает рекурсивные запросы. Это важно для DNS-серверов, предназначенных для кэширования.
14. `dnssec-validation no;`:: Отключает проверку DNSSEC.
15. `managed-keys-directory "/var/named/dynamic";`:: Указывает директорию, где будут храниться ключи для управляемых зон.
16. `geoip-directory "/usr/share/GeoIP";`:: Директория для файлов GeoIP, используемых для географической локализации IP-адресов.
17. `pid-file "/run/named/named.pid";`:: Путь к файлу, в котором будет сохранен PID процесса `named`.
18. `session-keyfile "/run/named/session.key";`:: Путь к файлу, в котором будет сохранен ключ сессии.
19. `include "/etc/crypto-policies/back-ends/bind.config";`:: Включает файл конфигурации для поддержки политики шифрования.
20. `logging { ...};`:: Начало блока настроек для логирования событий.
21. `zone "." IN { type hint; file "named.ca"; }`:: Задаёт зону для корневых DNS-серверов.
22. `include "/etc/named.rfc1912.zones";`:: Включает файл конфигурации с предопределёнными зонами, соответствующими стандартам RFC 1912.
23. `include "/etc/named.root.key";`:: Включает файл с корневыми ключами DNSSEC.
24. `include "/etc/named/nskarmatskiy.net";`:: Включает файл конфигурации для зоны `nskarmatskiy.net`.



```
1
2 ; <<>> DiG 9.11.3-RedHat-9.11.3-3.fc27 <<>> +bufsize=1200 +norec @a.root-servers.net
3 ; (2 servers found)
4 ;; global options: +cmd
5 ;; Got answer:
6 ;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46900
7 ;; flags: qr aa; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 27
8
9 ;; OPT PSEUDOSECTION:
10 ; EDNS: version: 0, flags:: udp: 1472
11 ;; QUESTION SECTION:
12 ;.                                IN      NS
13
14 ;; ANSWER SECTION:
15 .                                518400  IN      NS      a.root-servers.net.
16 .                                518400  IN      NS      b.root-servers.net.
17 .                                518400  IN      NS      c.root-servers.net.
18 .                                518400  IN      NS      d.root-servers.net.
19 .                                518400  IN      NS      e.root-servers.net.
20 .                                518400  IN      NS      f.root-servers.net.
21 .                                518400  IN      NS      g.root-servers.net.
22 .                                518400  IN      NS      h.root-servers.net.
23 .                                518400  IN      NS      i.root-servers.net.
24 .                                518400  IN      NS      j.root-servers.net.
25 .                                518400  IN      NS      k.root-servers.net.
```

Рис.2.3: named.ca

Этот код представляет собой вывод команды dig, выполняемой с использованием утилиты для DNS-запросов.

1. ; <<>> DiG 9.11.3-RedHat-9.11.3-3.fc27 <<>> +bufsize=1200 +norec @a.root-servers.net - Это заголовок, который указывает на версию DiG (версия 9.11.3) и параметры запроса, такие как размер буфера и отключение рекурсии. Запрос адресован серверу a.root-servers.net.

2. ; (2 servers found) - Это сообщение указывает на то, что было найдено 2 сервера.

3. ;; global options: +cmd - Это сообщение показывает глобальные опции, в данном случае, что используется командная строка.

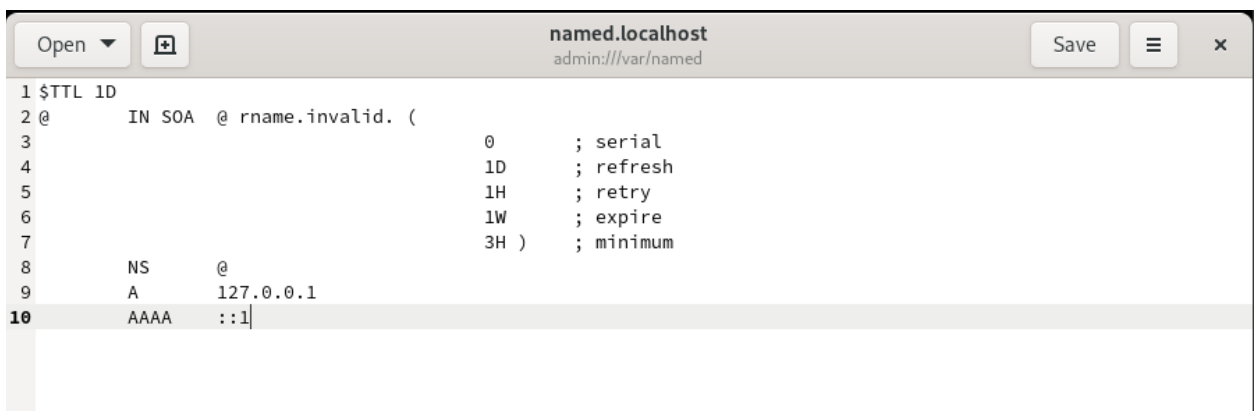
4. ;; Got answer: - Это указывает на то, что получен ответ от DNS-сервера.

5. ;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46900 - Это заголовок ответа, где указываются параметры запроса (opcode: QUERY), статус (status: NOERROR), и идентификационный номер запроса (id: 46900).

6. ;; flags: qr aa; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 27 - Это флаги ответа, где qr aa указывает на факт, что это ответ (qr) и сервер является авторитетным (aa). Далее идут счетчики запросов, ответов, авторитетных серверов и дополнительной информации.

7. ;; OPT PSEUDOSECTION: - Начало секции опций (EDNS).

8. ; EDNS: version: 0, flags:; udp: 1472 - Это параметры EDNS: версия 0, отсутствие флагов, размер UDP-пакета 1472 байта.
9. ;; QUESTION SECTION: . IN NS - Это раздел с вопросом, где запрашиваются записи NS для домена "." (корневого домена).
10. ;; ANSWER SECTION: - Начало секции с ответами.
11. . 518400 IN NS a.root-servers.net. - Запись о том, что корневой домен имеет 13 серверов и начинается перечисление их адресов.
12. ;; ADDITIONAL SECTION: - Начало секции с дополнительной информацией.
13. a.root-servers.net. 518400 IN A 198.41.0.4 - Запись с IP-адресом для сервера a.root-servers.net.
14. ;; Query time: 24 msec - Время выполнения запроса.
15. ;; SERVER: 198.41.0.4#53(198.41.0.4) - Информация о сервере, который предоставил ответ.
16. ;; WHEN: Thu Apr 05 15:57:34 CEST 2018 - Дата и время выполнения запроса.
17. ;; MSG SIZE rcvd: 811 - Размер полученного сообщения в байтах.



