

Администрирование сетевых подсистем

Лабораторная работа №2

Тойчубекова Асель Нурлановна

Содержание

1	Цель работы	6
2	Задание	7
3	Теоретическое введение	8
4	Выполнение лабораторной работы	10
5	Выводы	31
	Список литературы	32

Список иллюстраций

4.1	Переход в режим суперпользователя	10
4.2	Установка bind	11
4.3	Утилита dig	12
4.4	Файл /etc/resolv.conf	12
4.5	Файл named.conf	13
4.6	Файл named.ca	14
4.7	Файл named.localhost	15
4.8	Файл named.loopback	15
4.9	Запуск DNS-сервера	16
4.10	Вывод <code>ig 127.0.0.1 www.yandex.ru.</code>	16
4.11	DNS-сервер сервером по умолчанию для хоста server и внутренней виртуальной сети	17
4.12	DNS-сервер сервером по умолчанию для хоста server и внутренней виртуальной сети	17
4.13	Перезапуск NetworkManager	17
4.14	Настройка направление DNS-запросов	18
4.15	Внесение изменений	18
4.16	Проверка DNS-запросов	19
4.17	Перенаправление DNS-запросов	20
4.18	Редактирование файлов	20
4.19	Включение файла описания зоны /etc/named/antoychubekova.net	21
4.20	Редактирование /etc/named/antoychubekova.net	22
4.21	Редактирование /var/named	22
4.22	Копирование шаблон прямой DNS-зоны	22
4.23	Редактирование /var/named/master/fz/antoychubekova.net	23
4.24	Копирование шаблон обратной DNS-зоны named.loopback	23
4.25	Редактирование /var/named/master/rz/192.168.1	24
4.26	Изменение прав доступа к /etc/named и /var/named	24
4.27	Корректное восстановление меток в SELinux	24
4.28	Состояния переключателей SELinux	25
4.29	Предоставление разрешения на запись в файлы DNS-зоны	25
4.30	Проверка корректности работы системы	25
4.31	Описание DNS-зоны с сервера ns.antoychubekova.net	26
4.32	Корректность работы DNS-сервера	26
4.33	Корректность работы DNS-сервера	27
4.34	Корректность работы DNS-сервера	27

4.35	Корректность работы DNS-сервера	27
4.36	Редактирование конфигурационных файлов DNS	28
4.37	Создание исполняемого файла dns.sh	28
4.38	Редактирование исполняемого файла dns.sh	29
4.39	Редактирование Vagrantfile	30

Список таблиц

1 Цель работы

Целью данной лабораторной работы является 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 Теоретическое введение

Система доменных имён (Domain Name System, DNS) — распределённая система (распределённая база данных), ставящая в соответствие доменному имени хоста (компьютера или другого сетевого устройства) IP-адрес, и наоборот.

Компоненты:

- DNS-сервер – программа для обслуживания DNS-запросов (чаще BIND).
- DNS-клиент – библиотека или программа для работы с DNS.
- Зона – логический узел в дереве доменов.
- Домен – название зоны.
- Поддомен – подчинённая зона.

Типы DNS-серверов:

- Primary master – основной сервер зоны, читает данные из файла.
- Secondary master – получает данные зоны от primary.
- Кэширующий – обрабатывает рекурсивные запросы клиентов.

Файлы зоны и директивы:

- \$ORIGIN – задаёт текущий домен.
- \$INCLUDE – включает другой файл в описание зоны.

Записи ресурсов (RR):

- SOA – авторитетная запись зоны, содержит origin, contact, serial, refresh, retry, expire, minimum.
- NS – DNS-серверы зоны.
- A – имя хоста → IP-адрес.
- PTR – IP-адрес → имя хоста.
- CNAME – каноническое имя для псевдонимов.
- MX – почтовые серверы с приоритетом.

Примеры форматов:

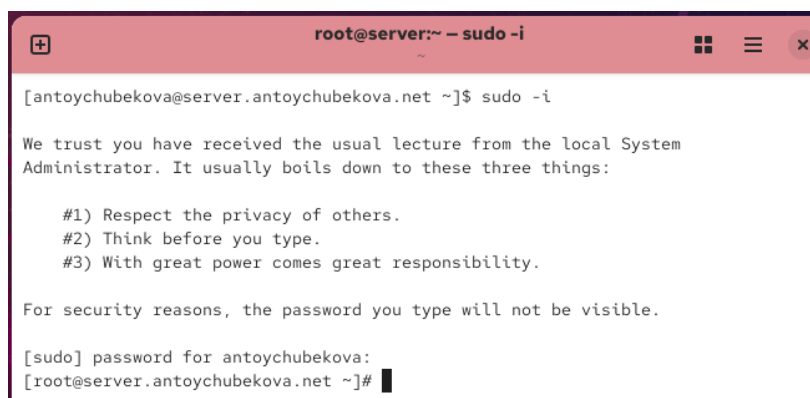
- SOA: [zone] [ttl] IN SOA origin contact (serial refresh retry expire minimum)
- NS: [domain] [ttl] IN NS [server]
- A: [host] [ttl] IN A [address]
- PTR: [name] [ttl] IN PTR [host]
- MX: [name] [ttl] IN MX [preference] [host]
- CNAME: [nickname] [ttl] IN CNAME [host]

