Отчет по лабораторной работе №2

Дисциплина: Администрирование сетевых подсистем

Иванов Сергей Владимирович

Содержание

# 1 Цель работы

Целью данной работы является приобретение практических навыков по установке и конфигурированию DNSсервера, 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

Загрузим операционную систему и перейдем в рабочий каталог с проектом: cd /var/tmp/svivanov/vagrant Запустим виртуальную машину server: vagrant up server (рис. 1).

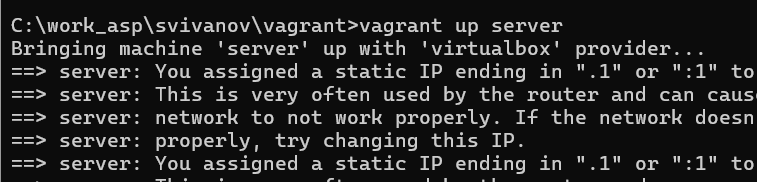


Рис. 1: Запуск server

На виртуальной машине server войдем под созданным в предыдущей работе пользователем и откроем терминал. Перейдем в режим суперпользователя: sudo -i. Установим bind и bind-utils: dnf -y install bind bind-utils (рис. 2).

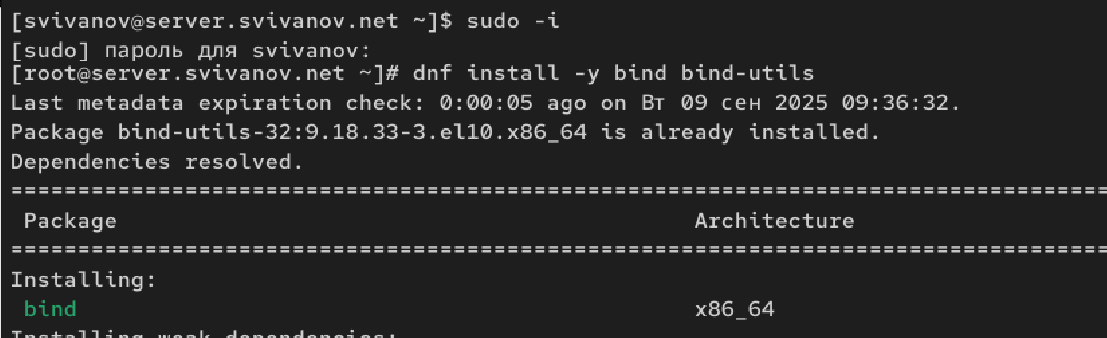


Рис. 2: Установка bind

В качестве упражнения с помощью утилиты dig сделайте запрос, например, к DNSадресу www.yandex.ru: dig www.yandex.ru. (рис. 3)

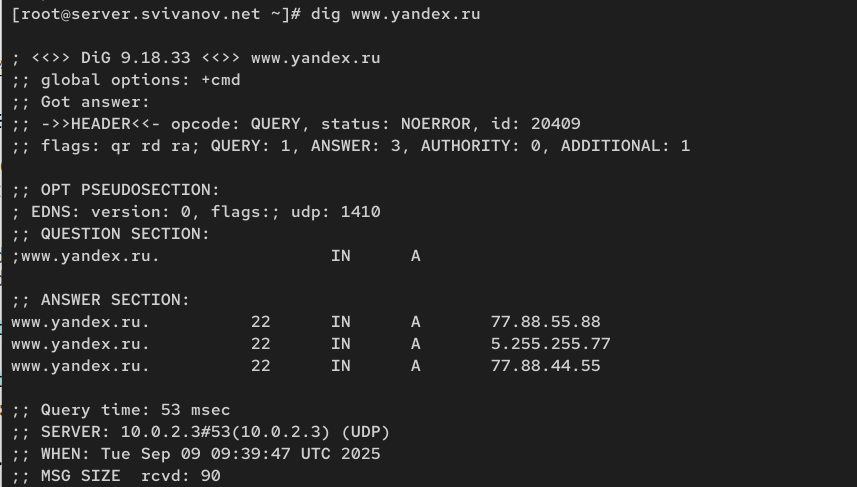


Рис. 3: Запрос с помощью утилиты dig

Анализ выведенной информации:

dig - Это инструмент для запроса DNS-серверов. В данном случае она запросила у DNS-сервера IP-адреса, связанные с доменным именем www.yandex.ru.

1. Программа отправила запрос на www.yandex.ru
2. Результат: успешный ответ с тремя IP-адресами для www.yandex.ru:

• 77.88.55.88

• 5.255.255.77

• 77.88.44.55

1. Время выполнения составило 53мс.