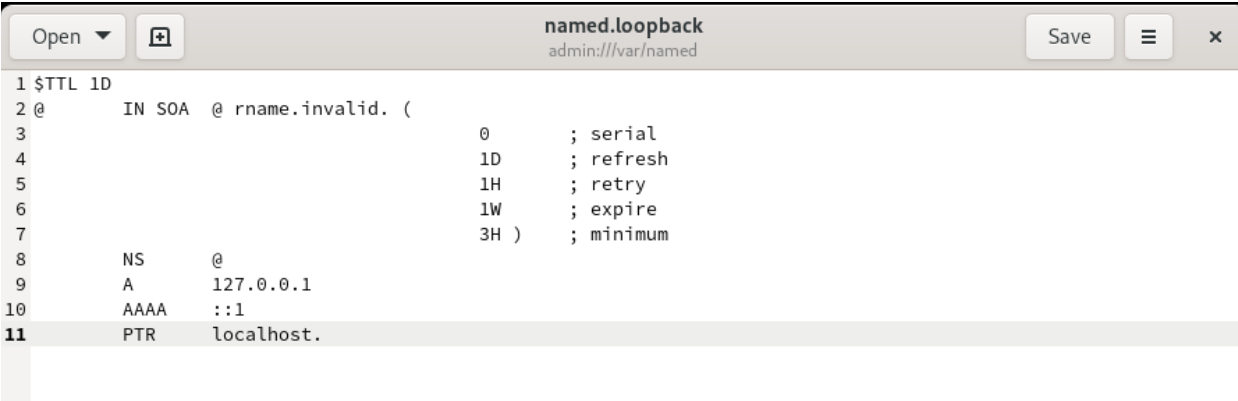
```
1 $TTL 1D
2 @      IN SOA  @ rname.invalid. (
3                               0      ; serial
4                               1D     ; refresh
5                               1H     ; retry
6                               1W     ; expire
7                               3H )   ; minimum
8      NS      @
9      A       127.0.0.1
10     AAAA    ::1
```

Рис.2.4: named.localhost

Данный код представляет собой запись в файле настройки DNS-сервера с использованием формата BIND. Вот комментарии для каждой строки:

1. `$TTL 1D`: Это указывает на время жизни (Time to Live) записей в кеше. В данном случае, 1D означает 1 день.
2. `@ IN SOA @ rname.invalid. (`: Это начало определения ресурсной записи SOA (Start of Authority). `@` означает текущую доменную зону. `rname.invalid.` - это адрес электронной почты владельца доменной зоны (в данном случае, это некорректный адрес).

3. 0 ; serial: Это номер версии (серийный номер) доменной зоны. При каждом изменении зоны, этот номер должен увеличиваться.
4. 1D ; refresh: Время, через которое другие DNS-серверы должны проверить, обновилась ли зона.
5. 1H ; retry: Время, через которое другие DNS-серверы должны повторить попытку связаться с первичным сервером в случае невозможности связи.
6. 1W ; expire: Максимальное время, в течение которого другие DNS-серверы могут использовать данные из кеша, если первичный сервер недоступен.
7. 3H) ; minimum: Минимальное время жизни записей в кеше.
8. NS @: Определение имени сервера (NS - Name Server). В данном случае, это текущая зона.
9. A 127.0.0.1: Указывает на IPv4-адрес (A - Address) для текущей зоны. В данном случае, это локальный адрес 127.0.0.1.
10. AAAA ::1: Указывает на IPv6-адрес (AAAA - IPv6 Address) для текущей зоны. В данном случае, это локальный адрес ::1 (IPv6-адрес для localhost).



```
1 $TTL 1D
2 @      IN SOA  @ rname.invalid. (
3                                     0      ; serial
4                                     1D     ; refresh
5                                     1H     ; retry
6                                     1W     ; expire
7                                     3H )   ; minimum
8
9      NS   @
10     A    127.0.0.1
11     AAAA ::1
12     PTR  localhost.
```

Рис.2.5: named.loopback

Этот код представляет собой запись в формате DNS (Domain Name System) для настройки основных параметров DNS-зоны.

1. \$TTL 1D: Это устанавливает время жизни (Time To Live) записей в зоне на 1 день. Это означает, что изменения в зоне DNS будут распространяться по всем серверам за один день.
2. @ IN SOA @ rname.invalid. (...): Это начало определения "Start of Authority" (SOA) записи для текущей DNS-зоны. @ здесь представляет собой корень

домена. `rname.invalid.` - это адрес электронной почты владельца домена. Затем следуют параметры SOA записи, такие как серийный номер, время обновления, время повтора, время истечения и минимальное время.

3. NS @: Это устанавливает имя сервера (NS) для текущей DNS-зоны. @ снова представляет собой корень домена.

4. A 127.0.0.1: Это устанавливает IPv4-адрес для текущего домена. В данном случае, это устанавливает соответствие между доменным именем и IP-адресом [127.0.0.1](#) (локальный адрес).

5. AAAA ::1: Это устанавливает IPv6-адрес для текущего домена. Здесь используется IPv6 адрес "::1", который является эквивалентом IPv4 адреса [127.0.0.1](#) и также означает локальный адрес.

6. PTR localhost.: Эта строка устанавливает обратную запись (PTR) для IP-адреса [127.0.0.1](#). Она указывает, что IP-адрес [127.0.0.1](#) соответствует хосту "localhost".