Утилита dig (domain information groper) предоставляет пользователю интерфейс командной строки для обращения к системе DNS, позволяет формировать запросы о доменах DNS-серверам. Утилита dig входит в стандартный комплект DNS сервера BIND.

Утилита host предназначена для выполнения запросов к DNS-серверам.

4 Выполнение лабораторной работы

Для начала выполнения второй лабораторной работы запустить виртуальную машину server. На виртуальной машине server вхожу под созданным предыдущей работе пользователем, antoychubekova и открываю терминал и перехожу в режим суперпользователя (рис. 4.1).



```
root@server:~ - sudo -i
[antoychubekova@server.antoychubekova.net ~]$ sudo -i

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

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

For security reasons, the password you type will not be visible.

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

Рисунок 4.1: Переход в режим суперпользователя

Устанавливаю bind и bind-utils. (рис. 4.2).

```
root@server:~ -- sudo -i

[root@server.antoychubekova.net ~]# dnf -y install bind bind-utils
Last metadata expiration check: 2:21:54 ago on Thu 11 Sep 2025 08:46:12 AM UTC.
Package bind-utils-32:9.18.33-3.el10.x86_64 is already installed.
Dependencies resolved.
=====
Package                Arch      Version              Repository           Size
=====
Installing:
bind                   x86_64    32:9.18.33-3.el10    appstream            333 k
Installing weak dependencies:
bind-dnssec-utils      x86_64    32:9.18.33-3.el10    appstream            150 k
=====
Transaction Summary
=====
Install 2 Packages
-----
```

Рисунок 4.2: Установка bind

В качестве упражнения с помощью утилиты `dig` делаю запрос, к DNS-адресу `www.yandex.ru`.

В первой строке указывается версия утилиты `DiG 9.18.33` и сам домен, к которому выполняется запрос. Далее сообщается, что ответ получен, в заголовке видно, что это был обычный запрос (`QUERY`) со статусом `NOERROR` (ошибок нет), и указан идентификатор запроса. Флаги показывают, что это ответ (`qr`), была запрошена рекурсия (`rd`), и сервер её поддерживает (`ra`). Также указано, что в ответе одна секция запроса, три записи-ответа и одна дополнительная.

В блоке `OPT PSEUDOSECTION` видно, что используется расширение `EDNS(0)` версии 0, без дополнительных флагов, а максимальный размер UDP-пакета — 1232 байта. В секции `QUESTION` повторяется сам запрос: домен `www.yandex.ru` и тип записи `A` (IPv4-адрес).

В `ANSWER SECTION` приходят три результата: `www.yandex.ru` сопоставлен с IP-адресами `77.88.55.88`, `77.88.44.55` и `5.255.255.77`, каждая запись имеет `TTL 600` секунд (10 минут), то есть столько времени она может храниться в кэше.

В дополнительной информации указывается, что запрос занял 329 мс, ответ пришёл от DNS-сервера `213.186.33.99` (порт 53, UDP). Также зафиксировано точное время выполнения запроса — 11 сентября 2025 года, 11:10:30 (UTC). Размер полученного сообщения составил 90 байт. (рис. 4.3).

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

; <<> DiG 9.18.33 <<> www.yandex.ru
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48216
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;www.yandex.ru.                IN      A

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

;; Query time: 329 msec
;; SERVER: 213.186.33.99#53(213.186.33.99) (UDP)
;; WHEN: Thu Sep 11 11:10:30 UTC 2025
;; MSG SIZE rcvd: 90

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

Рисунок 4.3: Утилита dig

Открываю файл /etc/resolv.conf.

Файл /etc/resolv.conf задаёт настройки DNS для системы: в нём указываются адреса DNS-серверов (nameserver), к которым компьютер обращается для преобразования доменных имён в IP-адреса (по порядку, пока один не ответит), и домен поиска (search antoychubekova.net), который автоматически добавляется к коротким именам хостов. (рис. 4.4).