## 3.2 Конфигурирование кэширующего DNS-сервера

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

Посмотрим содержание файлов */etc/resolv.conf*, */etc/named.conf*, */var/named/named.ca*, */var/named/named.localhost* и */var/named/named.loopback* (рис. 4, 5, 6, 7, 8)

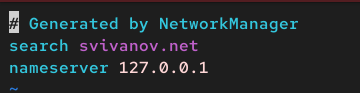


Рис. 4: Файл /etc/resolv.conf

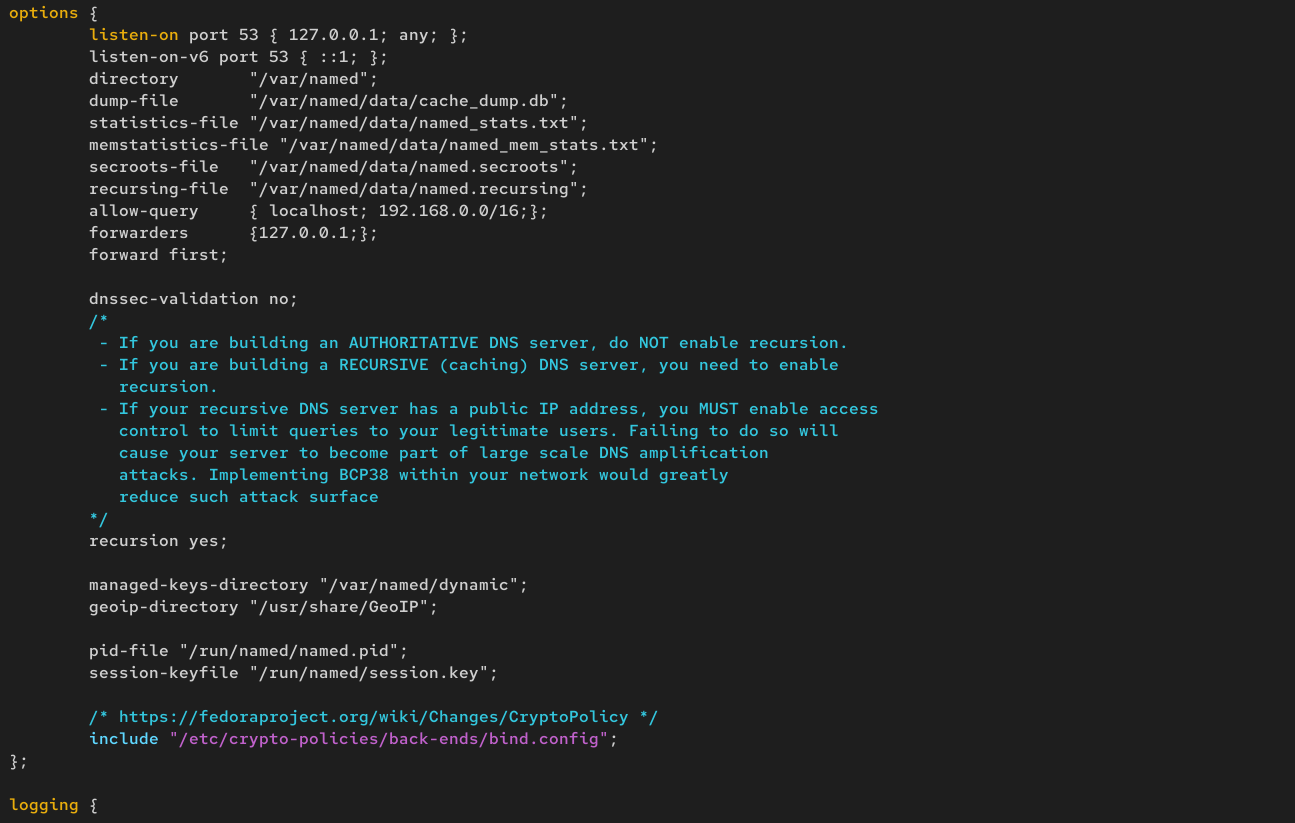


Рис. 5: Файл /etc/named.conf

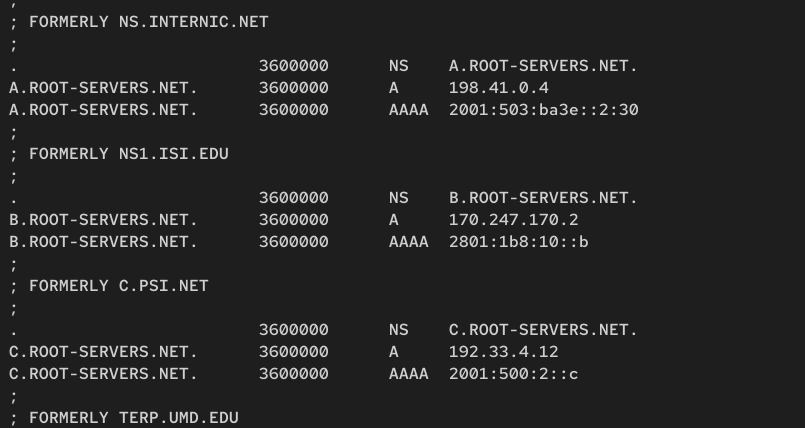


Рис. 6: Файл /var/named/named.ca

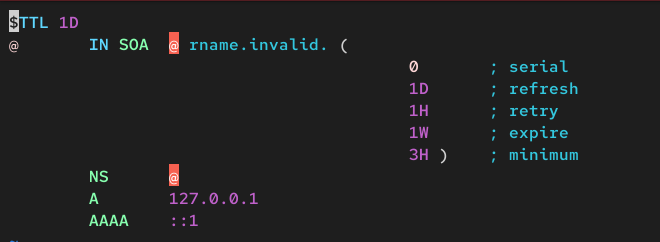


Рис. 7: Файл /var/named/named.localhost

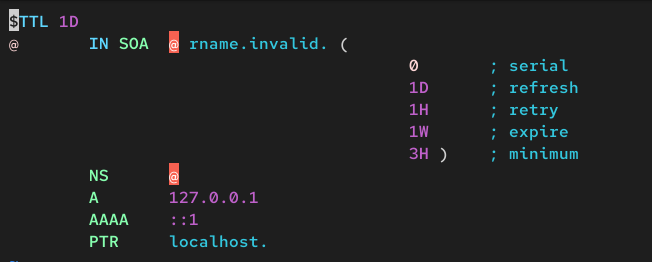


Рис. 8: Файл /var/named/named.loopback

Анализ файла */etc/resolv.conf* (уже отредатирован):

* Конфигурация DNS-клиента
* Указан DNS-сервер: 127.0.0.1
* Система использует публичные DNS-серверы

Анализ файла */etc/named.conf*:

* Основной конфигурационный файл BIND (DNS-сервер)
* Сервер настроен только для localhost (127.0.0.1)
* DNS-сервер работает только для локальных запросов

Анализ файла */var/named/named.ca*:

* Корневые DNS-серверы интернета
* Список root-серверов (A.ROOT-SERVERS.NET и т.д.)
* Кэш корневых серверов для работы DNS

Анализ файла */var/named/named.localhost*:

* Зона localhost для прямых запросов