3. Запустим DNS-сервер, Так же включим запуск DNS-сервера в автозапуск при загрузке системы.

```
Locati ;; MSG SIZE rcvd: 134
[root@server.nskarmatskiy.net ~]# systemctl start named
[root@server.nskarmatskiy.net ~]# systemctl enable named
[root@server.nskarmatskiy.net ~]#
```

Рис.2.3: Запущенный DNS-сервер

4. Сделаем DNS-сервер сервером по умолчанию для хоста server и внутренней виртуальной сети. Так же перезапустим NetworkManager

```
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, 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' (eea5037a-0c32-4f77-8b4c-c886aa278923) successfully updated.
nmcli> quit
bash: remove: command not found...
bash: save: command not found...
bash: quit: command not found...
[root@server.nskarmatskiy.net ~]# systemctl restart NetworkManager
[root@server.nskarmatskiy.net ~]#
```

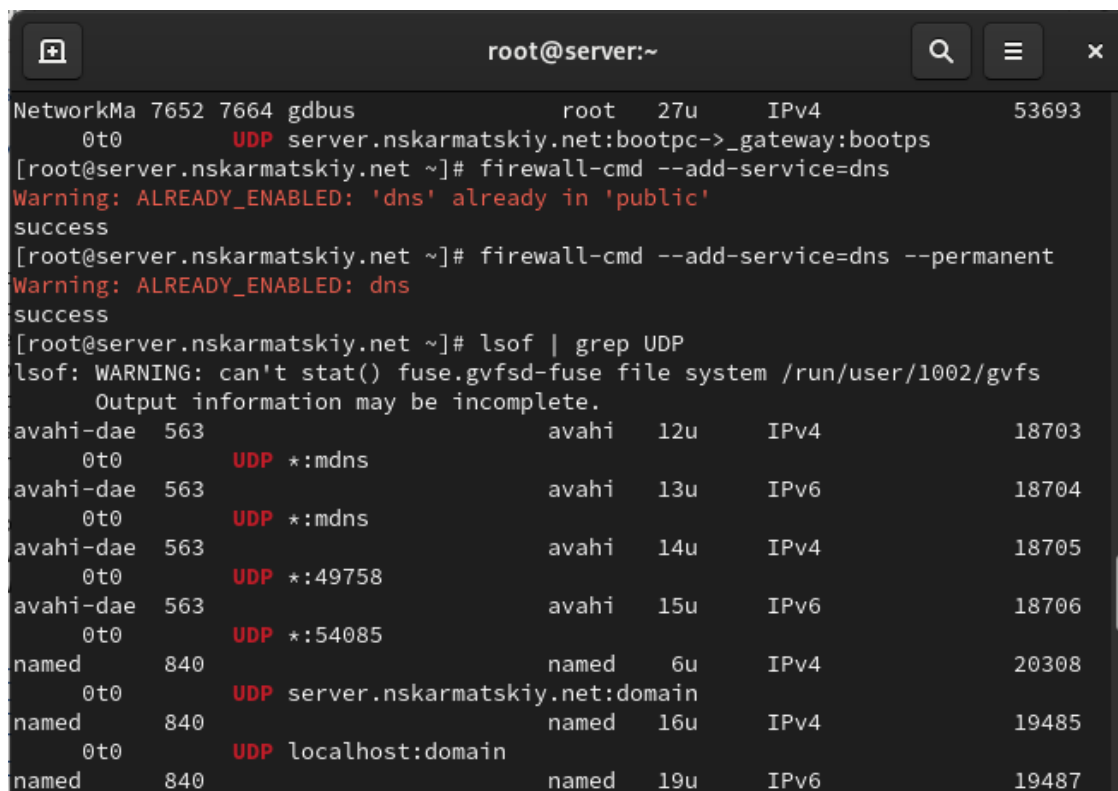
Рис.2.4: Установка DNS-сервера по умолчанию

5. Настроим направление DNS-запросов от всех узлов внутренней сети, включая запросы от узла server, через узел server.

```
8 //
9
10 options {
11     listen-on port 53 { 127.0.0.1; any; };
12     listen-on-v6 port 53 { ::1; };
13     directory      "/var/named";
14     dump-file       "/var/named/data/cache_dump.db";
15     statistics-file "/var/named/data/named_stats.txt";
16     memstatistics-file "/var/named/data/named_mem_stats.txt";
17     secroots-file   "/var/named/data/named.secroots";
18     recursing-file  "/var/named/data/named.recursing";
19     allow-query     { localhost; 192.168.0.0/16; };
20     forwarders { 127.0.0.1; };
21     forward first;
22 }
```

Рис.2.5: Измененный параметры в файле named.conf

6. Внесем изменения в настройки межсетевого экрана узла server, разрешив работу с DNS. Так же убедимся что DNS-запросы идут через узел server, который прослушивает порт 53.



The terminal window shows the following commands and output:

```
root@server:~
NetworkMa 7652 7664 gdbus          root  27u   IPv4          53693
0t0      UDP server.nskarmatskiy.net:bootpc->_gateway:bootps
[root@server.nskarmatskiy.net ~]# firewall-cmd --add-service=dns
Warning: ALREADY_ENABLED: 'dns' already in 'public'
success
[root@server.nskarmatskiy.net ~]# firewall-cmd --add-service=dns --permanent
Warning: ALREADY_ENABLED: dns
success
[root@server.nskarmatskiy.net ~]# lsof | grep UDP
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1002/gvfs
Output information may be incomplete.
avahi-dae 563      UDP *:mdns          avahi 12u   IPv4          18703
0t0
avahi-dae 563      UDP *:mdns          avahi 13u   IPv6          18704
0t0
avahi-dae 563      UDP *:49758        avahi 14u   IPv4          18705
0t0
avahi-dae 563      UDP *:54085        avahi 15u   IPv6          18706
0t0
named     840      UDP server.nskarmatskiy.net:domain  named 6u   IPv4          20308
0t0
named     840      UDP localhost:domain  named 16u   IPv4          19485
0t0
named     840      UDP                named 19u   IPv6          19487
```

Рис.2.6: Внесенные изменения и DNS-запросы

3. Конфигурирование кэширующего DNS-сервера при наличии фильтрации DNS-запросов маршрутизаторами

1. Добавим перенаправление DNS-запросов на конкретный вышестоящий DNS-сервер. Кроме того, возможно вышестоящий DNS-сервер может не поддерживать технологию DNSSEC, изменим код и в этом случае тоже

```
forwarders { 127.0.0.1; };  
forward first;  
  
/*  
- If you are building an AUTHORITATIVE DNS server, do NOT enable recursion.  
- If you are building a RECURSIVE (caching) DNS server, you need to enable  
  recursion.  
- If your recursive DNS server has a public IP address, you MUST enable access  
  control to limit queries to your legitimate users. Failing to do so can  
  cause your server to become part of large scale DNS amplification  
  attacks. Implementing BCP38 within your network would greatly  
  reduce such attack surface  
*/  
recursion yes;  
  
dnssec-validation no;
```

Рис. 3.1: Измененные параметры

4. Конфигурирование первичного DNS-сервера.