```
GNU nano 8.1                                resolv.conf
# Generated by NetworkManager
search antoychubekova.net
nameserver 213.186.33.99
nameserver 91.239.100.100
nameserver 192.168.150.36
```

Рисунок 4.4: Файл /etc/resolv.conf

Открываю файл named.conf.

Этот файл named.conf — конфигурация DNS-сервера BIND. В нём указано, что

сервер слушает запросы только на локальных адресах 127.0.0.1 (IPv4) и ::1 (IPv6), то есть работает как локальный кэширующий резолвер. Заданы служебные файлы: директория для работы /var/named, файлы для дампов кэша, статистики, памяти и ключей, а также файл для рекурсивных запросов. В настройке allow-query { localhost; }; указано, что отвечать на DNS-запросы сервер будет только самому себе (локальной машине). Включена опция recursion yes;, что делает сервер рекурсивным кэширующим DNS, пригодным для локального разрешения имён, и включена проверка DNSSEC (dnssec-validation yes;) для дополнительной безопасности. (рис. 4.5).

A screenshot of a terminal window showing the contents of the /etc/named.conf file. The terminal title is 'GNU nano 8.1 named.conf'. The file content includes comments about the Red Hat bind package, the 'options' block with settings for listening on 127.0.0.1 and ::1, file paths for cache, statistics, and recursion, and the 'recursion yes;' and 'dnssec-validation yes;' settings. A multi-line comment explains the importance of enabling recursion for a caching DNS server and the risks of not limiting queries.

```
GNU nano 8.1 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";
    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.recursing";
    allow-query { localhost; };

    /*
     - 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 will
       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 yes;
```

Рисунок 4.5: Файл named.conf

Открываю файл named.ca.

Этот файл named.ca содержит список корневых DNS-серверов Интернета (root servers), необходимых для инициализации работы DNS-сервера BIND: он задаёт, какие серверы считаются корневыми и по каким IP-адресам (IPv4 и IPv6) их можно достичь. Благодаря этому DNS-сервер знает, с чего начинать поиск доменов в

глобальной сети. (рис. 4.6).

```
GNU nano 8.1 named.ca
; This file holds the information on root name servers needed to
; initialize cache of Internet domain name servers
; (e.g. reference this file in the "cache . <file>"
; configuration file of BIND domain name servers).
;
; This file is made available by InterNIC
; under anonymous FTP as
;   file             /domain/named.cache
;   on server        FTP.INTERNIC.NET
; -OR-              RS.INTERNIC.NET
;
; last update:      December 20, 2023
; related version of root zone: 2023122001
;
; FORMERLY NS.INTERNIC.NET
;
.                 3600000      NS      A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 3600000      A       198.41.0.4
A.ROOT-SERVERS.NET. 3600000      AAAA    2001:503:ba3e::2:30
;
; FORMERLY NS1.ISI.EDU
;
.                 3600000      NS      B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET. 3600000      A       170.247.170.2
B.ROOT-SERVERS.NET. 3600000      AAAA    2801:1b8:10::b
;
; FORMERLY C.PSI.NET
;
.                 3600000      NS      C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET. 3600000      A       192.33.4.12
C.ROOT-SERVERS.NET. 3600000      AAAA    2001:500:2::c
;
; FORMERLY TERP.UMD.EDU
```

Рисунок 4.6: Файл named.ca

Открываю файл named.localhost.

Этот файл named.localhost описывает зону DNS для домена localhost. В нём задаётся основная запись SOA (Start of Authority) с параметрами обновления зоны и фиктивным адресом администратора (rname.invalid.), а также указываются записи: NS (сервер имён — сам localhost), A (IPv4-адрес 127.0.0.1) и AAAA (IPv6-адрес ::1). Таким образом, он нужен для того, чтобы DNS-сервер BIND правильно обрабатывал запросы к имени localhost и всегда резолвил его в локальный адрес. (рис. 4.7).

```

GNU nano 8.1                                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

```

Рисунок 4.7: Файл named.localhost

Открываю файл named.loopback.

Этот файл named.loopback описывает обратную (reverse) DNS-зону для интерфейса loopback. В нём задаётся запись SOA (Start of Authority) с параметрами зоны, указывается, что сервер имён (NS) — это сам localhost, и определяются адреса: A (127.0.0.1 для IPv4) и AAAA (:::1 для IPv6). Дополнительно здесь есть запись PTR, которая обеспечивает обратное преобразование IP-адреса 127.0.0.1 в имя localhost. То есть этот файл нужен, чтобы при обратных DNS-запросах (по IP) адрес loopback корректно резолвился в localhost. (рис. 4.8).

```

GNU nano 8.1                                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.

```

Рисунок 4.8: Файл named.loopback

Запускаю DNS-сервер. Включаю запуск днс-сервера в авозапуск при загрузке системы. (рис. 4.9).