* Настройки зоны для localhost (127.0.0.1)
* Базовая конфигурация для локальной зоны

Анализ файла */var/named/named.loopback*:

* Зона обратных запросов для localhost
* Обратная зона для 127.0.0.1
* Настройки reverse DNS для локальной сети

Запускаем DNS-сервер: systemctl start named. Включим запуск DNS-сервера в автозапуск при загрузке системы: systemctl enable named (рис. 9)

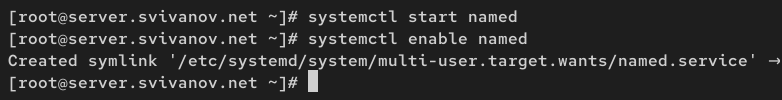


Рис. 9: Запуск DNS-сервера

Теперь выполним команду dig [**127.0.0.1?**] www.yandex.ru. При выполнении команды dig www.yandex.ru 1.1.1.1 (публичный DNS Cloudflare) - использовался автоматически, а при выполнении команды dig [**127.0.0.1?**] www.yandex.ru 127.0.0.1 (локальный DNS-сервер) - указан явно. (рис. 10)

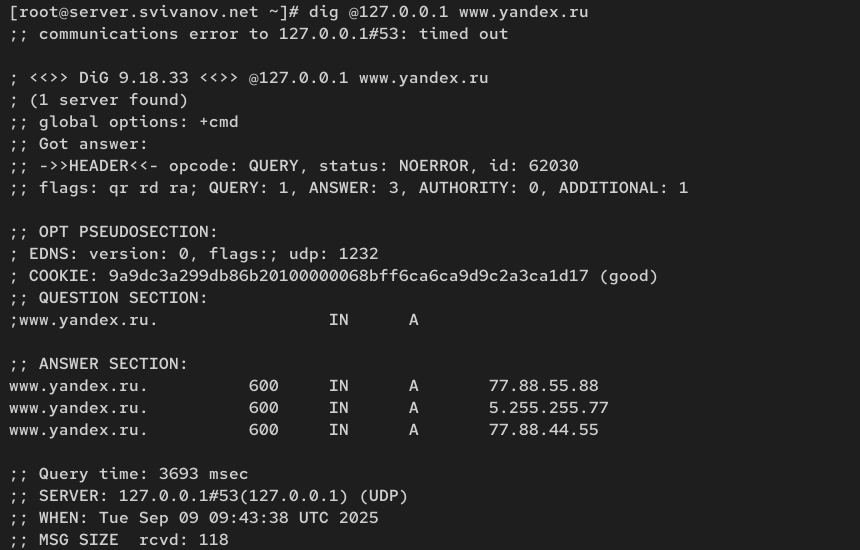


Рис. 10: Команда dig 127.0.0.1 www.yandex.ru

Сделаем DNS-сервер сервером по умолчанию для хоста server и внутренней виртуальной сети. Для этого требуется изменить настройки сетевого соединения eth0 в NetworkManager, переключив его на работу с внутренней сетью и указав для него в качестве DNS-сервера по умолчанию адрес 127.0.0.1:

nmcli connection edit eth0

remove ipv4.dns

set ipv4.ignore-auto-dns yes

set ipv4.dns 127.0.0.1

save

quit (рис. 11)