1. Скопируем шаблон описания DNS-зон named.rfc1912.zones из каталога /etc в каталог /etc/named и переименуем его в user.net (вместо user укажем свой логин)

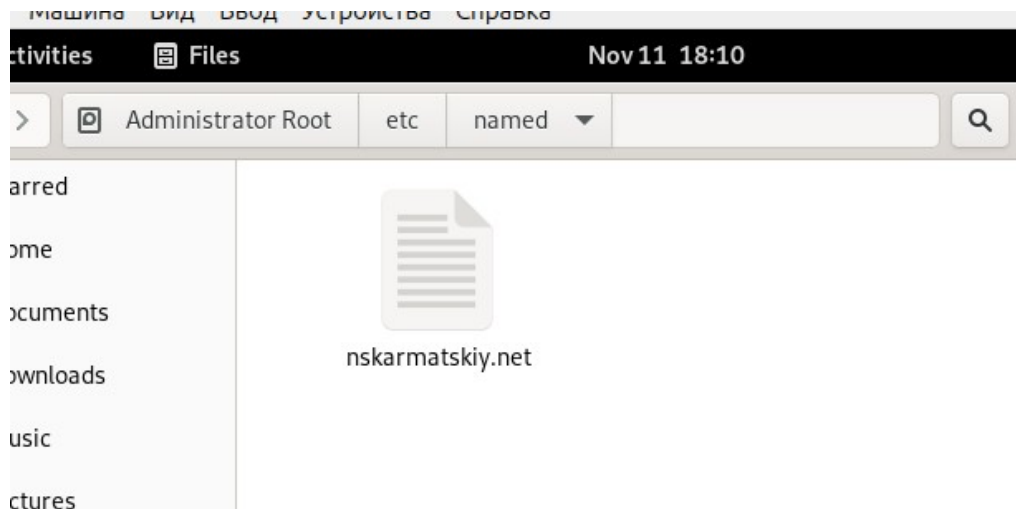


Рис. 4.1: Скопированный шаблон

2. Редактируем этот файл. Переписываем зоны.

```
14 // disable-empty-zone "."; into options
15 //
16
17 zone "nskarmatskiy.net" IN {
18     type master;
19     file "master/fz/nskarmatskiy.net";
20     allow-update { none; };
21 };
22
23 zone "1.168.192.in-addr.arpa" IN {
24     type master;
25     file "master/rz/192.168.1";
26     allow-update { none; };
27 };
28
29
```

Рис. 4.2: Отредактированный файл

3. В каталоге /var/named создаем подкаталоги master/fz и master/rz, в которых будут располагаться файлы прямой и обратной зоны соответственно.

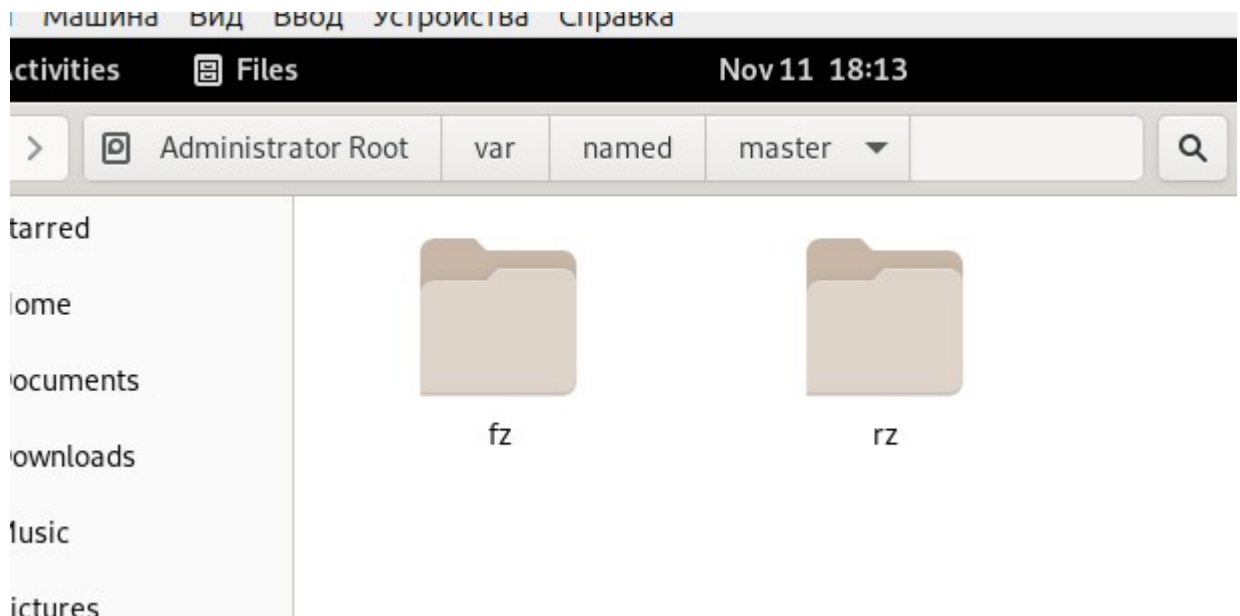


Рис.4.3: Созданные каталоги

4. Скопируем шаблон прямой DNS-зоны в каталог fz и переименуем его на nskarmatskiy.netю Изменим его, указав необходимые DNS-записи для прямой зоны.



Рис.4.4: Созданный и измененный файл прямой зоны

5. Скопируем шаблон обратной DNS-зоны в каталог rz и переименуем его в 192.168.1. Изменим его, указав необходимые DNS-записи для обратной зоны.