```
[root@server.antoychubekova.net named]# systemctl start named
[root@server.antoychubekova.net named]# systemctl enable named
Created symlink '/etc/systemd/system/multi-user.target.wants/named.service' → '/usr/lib/systemd/system/named.service'
[root@server.antoychubekova.net named]# █
```

Рисунок 4.9: Запуск DNS-сервера

Ввожу в команду `dig 127.0.0.1 www.yandex.ru`. `dig www.yandex.ru` показывает работу с внешним DNS (сразу из `resolv.conf`), а `dig 127.0.0.1 www.yandex.ru` — проверку твоего локального BIND как резолвера. В твоём случае оба возвращают одинаковые IP-адреса Яндекса, но второй запрос идёт через твой сервер, а не напрямую к провайдерскому DNS. (рис. 4.10).

```
[root@server.antoychubekova.net named]# dig @127.0.0.1 www.yandex.ru
;; communications error to 127.0.0.1#53: timed out

; <<>> DiG 9.18.33 <<>> @127.0.0.1 www.yandex.ru
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 12025
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1232
;; COOKIE: ead5cdf1d7fe44f90100000068c2b2bbf3d09cecb03c8e2c (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: 3928 msec
;; SERVER: 127.0.0.1#53(127.0.0.1) (UDP)
;; WHEN: Thu Sep 11 11:30:03 UTC 2025
;; MSG SIZE rcvd: 118

[root@server.antoychubekova.net named]# █
```

Рисунок 4.10: Вывод `ig 127.0.0.1 www.yandex.ru`.

Делаю DNS-сервер сервером по умолчанию для хоста `server` и внутренней виртуальной сети. Для этого изменяю 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`. (рис. 4.11).


```
[root@server.antoychubekova.net named]# 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' (621f064b-e62f-4ddf-bfe7-fb21098ed01c) successfully updated.
nmcli> quit
[root@server.antoychubekova.net named]#
```

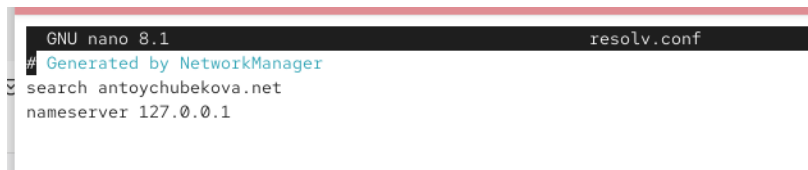
Рисунок 4.11: DNS-сервер сервером по умолчанию для хоста server и внутренней виртуальной сети

Делаю тоже самое для соединения System eth0. Оно у нас не активно. (рис. 4.12).

```
[root@server.antoychubekova.net named]# nmcli connection edit System\ eth0
Error: Unknown connection 'System eth0'.
[root@server.antoychubekova.net named]#
```

Рисунок 4.12: DNS-сервер сервером по умолчанию для хоста server и внутренней виртуальной сети

Перезапускаю NetworkManager. Проверяю наличие изменений в файле /etc/resolv.conf. (рис. 4.13).



```
GNU nano 8.1 resolv.conf
# Generated by NetworkManager
search antoychubekova.net
nameserver 127.0.0.1
```

Рисунок 4.13: Перезапуск NetworkManager

Настраиваю направление DNS-запросов от всех узлов внутренней сети, включая запросы от узла server. Для этого вношу изменения в файл /etc/named.conf, заменив строку:

listen-on port 53 { 127.0.0.1; };

на

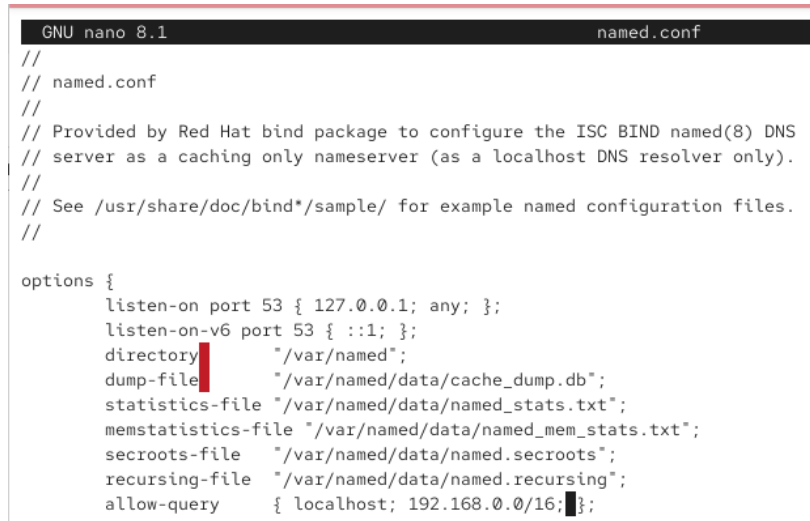
listen-on port 53 { 127.0.0.1; any; };

и строку

allow-query { localhost; };

на

allow-query { localhost; 192.168.0.0/16; }; (рис. 4.14).

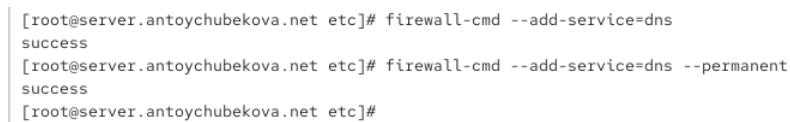