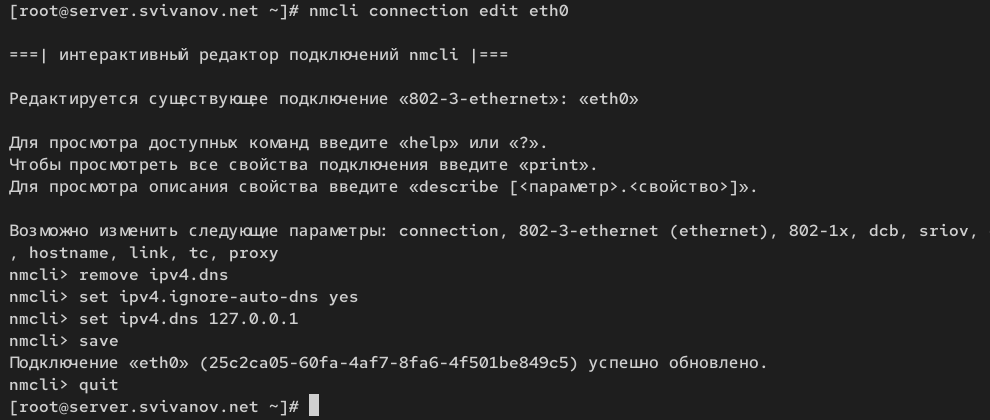


Рис. 11: Скрипт маршрутизации

Перезапустим NetworkManager: systemctl restart NetworkManager. Проверим наличие изменений в файле /etc/resolv.conf. (рис. 12)

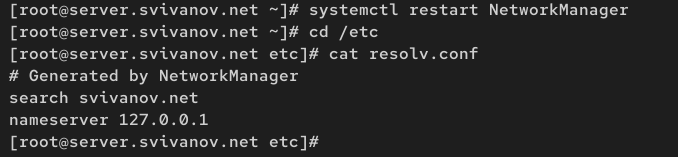


Рис. 12: Перезапуск NetworkManager

Настроим направление DNS-запросов от всех узлов внутренней сети, включая запросы от узла server, через узел server. Для этого внесем изменения в файл /etc/named.conf, заменив строку

listen-on port 53 { 127.0.0.1; };

на

listen-on port 53 { 127.0.0.1; any; };

и строку

allow-query { localhost; };

на

allow-query { localhost; 192.168.0.0/16; }; (рис. 13)

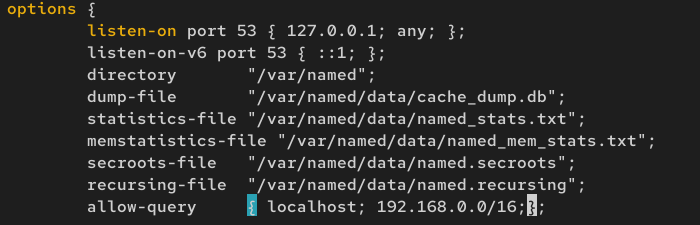


Рис. 13: Редактирование файла /etc/named.conf

Внесем изменения в настройки межсетевого экрана узла server, разрешив работу с DNS: (рис. 14)

firewall-cmd –add-service=dns

firewall-cmd –add-service=dns –permanent

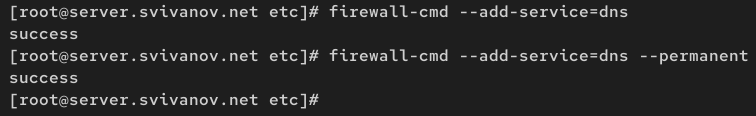


Рис. 14: Настройки межсетевого экрана server

Убедимся, что DNS-запросы идут через узел server, который прослушивает порт 53. Для этого используем команду lsof: lsof | grep UDP (рис. 15)