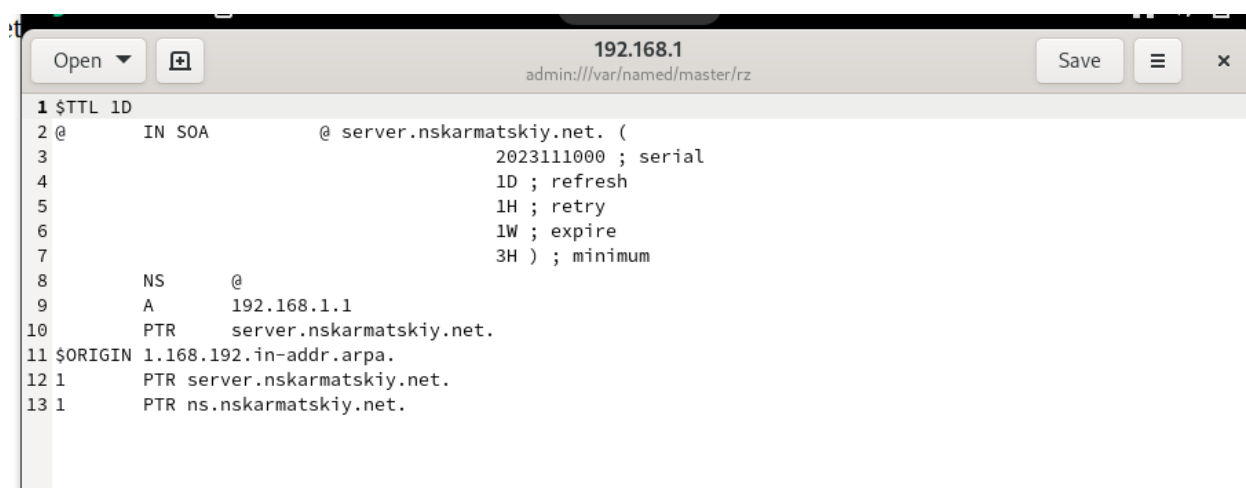
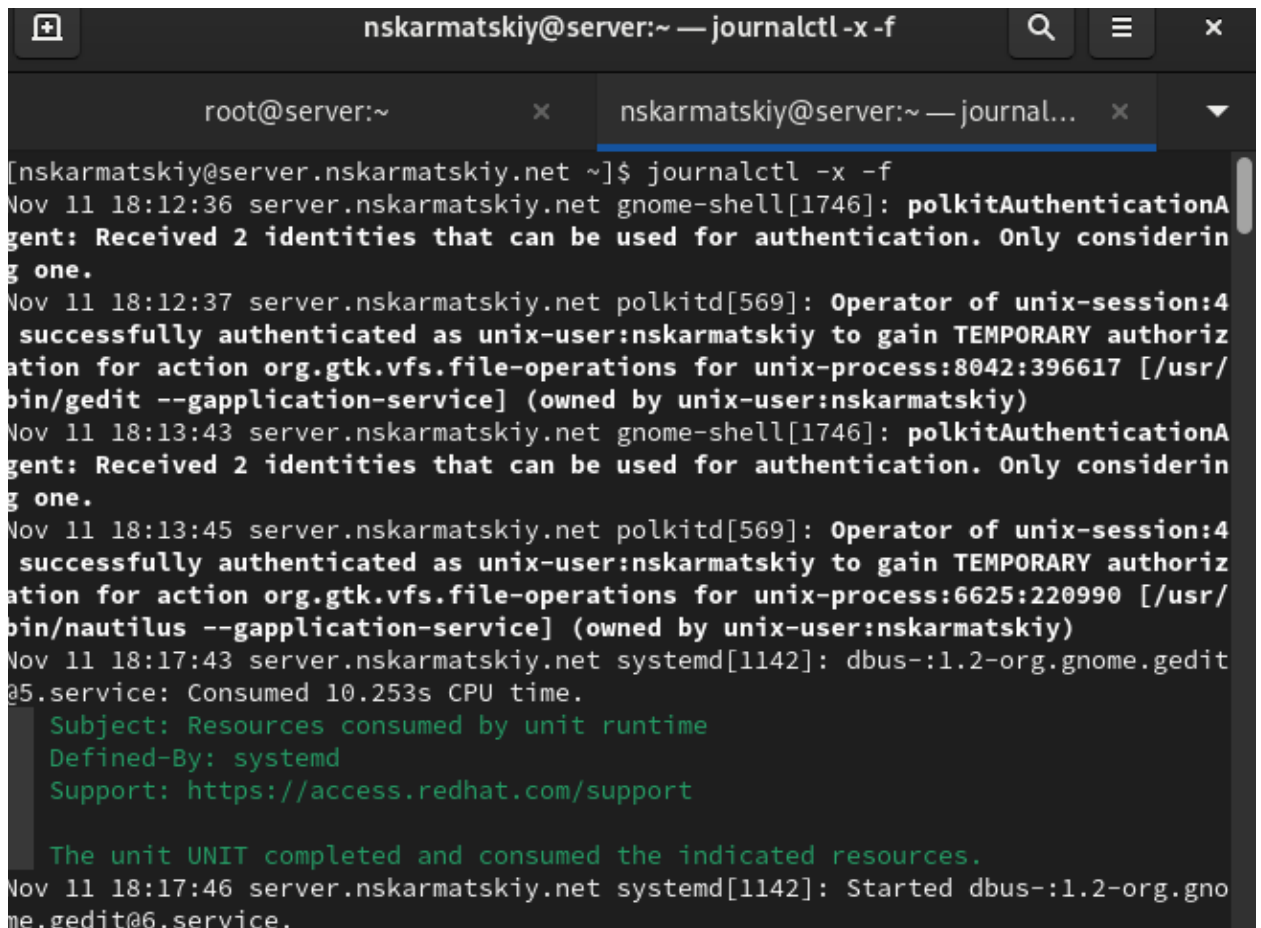


Рис.4.5: Созданный и измененный файл обратной зоны

6. Исправляем права доступа, чтобы демон named мог с ними работать. Так же восстановим метки в SELinux. В дополнительном терминале запустим в режиме реального времени расширенный лог системных сообщений, чтобы проверить работы системы, а в первом терминале перезапустим DNS-сервер.