```
GNU nano 8.1 named.conf
//
// named.conf
//
// Provided by Red Hat bind package to configure the ISC BIND named(8) DNS
// server as a caching only nameserver (as a localhost DNS resolver only).
//
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//

options {
    listen-on port 53 { 127.0.0.1; 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.recursing";
    allow-query { localhost; 192.168.0.0/16; };
```

Рисунок 4.14: Настройка направление DNS-запросов

Вношу изменение в настройки межсетевого экрана узла server, разрешив работу с DNS. (рис. 4.15).



```
[root@server.antoychubekova.net etc]# firewall-cmd --add-service=dns
success
[root@server.antoychubekova.net etc]# firewall-cmd --add-service=dns --permanent
success
[root@server.antoychubekova.net etc]#
```

Рисунок 4.15: Внесение изменений

Введя команду `lsof | grep UDP` мы убедились что DNS-запросы идут через узел server, который прослушивает порт 53. (рис. 4.16).

```

n
named 15616 15621 isc-timer named 32u IPv6 85714 0t0 UDP localhost:domai
n
named 15616 15682 isc-net-0 named 25u IPv4 86199 0t0 UDP localhost:domai
n
named 15616 15682 isc-net-0 named 26u IPv4 86200 0t0 UDP localhost:domai
n
named 15616 15682 isc-net-0 named 31u IPv6 85713 0t0 UDP localhost:domai
n
named 15616 15682 isc-net-0 named 32u IPv6 85714 0t0 UDP localhost:domai
n
named 15616 15683 isc-net-0 named 25u IPv4 86199 0t0 UDP localhost:domai
n
named 15616 15683 isc-net-0 named 26u IPv4 86200 0t0 UDP localhost:domai
n
named 15616 15683 isc-net-0 named 31u IPv6 85713 0t0 UDP localhost:domai
n
named 15616 15683 isc-net-0 named 32u IPv6 85714 0t0 UDP localhost:domai
NetworkMa 19387 root 31u IPv4 139905 0t0 UDP server.antoychu
bekova.net:bootpc->_gateway:bootps root 31u IPv4 139905 0t0 UDP server.antoychu
NetworkMa 19387 19389 gmain root 31u IPv4 139905 0t0 UDP server.antoychu
bekova.net:bootpc->_gateway:bootps root 31u IPv4 139905 0t0 UDP server.antoychu
NetworkMa 19387 19390 pool-spaw root 31u IPv4 139905 0t0 UDP server.antoychu
bekova.net:bootpc->_gateway:bootps root 31u IPv4 139905 0t0 UDP server.antoychu
NetworkMa 19387 19391 gdbus root 31u IPv4 139905 0t0 UDP server.antoychu
bekova.net:bootpc->_gateway:bootps
[root@server.antoychubekova.net etc]#

```

Рисунок 4.16: Проверка DNS-запросов

В случае возникновения в сети ситуаций, когда DNS-запросы от сервера фильтруются сетевым оборудованием, следует добавить перенаправление DNS-запросов на конкретный вышестоящий DNS-сервер. Для этого в конфигурационном файле `named.conf` в секции `option` следует добавить:

```
forwarders { список DNS-серверов };
```

```
forward first;
```

Возможно вышестоящий DNS-сервер может не поддерживать технологию `dnssec`, из за этого в конфигурационном файлике указываю следующие настройки:

```
dnssec-enable no;
```

```
dnssec-validation no; (рис. 4.17).
```

```
GNU nano 8.1 named.conf
//
options {
    forwarders { 127.0.0.1 }
    forward first;
    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.recursing";
    allow-query { localhost; 192.168.0.0/16; };

    /*
     - 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 will
       cause your server to become part of large scale DNS amplification
       attacks. Implementing BCP38 within your network would greatly
       reduce such attack surface
    */

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

Рисунок 4.17: Перенаправление DNS-запросов

Копирую шаблон описания DNS-зон `named.rfc1912.zone` из каталога `/etc` в каталог `/etc/named` и переименовываю его в `antoychubekova.net`. (рис. 4.18).

```
[root@server.antoychubekova.net etc]# cp /etc/named.rfc1912.zones /etc/named/
[root@server.antoychubekova.net etc]# cd /etc/named
[root@server.antoychubekova.net named]# mv /etc/named/named.rfc1912.zones /etc/named/antoychubekova.net
[root@server.antoychubekova.net named]#
```