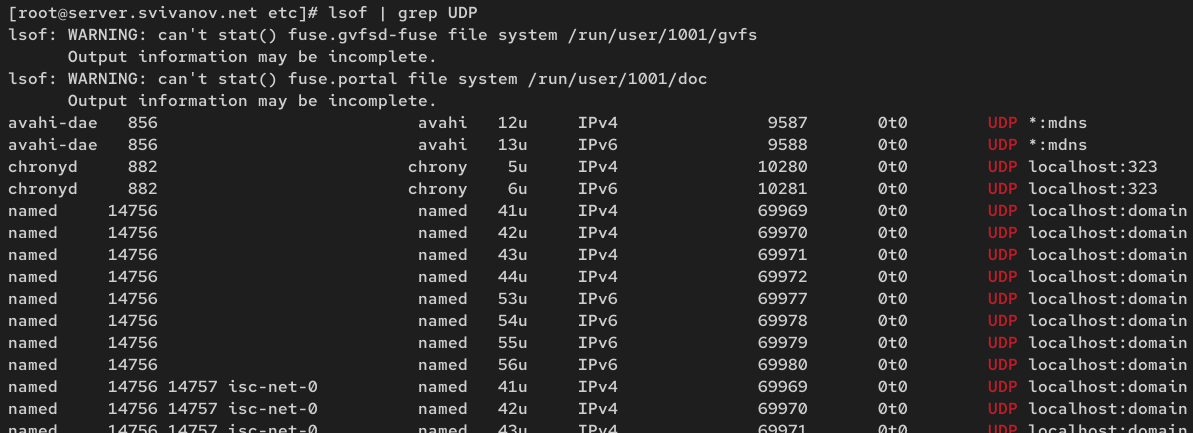


Рис. 15: Проверка что DNS-запросы идут через узел server (порт 53)

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

В случае возникновения в сети ситуации, когда DNS-запросы от сервера фильтруются сетевым оборудованием, следует добавить перенаправление DNS-запросов на конкретный вышестоящий DNS-сервер. Для этого в конфигурационный файл named.conf в секцию options следует добавить:

forwarders { список DNS-серверов };

forward first; (рис. 16)

Кроме того, возможно вышестоящий DNS-сервер может не поддерживать технологию DNSSEC, тогда следует в конфигурационном файле named.conf укажем следующие настройки:

dnssec-enable no;

dnssec-validation no;

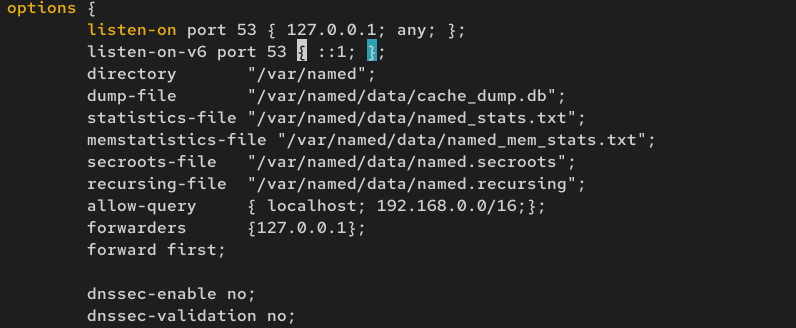


Рис. 16: Редактирование named.conf

Скопируем шаблон описания DNS-зон named.rfc1912.zones из каталога /etc в каталог /etc/named и переименуем его в svivanov.net:

cp /etc/named.rfc1912.zones /etc/named/

cd /etc/named

mv /etc/named/named.rfc1912.zones /etc/named/svivanov.net (рис. 17)

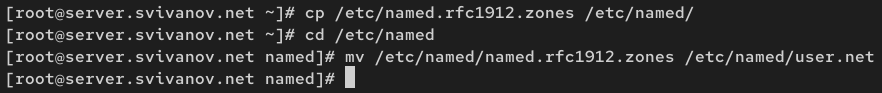


Рис. 17: Копирование шаблона описания DNS-зон

Включим файл описания зоны /etc/named/svivanov.net в конфигурационном файле DNS /etc/named.conf, добавив в нём в конце строку: include “/etc/named/svivanov.net”; (рис. 18)

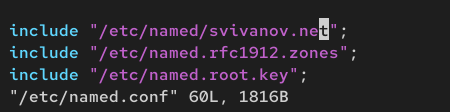


Рис. 18: Редактирование svivanov.net

Откроем файл /etc/named/svivanov.net на редактирование и вместо зоны

zone “localhost.localdomain” IN {

type master;

file “named.localhost”;

allow-update { none; };

};

пропишем свою прямую зону:

zone “user.net” IN {

type master;

file “master/fz/user.net”;

allow-update { none; };

};

Далее, вместо зоны

zone “1.0.0.127.in-addr.arpa” IN {

type master;

file “named.loopback”;

allow-update { none; };

};

пропишем свою обратную зону:

zone “1.168.192.in-addr.arpa” IN {

type master;

file “master/rz/192.168.1”;

allow-update { none; };

};

Остальные записи в файле /etc/named/user.net удалим. (рис. 19)

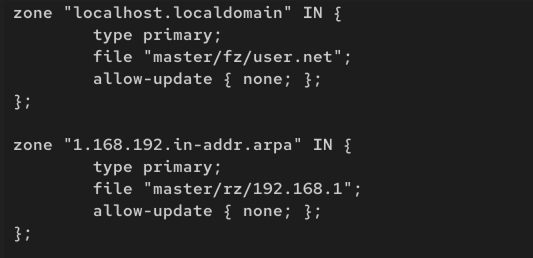


Рис. 19: Редактирование svivanov.net

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

cd /var/named

mkdir -p /var/named/master/fz

mkdir -p /var/named/master/rz (рис. 20)