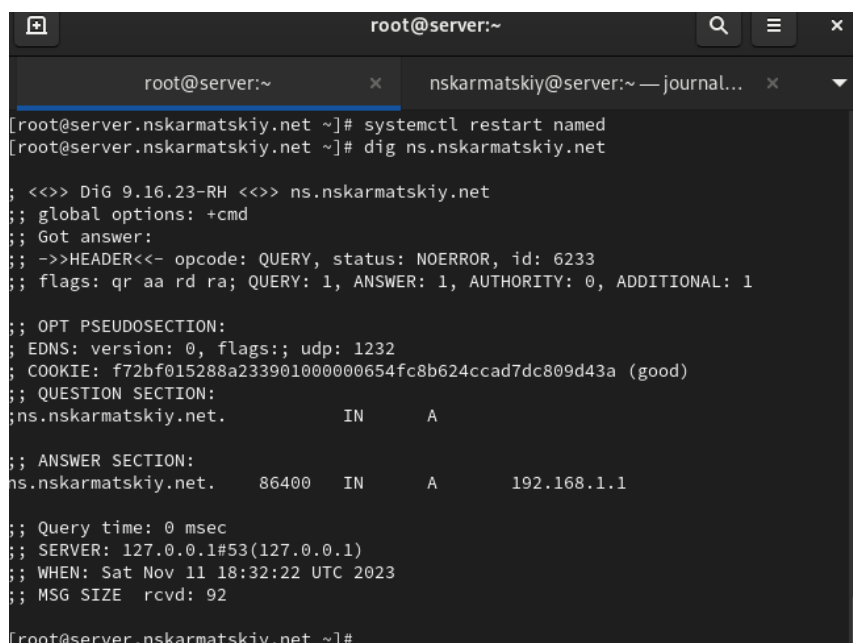
```
nskarmatskiy@server:~ — journalctl -x -f
root@server:~
[nskarmatskiy@server.nskarmatskiy.net ~]$ journalctl -x -f
Nov 11 18:12:36 server.nskarmatskiy.net gnome-shell[1746]: polkitAuthenticationAgent: Received 2 identities that can be used for authentication. Only considering one.
Nov 11 18:12:37 server.nskarmatskiy.net polkitd[569]: Operator of unix-session:4 successfully authenticated as unix-user:nskarmatskiy to gain TEMPORARY authorization for action org.gtk.vfs.file-operations for unix-process:8042:396617 [/usr/bin/gedit --gapplication-service] (owned by unix-user:nskarmatskiy)
Nov 11 18:13:43 server.nskarmatskiy.net gnome-shell[1746]: polkitAuthenticationAgent: Received 2 identities that can be used for authentication. Only considering one.
Nov 11 18:13:45 server.nskarmatskiy.net polkitd[569]: Operator of unix-session:4 successfully authenticated as unix-user:nskarmatskiy to gain TEMPORARY authorization for action org.gtk.vfs.file-operations for unix-process:6625:220990 [/usr/bin/nautilus --gapplication-service] (owned by unix-user:nskarmatskiy)
Nov 11 18:17:43 server.nskarmatskiy.net systemd[1142]: dbus-:1.2-org.gnome.gedit@5.service: Consumed 10.253s CPU time.
    Subject: Resources consumed by unit runtime
    Defined-By: systemd
    Support: https://access.redhat.com/support

    The unit UNIT completed and consumed the indicated resources.
Nov 11 18:17:46 server.nskarmatskiy.net systemd[1142]: Started dbus-:1.2-org.gnome.gedit@6.service.
```

Рис.4.6: Запущенный лог системных сообщений

5. Анализ работы DNS-сервера

1. При помощи утилиты dig получим описание DNS-зоны с сервера ns.user.net (вместо user должен быть указан наш логин):



```
root@server:~
[root@server.nskarmatskiy.net ~]# systemctl restart named
[root@server.nskarmatskiy.net ~]# dig ns.nskarmatskiy.net

;<>> DiG 9.16.23-RH <>> ns.nskarmatskiy.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6233
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1232
;; COOKIE: f72bf015288a233901000000654fc8b624ccad7dc809d43a (good)
;; QUESTION SECTION:
ns.nskarmatskiy.net.      IN      A

;; ANSWER SECTION:
ns.nskarmatskiy.net.      86400   IN      A       192.168.1.1

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sat Nov 11 18:32:22 UTC 2023
;; MSG SIZE rcvd: 92

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

Рис.5.1: Описание DNS-зоны

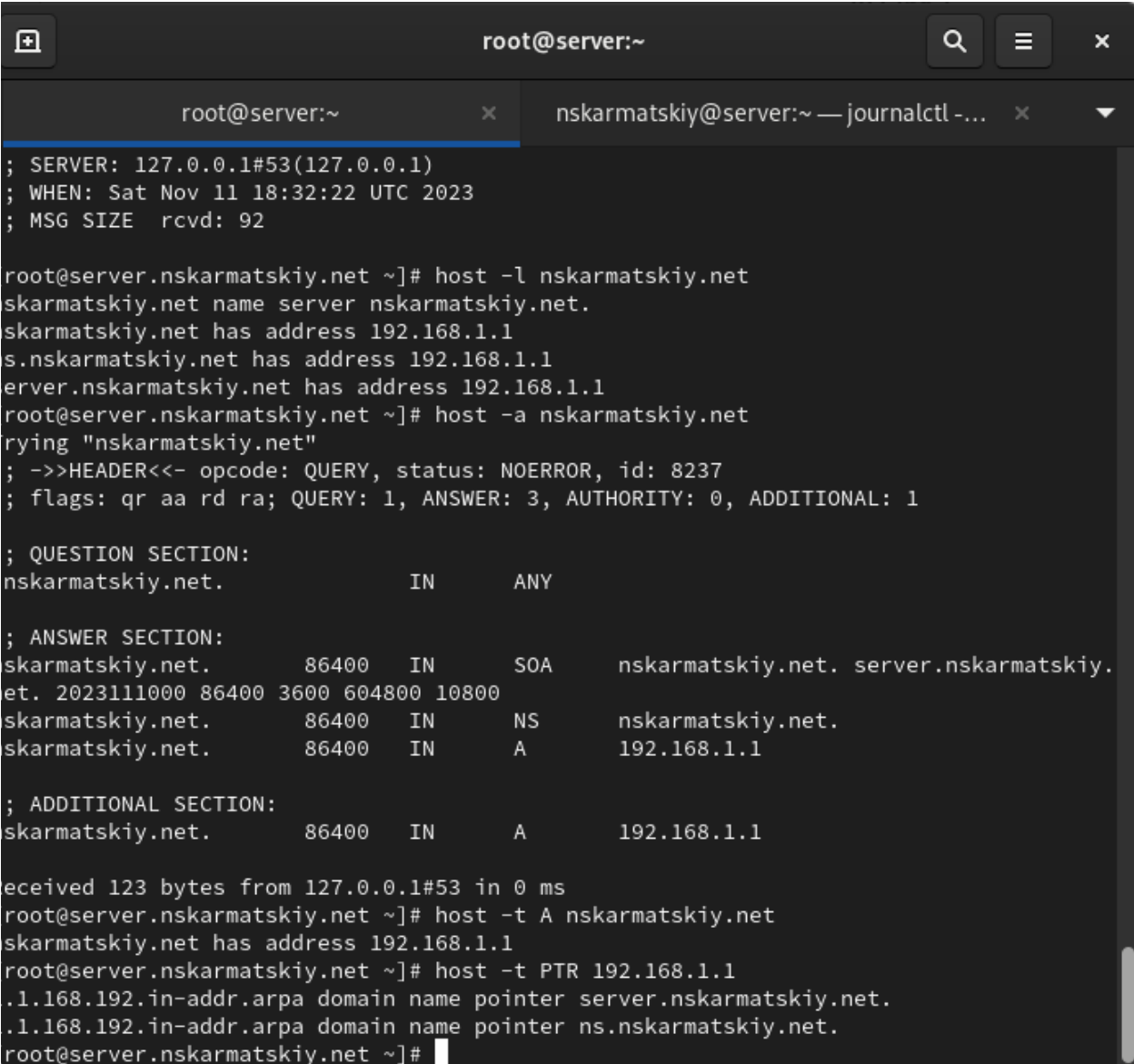
HEADER — отображает информацию о версии утилиты, ID запроса, полученных ошибках и использованных флагах вывода.

QUESTION SECTION — секция, которая отображает текущий запрос(ns.karmatskiy.net);

ANSWER SECTION — секция, в которой отображается результат обработки созданного запроса (в данном случае это IP-адрес домена).

Так же отображается время запросы, cookie, которые используются

2. При помощи утилиты host проанализируем корректность работы DNS-сервера. Как видим ниже, все работает корректно.