Рисунок 4.18: Редактирование файлов

Включаю файл описания зоны `/etc/named/antoychubekova.net` в конфигурационном файле DNS `/etc/named.conf`, добавив в нём в конце строку:

`include «/etc/named/antoychubekova.net»;` (рис. 4.19).

```

zone "." IN {
    type hint;
    file "named.ca";
};

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

```

```

^G Help      ^O Write Out  ^F Where Is   ^_
^X Exit      ^R Read File  ^\ Replace    ^

```

Рисунок 4.19: Включение файла описания зоны /etc/named/antoychubekova.net

Открываю файл /etc/named/antoychubekova.net на редактирование и записываю:

```

zone «antoychubekova.net» IN {
type master;
file «master/fz/antoychubekova.net»;
allow-update { none; };
};
zone «1.168.192.in-addr.arpa» IN {
type master;
file «master/rz/192.168.1»;
allow-update { none; };
}; (рис. 4.20).

```



```
GNU nano 8.1 antoychubekova.net

zone "antoychubekova.net" IN {
    type master;
    file "master/fz/antoychubekova.net";
    allow-update { none; };
};

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

Рисунок 4.20: Редактирование /etc/named/antoychubekova.net

В каталоге /var/named создаю подкаталоги master/fz и master/rz, в которых будут располагаться файлы прямой и обратной зоны соответственно. (рис. 4.21).

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

Рисунок 4.21: Редактирование /var/named

Копирую шаблон прямой DNS-зоны named.localhost из каталога /var/named в каталог /var/named/master/fz и переименовываю его в antoychubekova.net. (рис. 4.22).

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

Рисунок 4.22: Копирование шаблон прямой DNS-зоны

Изменяю файл /var/named/master/fz/antoychubekova.net, указав необходимые DNS-записи для прямой зоны. В этом файле DNS-имя сервера @ rname.invalid. заменяю на @ server.antoychubekova.net. ; формат серийного номера ГТГТММДДВВ (2025091100); адрес в А-записи заменяю с 127.0.0.1 на 192.168.1.1; в директиве \$ORIGIN задаю текущее имя домена antoychubekova.net., а затем указываю имена и адреса серверов в этом домене в виде А-записей DNS. (рис. 4.23).

```
GNU nano 8.1 antoychubekova.net
$TTL 1D
@      IN SOA  @ server.antoychubekova.net (
                                2025091100      ; serial
                                1D                ; refresh
                                1H                ; retry
                                1W                ; expire
                                3H )             ; minimum

NS      @
A       192.168.1.1
$ORIGIN antoychubekova.net.
server  A       192.168.1.1
ns      A       192.168.1.1
```

Рисунок 4.23: Редактирование /var/named/master/fz/antoychubekova.net

Копирую шаблон обратной DNS-зоны named.loopback из каталога /var/named в каталог /var/named/master/rz и переименовываю его в 192.168.1 (рис. 4.24).

```
[root@server.antoychubekova.net fz]# cp /var/named/named.loopback /var/named/master/rz/
[root@server.antoychubekova.net fz]# cd /var/named/master/rz
[root@server.antoychubekova.net rz]# mv named.loopback 192.168.1
[root@server.antoychubekova.net rz]#
```

Рисунок 4.24: Копирование шаблон обратной DNS-зоны named.loopback

Изменяю файл /var/named/master/rz/192.168.1, указав необходимые DNS-записи для обратной зоны. В этом файле DNS-имя сервера @ rname.invalid. заменяю на @ server.user.net. ; формат серийного номера ГГГГММДДВВ (2025091100); адрес в А-записи заменяю с 127.0.0.1 на 192.168.1.1; в директиве \$ORIGIN задаю название обратной зоны в виде 1.168.192.in-addr.arpa., затем задаю PTR-записи. (рис. 4.25).

```

GNU nano 8.1                                192.168.1
$TTL 1D
@      IN SOA  @ server.antoychubekova.net. (
                                2025091100      ; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H )   ; minimum

      NS      @
      A      192.168.1.1
      PTR     server.antoychubekova.net.
$ORIGIN 1.168.192.in-addr.arpa.
1      PTR     server.antoychubekova.net,
1      PTR     ns.antoychubekova.net.

```

Рисунок 4.25: Редактирование /var/named/master/rz/192.168.1

Далее исправляю права доступа к файлам в каталогах /etc/named и /var/named, чтобы демон named мог с ними работать. (рис. 4.26).

```

[root@server.antoychubekova.net rz]# chown -R named:named /etc/named
[root@server.antoychubekova.net rz]# chown -R named:named /var/named
[root@server.antoychubekova.net rz]#

```

Рисунок 4.26: Изменение прав доступа к /etc/named и /var/named

В системах с запущенным SELinux все процессы и файлы имеют специальные метки безопасности (так называемый «контекст безопасности»), используемые системой для принятия решений по доступу к этим процессам и файлам. После изменения доступа к конфигурационным файлам named корректно восстанавливаю их метки в SELinux. (рис. 4.27).