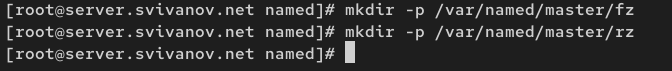


Рис. 20: Создание подкаталогов

Скопируем шаблон прямой DNS-зоны named.localhost из каталога /var/named в каталог /var/named/master/fz и переименуем его в user.net:

cp /var/named/named.localhost /var/named/master/fz/

cd /var/named/master/fz/

mv named.localhost user.net (рис. 21)

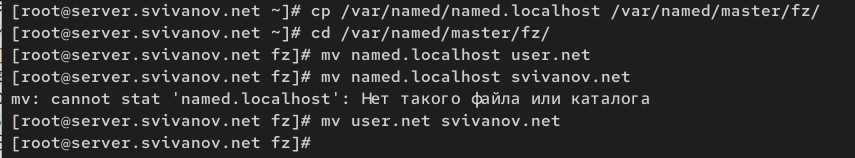


Рис. 21: Копирование шаблона прямой DNS-зоны

Изменим файл /var/named/master/fz/user.net, указав необходимые DNS-записи для прямой зоны. В этом файле DNS-имя сервера @ rname.invalid. должно быть заменено на @ server.user.net.; формат серийного номера ГГГГММДДВВ (ГГГГ — год, ММ — месяц, ДД — день, ВВ — номер ревизии); адрес в A-записи должен быть заменён с 127.0.0.1 на 192.168.1.1; в директиве $ORIGIN должно быть задано текущее имя домена user.net. (вместо user должен быть указан ваш логин), а затем указаны имена и адреса серверов в этом домене в виде A-записей DNS (на данном этапе должен быть прописан сервер с именем ns и адресом 192.168.1.1). (рис. 22)

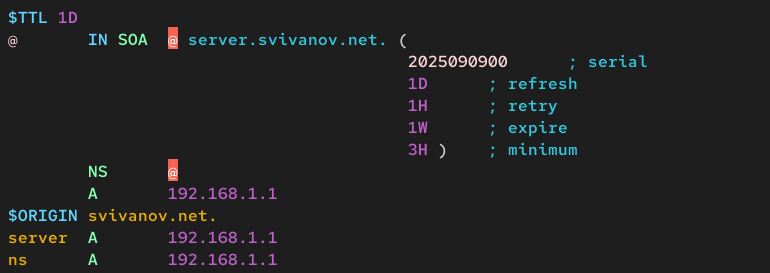


Рис. 22: Редактирование файла /var/named/master/fz/user.net

Скопируем шаблон обратной DNS-зоны named.loopback из каталога /var/named в каталог /var/named/master/rz и переименуем его в 192.168.1:

cp /var/named/named.loopback /var/named/master/rz/

cd /var/named/master/rz/

mv named.loopback 192.168.1 (рис. 23)

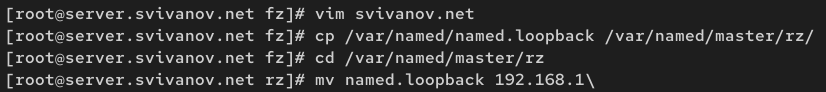


Рис. 23: Копирование шаблона обратной DNS-зоны

Изменим файл /var/named/master/rz/192.168.1, указав необходимые DNS-записи для обратной зоны. В этом файле DNS-имя сервера @ rname.invalid. должно быть заменено на @ server.user.net.; формат серийного номера ГГГГММДДВВ (ГГГГ — год, ММ — месяц, ДД — день, ВВ — номер ревизии); адрес в A-записи должен быть заменён с 127.0.0.1 на 192.168.1.1; в директиве $ORIGIN должно быть задано название обратной зоны в виде 1.168.192.in-addr.arpa., затем заданы PTR-записи. (рис. 24)

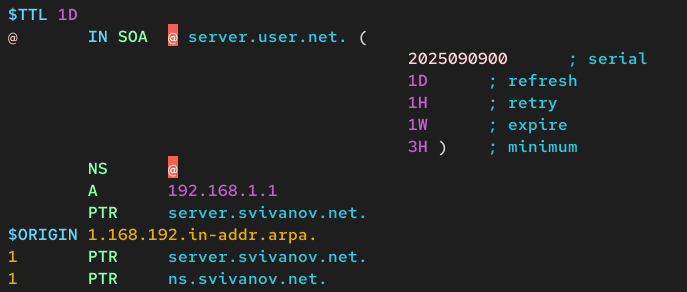


Рис. 24: Редактирование файла /var/named/master/rz/192.168.1

Далее требуется исправить права доступа к файлам в каталогах /etc/named и /var/named, чтобы демон named мог с ними работать:

chown -R named:named /etc/named

chown -R named:named /var/named (рис. 25)