```
root@server:~  
; SERVER: 127.0.0.1#53(127.0.0.1)  
; WHEN: Sat Nov 11 18:32:22 UTC 2023  
; MSG SIZE rcvd: 92  
  
root@server.nskarmatskiy.net ~]# host -l ns.karmatskiy.net  
skarmatskiy.net name server ns.karmatskiy.net.  
skarmatskiy.net has address 192.168.1.1  
s.nskarmatskiy.net has address 192.168.1.1  
erver.nskarmatskiy.net has address 192.168.1.1  
root@server.nskarmatskiy.net ~]# host -a ns.karmatskiy.net  
rying "ns.karmatskiy.net"  
; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8237  
; flags: qr aa rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1  
  
; QUESTION SECTION:  
ns.karmatskiy.net.          IN      ANY  
  
; ANSWER SECTION:  
skarmatskiy.net.           86400   IN      SOA     ns.karmatskiy.net. server.nskarmatskiy.  
et. 20231111000 86400 3600 604800 10800  
skarmatskiy.net.           86400   IN      NS      ns.karmatskiy.net.  
skarmatskiy.net.           86400   IN      A       192.168.1.1  
  
; ADDITIONAL SECTION:  
skarmatskiy.net.           86400   IN      A       192.168.1.1  
  
received 123 bytes from 127.0.0.1#53 in 0 ms  
root@server.nskarmatskiy.net ~]# host -t A ns.karmatskiy.net  
skarmatskiy.net has address 192.168.1.1  
root@server.nskarmatskiy.net ~]# host -t PTR 192.168.1.1  
.1.168.192.in-addr.arpa domain name pointer server.nskarmatskiy.net.  
.1.168.192.in-addr.arpa domain name pointer ns.nskarmatskiy.net.  
root@server.nskarmatskiy.net ~]#
```

Рис.5.2: Проверка корректности работы.

6. Внесение изменений в настройки внутреннего окружения виртуальной машины

1. На виртуальной машине server перейдем в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/`, создаем в нём каталог `dns`, в который поместите в соответствующие каталоги конфигурационные файлы DNS

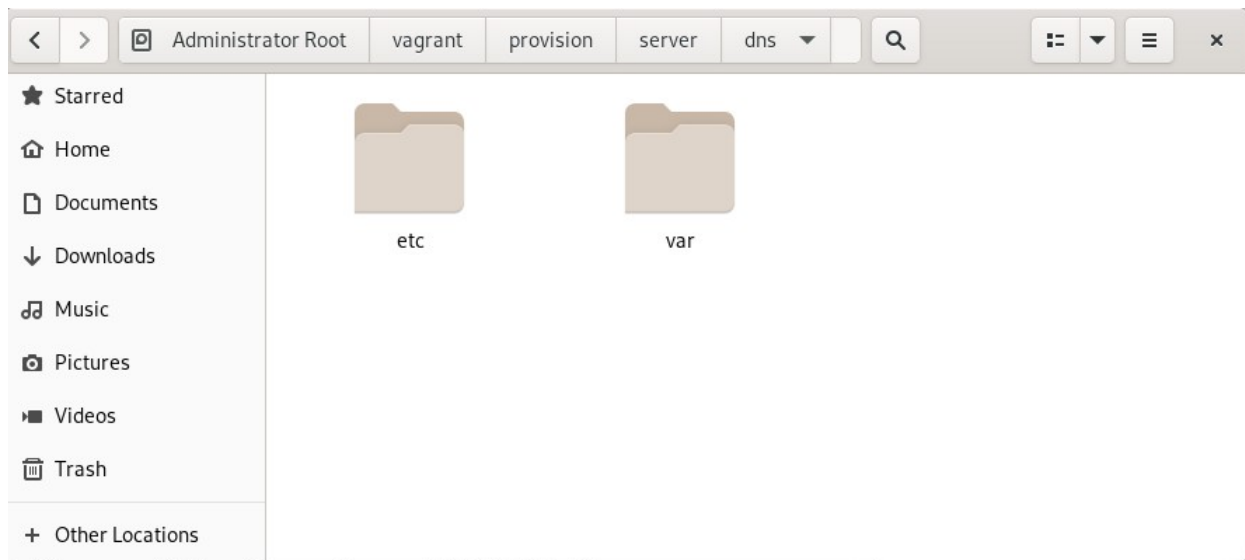


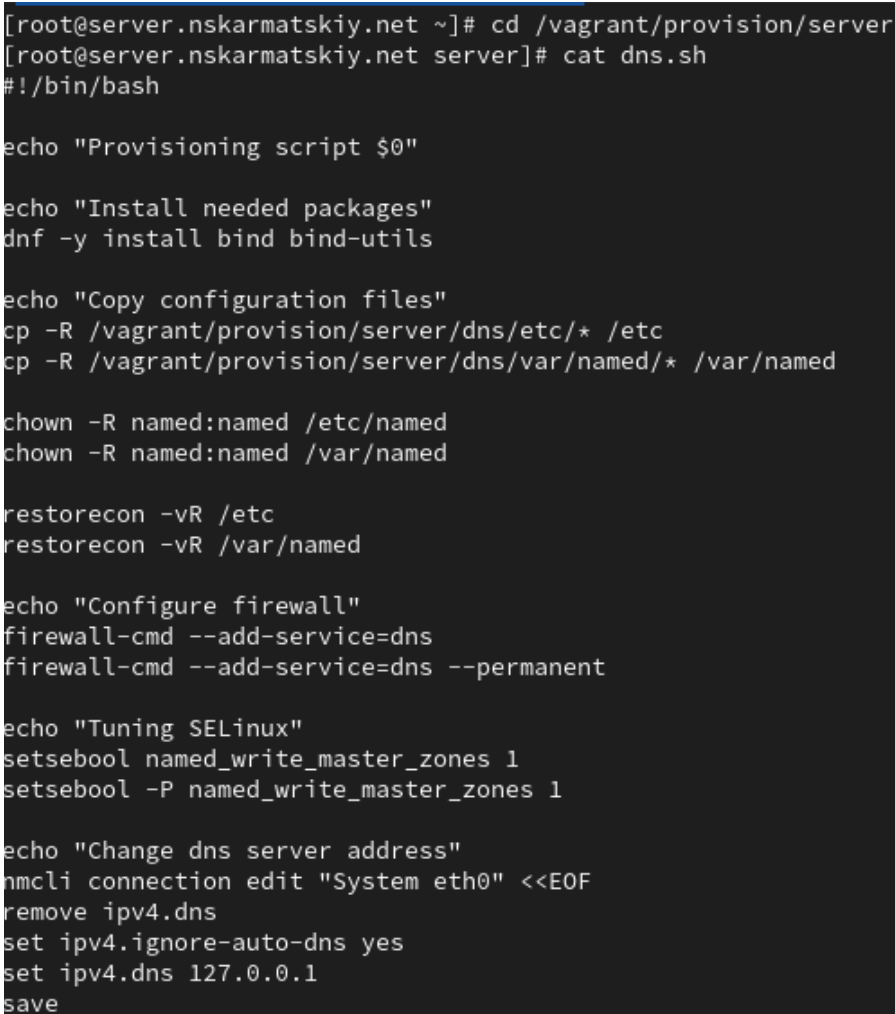
Рис.6.1: Перемещенные каталоги и файлы в них

2. В каталоге `/vagrant/provision/server` создаем исполняемый файл `dns.sh` и редактируем его. Вписываем данный скрипт:

```
#!/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
```



```
[root@server.nskarmatskiy.net ~]# cd /vagrant/provision/server
[root@server.nskarmatskiy.net server]# cat 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
```

Рис.6.2: Используемый скрипт в файле

3. 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 dns",  
type: "shell",  
preserve_order: true,  
path: "provision/server/dns.sh"
```



```
ip: "192.168.1.1",  
virtualbox____intnet: true  
  
server.vm.provision "server dns",  
type: "shell",  
preserve_order: true,  
path: "provision/server/dns.sh"  
  
server.vm.provision "server dummy",  
type: "shell",
```

Рис.6.3: Измененный файл конфигурации.

Вывод: Мы приобрели практических навыков по установке и конфигурированию DNS-сервера, усвоили принципы работы системы доменных имён.

Контрольные вопросы

1. DNS (Domain Name System):

- DNS - это система, обеспечивающая преобразование человеко-читаемых доменных имен в IP-адреса, используемые компьютерами для обмена данными.

2. Кэширующий DNS-сервер:

- Кэширующий DNS-сервер хранит копии запросов и ответов DNS. Его задача - ускорить доступ к ресурсам, кэшируя уже полученные ранее ответы и предоставляя их при

повторных запросах.

3. Прямая DNS-зона и обратная DNS-зона:

- Прямая DNS-зона отвечает за соответствие доменных имен и IP-адресов.
- Обратная DNS-зона используется для преобразования IP-адресов в соответствующие доменные имена.

4. Настройки DNS-сервера:

- Настройки DNS-сервера обычно хранятся в файлах:
 - /etc/named.conf - основной конфигурационный файл.
 - /var/named/ - каталог с файлами зон.
 - /etc/resolv.conf - файл с настройками резолвера.

5. Файл resolv.conf:

- В файле resolv.conf указываются DNS-серверы, которые будут использоваться для разрешения доменных имен в IP-адреса.

6. Типы записей DNS:

- A (IPv4 адрес)
- AAAA (IPv6 адрес)
- NS (имя DNS-сервера)
- PTR (обратная запись)
- MX (запись почтового обмена) и др.

7. Домен in-addr.arpa:

- Используется для обратного разрешения IP-адресов в доменные имена.

8. Демон named:

- Демон named (BIND) является программой, реализующей DNS-сервер.

9. Slave-сервер и Master-сервер:

- Master-сервер - авторитетный источник для зоны.
- Slave-сервер - копия зоны, обновляющаяся от Master-сервера.

10. Параметры времени обновления зоны:

- refresh, retry, expire, и minimum в SOA записи.

11. Защита зоны от скачивания и просмотра:

- Использование правильных прав доступа к файлам зоны и ограничение доступа.

12. Запись RR для почтовых серверов:

- MX (Mail Exchange) запись.

13. Тестирование работы сервера DNS:

- Использование команды nslookup или dig.

14. Управление службой в системе:

- systemctl start, systemctl restart, systemctl stop.

15. Отладочная информация при запуске службы:

- Использование опций -d или -v при запуске службы.

16. Хранение отладочной информации:

- В журналах системы, например, /var/log/messages.

17. Просмотр используемых файлов процессом:

- lsof -p <PID>.

18. Изменение сетевого соединения с помощью nmcli:

- Примеры: nmcli connection up, nmcli connection down.

19. SELinux (Security-Enhanced Linux):

- Это система безопасности для ядра Linux.

20. Контекст SELinux:

- Метка, присваиваемая объектам в системе, определяющая их права и политики безопасности.

21. Восстановление контекста SELinux:

- restorecon -Rv /path/to/directory.

22. Создание разрешающих правил из журналов SELinux:

- Использование утилиты audit2allow.

23. Булевый переключатель в SELinux:

- Это параметр, который включает или отключает конкретную функциональность.

24. Просмотр и изменение булевых переключателей SELinux:

- `getsebool -a, setsebool`.

25. Изменение значения переключателя SELinux:

- `setsebool -P <переключатель> <значение>`.