```

[root@server.antoychubekova.net rz]# restorecon -vR /etc
Relabeled /etc/lvm/devices/system.devices from system_u:object_r:lvm_metadata_t:s0 to system_u:object_r:lvm_etc_t:s0
Relabeled /etc/lvm/devices/backup/system.devices-20250911.072347.0005 from system_u:object_r:lvm_metadata_t:s0 to system_u:object_r:lvm_etc_t:s0
Relabeled /etc/NetworkManager/system-connections/eth1.nmconnection from unconfined_u:object_r:user_tmp_t:s0 to unconfined_u:object_r:NetworkManager_etc_rw_t:s0
[root@server.antoychubekova.net rz]# restorecon -vR /var/named
[root@server.antoychubekova.net rz]#

```

Рисунок 4.27: Корректное восстановление меток в SELinux

Для проверки состояния переключателей SELinux, относящихся к named, ввожу: `getsebool -a | grep named`. (рис. 4.28).


```
[root@server.antoychubekova.net rz]# getsebool -a | grep named
named_tcp_bind_http_port --> off
named_write_master_zones --> on
[root@server.antoychubekova.net rz]#
```

Рисунок 4.28: Состояния переключателей SELinux

Даю named разрешение на запись в файлы DNS-зоны. (рис. 4.29).

```
[root@server.antoychubekova.net rz]# setsebool named_write_master_zones 1
[root@server.antoychubekova.net rz]# setsebool -P named_write_master_zones 1
[root@server.antoychubekova.net rz]#
```

Рисунок 4.29: Предоставление разрешения на запись в файлы DNS-зоны

В дополнительном терминале запускаю в режиме реального времени расширенный лог системных сообщений, чтобы проверить корректность работы системы. Мы видим, что корректно обрабатывается. (рис. 4.30).

```
Subject: Process 16643 (VBoxClient) dumped core
Defined-By: systemd
Support: https://wiki.rockylinux.org/rocky/support
Documentation: man:core(5)

Process 16643 (VBoxClient) crashed and dumped core.

This usually indicates a programming error in the crashing program and
should be reported to its vendor as a bug.
Sep 12 04:39:44 server.antoychubekova.net systemd[1]: systemd-coredump@785-16647-0.service: Deactivat
ed successfully.
Subject: Unit succeeded
Defined-By: systemd
Support: https://wiki.rockylinux.org/rocky/support

The unit systemd-coredump@785-16647-0.service has successfully entered the 'dead' state.
```

Рисунок 4.30: Проверка корректности работы системы

В первом терминале перезапускаю DNS-сервер. При помощи утилиты dig получаю описание DNS-зоны с сервера ns.antoychubekova.net. В выводе видно, что запрос успешно обработан (status: NOERROR) и возвращён авторитетный ответ (aa — authoritative answer) от вашего локального DNS-сервера на 127.0.0.1. В разделе ответа указано, что ns.antoychubekova.net имеет A-запись с адресом 192.168.1.1, TTL установлен на 86400 секунд (1 день). Время обработки запроса составило 2 мс, что подтверждает, что локальный сервер функционирует корректно и возвращает IP для указанного имени. (рис. 4.31).

```

[root@server.antoychubekova.net ~]# systemctl restart named
[root@server.antoychubekova.net ~]# dig ns.antoychubekova.net

; <<>> DiG 9.18.33 <<>> ns.antoychubekova.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 24621
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1232
;; COOKIE: 72ebbb984546434c0100000068c39604535d2a9db155c30d (good)
;; QUESTION SECTION:
;ns.antoychubekova.net.          IN      A

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

;; Query time: 2 msec
;; SERVER: 127.0.0.1#53(127.0.0.1) (UDP)
;; WHEN: Fri Sep 12 03:39:48 UTC 2025
;; MSG SIZE rcvd: 94

```

Рисунок 4.31: Описание DNS-зоны с сервера ns.antoychubekova.net

При помощи утилиты host проанализирую корректность работы DNS-сервера.

В выводе команды `host -l antoychubekova.net` видно, что указаны следующие записи: основной сервер имени antoychubekova.net и его IP 192.168.1.1, а также дополнительные записи ns.antoychubekova.net и server.antoychubekova.net, все с тем же IP 192.168.1.1. Это говорит о том, что локальный DNS-сервер правильно отвечает на запросы и возвращает все настроенные записи зоны. (рис. 4.32).