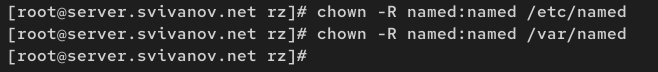


Рис. 25: Испрвление прав доступа

В системах с запущенным SELinux все процессы и файлы имеют специальные метки безопасности, используемые системой для принятия решений по доступу к этим процессам и файлам. После изменения доступа к конфигурационным файлам named требуется корректно восстановить их метки в SELinux:

restorecon -vR /etc

restorecon -vR /var/named

Для проверки состояния переключателей SELinux, относящихся к named, введем: getsebool -a | grep named . Дадим named разрешение на запись в файлы DNS-зоны:

setsebool named\_write\_master\_zones 1

setsebool -P named\_write\_master\_zones 1 (рис. 26)

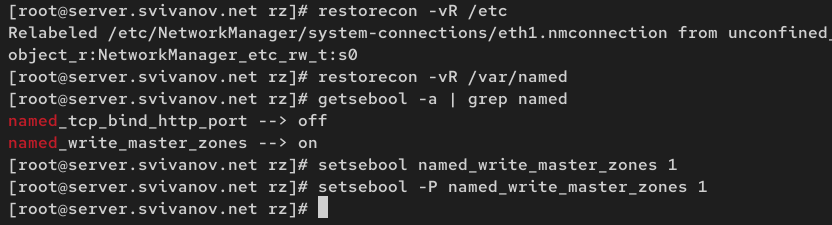


Рис. 26: Восстановление меток в SELinux и их рповерка

В дополнительном терминале запустим в режиме реального времени расширенный лог системных сообщений, чтобы проверить корректность работы системы: journalctl -x -f. В первом терминале перезапустим DNS-сервер: systemctl restart named (рис. 27)

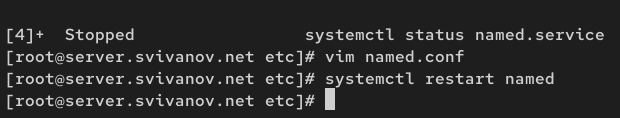


Рис. 27: Перезапуск DNS-сервера

При помощи утилиты dig получим описание DNS-зоны с сервера ns.svivanov.net: dig ns.user.net (рис. 28)

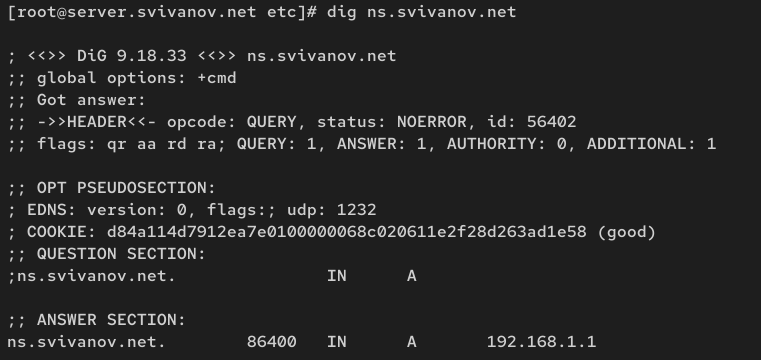


Рис. 28: Описание DNS-зоны с сервера ns.svivanov.net

Анализ выведенной информации: Команда dig ns.svivanov.net запросила IP-адрес DNS-сервера домена.

• Сервер: 127.0.0.1 (локальный)

• Ответ: ns.svivanov.net = 192.168.1.1

• Статус: aa (authoritative answer) - ответ авторитативный

• Время: 1 мс

Локальный DNS-сервер отвечает, что DNS-сервер домена находится по адресу 192.168.1.1.

При помощи утилиты host проанализируем корректность работы DNS-сервера:

host -l user.net

host -a user.net

host -t A user.net

host -t PTR 192.168.1.1

Как видим, сервер работает корректно. (рис. 29)

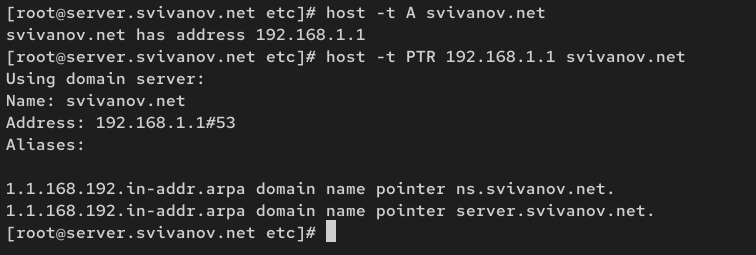


Рис. 29: Проверка корректности работы DNS-сервера

На виртуальной машине server перейдем в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создайте в нём каталог dns, в который поместим в соответствующие каталоги конфигурационные файлы DNS: (рис. 30)



Рис. 30: Создание каталога dns с нужными файлами

В каталоге /vagrant/provision/server создадим исполняемый файл dns.sh:

touch dns.sh

chmod +x dns.sh

Открыв его на редактирование, пропишем в нём следующий скрипт: (рис. 31)

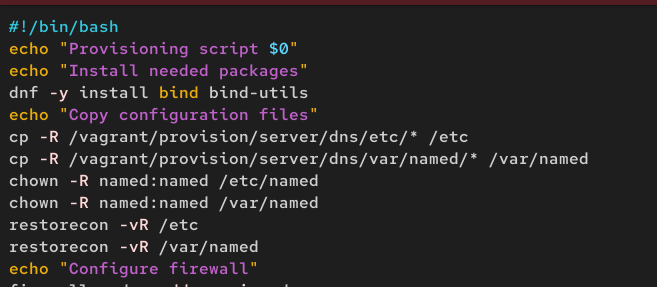


Рис. 31: Скрипт dns.sh

Для отработки созданного скрипта во время загрузки виртуальной машины server в конфигурационном файле Vagrantfile добавим в разделе конфигурации для сервера: (рис. 32)

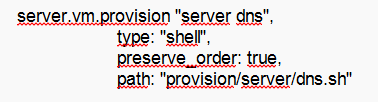


Рис. 32: Редактирование Vagrantfile

# 4 Ответы на контрольные вопросы

**1. Что такое DNS?**

Это система, предназначенная для преобразования человекочитаемых доменных имен в IP-адреса, используемые компьютерами для распознавания друг друга в сети.

**2. Каково назначение кэширующего DNS-сервера?**

Его главная цель — ускорить получение ответа и снизить нагрузку на сеть. Он запоминает результаты предыдущих запросов на некоторое время. Если два разных пользователя запрашивают один и тот же сайт, второму ответ придет мгновенно из кэша, без повторного опроса внешних серверов.

**3. Чем отличается прямая DNS-зона от обратной?**

Прямая зона преобразует доменные имена в IP-адреса, обратная зона выполняет обратное: преобразует IP-адреса в доменные имена.

**4. В каких каталогах и файлах располагаются настройки DNS-сервера? Кратко охарактеризуйте, за что они отвечают.**

В Linux-системах обычно используется файл /etc/named.conf для общих настроек. Зоны хранятся в файлах в каталоге /var/named/, например, /var/named/example.com.zone

**5. Что указывается в файле resolv.conf?**

В этом файле прописываются адреса DNS-серверов, которые будет использовать эта машина для преобразования имен.

**6. Какие типы записи описания ресурсов есть в DNS и для чего они используются?**

A (IPv4-адрес), AAAA (IPv6-адрес), CNAME (каноническое имя), MX (почтовый сервер), NS (имя сервера), PTR (обратная запись), SOA (начальная запись зоны), TXT (текстовая информация).

**7. Для чего используется домен in-addr.arpa?**

Используется для обратного маппинга IP-адресов в доменные имена.

**8. Для чего нужен демон named?**

Это DNS-сервер, реализация BIND (Berkeley Internet Name Domain).

**9. В чём заключаются основные функции slave-сервера и master-сервера?**

Master-сервер хранит оригинальные записи зоны, slave-серверы получают копии данных от master-сервера

**10. Какие параметры отвечают за время обновления зоны?**

refresh, retry, expire, и minimum.

**11. Как обеспечить защиту зоны от скачивания и просмотра?**

Можно запретить трансфер зоны (операцию zone transfer) для посторонних серверов, разрешив его только для своих слейвов с помощью директивы allow-transfer в named.conf.

**12. Какая запись RR применяется при создании почтовых серверов?**

MX (Mail Exchange).

**13. Как протестировать работу сервера доменных имён?**

Использовать команды nslookup, dig, или host.

**14. Как запустить, перезапустить или остановить какую-либо службу в системе?**

systemctl start|stop|restart .

**15. Как посмотреть отладочную информацию при запуске какого-либо сервиса или службы?**

Использовать опции, такие как -d или -v при запуске службы.

**16. Где храниться отладочная информация по работе системы и служб? Как её посмотреть?**

В системных журналах, доступных через journalctl

**17. Как посмотреть, какие файлы использует в своей работе тот или иной процесс?**

* lsof -p или fuser -v

**18. Приведите несколько примеров по изменению сетевого соединения при помощи командного интерфейса nmcli.**

nmcli connection up|down .

**19. Что такое SELinux?**

Это мандатный контроль доступа для ядра Linux.

**20. Что такое контекст (метка) SELinux?**

Метка, определяющая, какие ресурсы могут быть доступны процессу или объекту.

**21. Как восстановить контекст SELinux после внесения изменений в конфигурационные файлы?**

restorecon -Rv .

**22. Как создать разрешающие правила политики SELinux из файлов журналов, содержащих сообщения о запрете операций?**

audit2allow.

**23. Что такое булевый переключатель в SELinux?**

Это параметр, который включает или отключает определенные аспекты защиты SELinux.

**24. Как посмотреть список переключателей SELinux и их состояние?**

getsebool -a.

**25. Как изменить значение переключателя SELinux?**

setsebool -P <on|off>.

# 5 Выводы

В ходе выполнения лабораторной работы мы приобрели практические навыки по установке и конфигурированию DNSсервера, а также усвоили принципы работы системы доменных имён.