```

[root@server.antoychubekova.net ~]# host -l antoychubekova.net
antoychubekova.net name server antoychubekova.net.
antoychubekova.net has address 192.168.1.1
ns.antoychubekova.net has address 192.168.1.1
server.antoychubekova.net has address 192.168.1.1
[root@server.antoychubekova.net ~]# █

```

Рисунок 4.32: Корректность работы DNS-сервера

Команда `host a antoychubekova.net`, которая по сути запрашивает все записи (ANY) для домена antoychubekova.net. В выводе видно, что сервер вернул статус NOERROR, что означает успешное выполнение запроса. В разделе ответа содержатся три записи: SOA (Start of Authority) с корректными параметрами зоны, NS-запись, указывающая

на сам домен как сервер имён, и А запись с IP 192.168.1.1. Время ответа составило 8 мс, и ответ получен от локального сервера 127.0.0.1. Это говорит о том, что локальный DNS-сервер BIND работает корректно. (рис. 4.33).

```
[root@server.antoychubekova.net ~]# host -a antoychubekova.net
Trying "antoychubekova.net"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 50421
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;antoychubekova.net.          IN      ANY

;; ANSWER SECTION:
antoychubekova.net.  86400  IN      SOA      antoychubekova.net. server.an
toychubekova.net. 2025091100 86400 3600 604800 10800
antoychubekova.net.  86400  IN      NS       antoychubekova.net.
antoychubekova.net.  86400  IN      A        192.168.1.1

Received 109 bytes from 127.0.0.1#53 in 8 ms
[root@server.antoychubekova.net ~]#
```

Рисунок 4.33: Корректность работы DNS-сервера

Команда `host -t A antoychubekova.net` возвращает, что `antoychubekova.net` имеет адрес 192.168.1.1. Значит все корректно отрабатывается. (рис. 4.34).

```
Received 109 bytes from 127.0.0.1#53 in 8 ms
[root@server.antoychubekova.net ~]# host -t A antoychubekova.net
antoychubekova.net has address 192.168.1.1
[root@server.antoychubekova.net ~]#
```

Рисунок 4.34: Корректность работы DNS-сервера

В ответе сервер возвращает две PTR-записи: `ns.antoychubekova.net` и `server.antoychubekova.net`. Технически это означает, что обратное разрешение работает — IP связан с доменными именами. (рис. 4.35).

```
[root@server.antoychubekova.net ~]# host -t PTR 192.168.1.1
1.1.168.192.in-addr.arpa domain name pointer ns.antoychubekova.net.
1.1.168.192.in-addr.arpa domain name pointer server.antoychubekova.net.
[root@server.antoychubekova.net ~]#
```

Рисунок 4.35: Корректность работы DNS-сервера

На виртуальной машине `server` перехожу в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/`, и создаю в нём каталог

dns, в который помещаю в соответствующие каталоги конфигурационные файлы DNS. (рис. 4.36).

```
[root@server.antoychubekova.net ~]# cd /vagrant
[root@server.antoychubekova.net vagrant]# mkdir -p /vagrant/provision/server/dns/etc/named
[root@server.antoychubekova.net vagrant]# mkdir -p /vagrant/provision/server/dns/var/named/master/

[root@server.antoychubekova.net vagrant]# cp -R /etc/named.conf /vagrant/provision/server/dns/etc/
[root@server.antoychubekova.net vagrant]# cp -R /etc/named/* /vagrant/provision/server/dns/etc/named/

[root@server.antoychubekova.net vagrant]# cp -R /var/named/master/* /vagrant/provision/server/dns/var/
/named/master/
[root@server.antoychubekova.net vagrant]# █
```

Рисунок 4.36: Редактирование конфигурационных файлов DNS

В каталоге /vagrant/provision/server создаю исполняемый файл dns.sh. (рис. 4.37).

```
[root@server.antoychubekova.net vagrant]# cd provision
[root@server.antoychubekova.net provision]# cd server
[root@server.antoychubekova.net server]# touch dns.sh
[root@server.antoychubekova.net server]# chmod +x dns.sh
[root@server.antoychubekova.net server]#
```

Рисунок 4.37: Создание исполняемого файла dns.sh

Открыв его на редактирование, пропишите в нём скрипт, по сути, повторяющий произведённые выше действия по установке и настройке DNS-сервера: подставляет в нужные каталоги подготовленные конфигурационные файлы; меняет соответствующим образом права доступа, метки безопасности SELinux и правила межсетевого экрана; настраивает сетевое соединение так, чтобы сервер выступал DNS-сервером по умолчанию для узлов внутренней виртуальной сети; запускает DNS-сервер. (рис. 4.38).

```

GNU nano 8.1 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_wrire_master_zones 1

echo "Change dns srver address"
nmcli connection edit "System eth0" <<EOF
remove ipv4.dns
set ipv4.ignore-auto-dns yes
seet ipv4.dns 127.0.0.1
save
quit

```

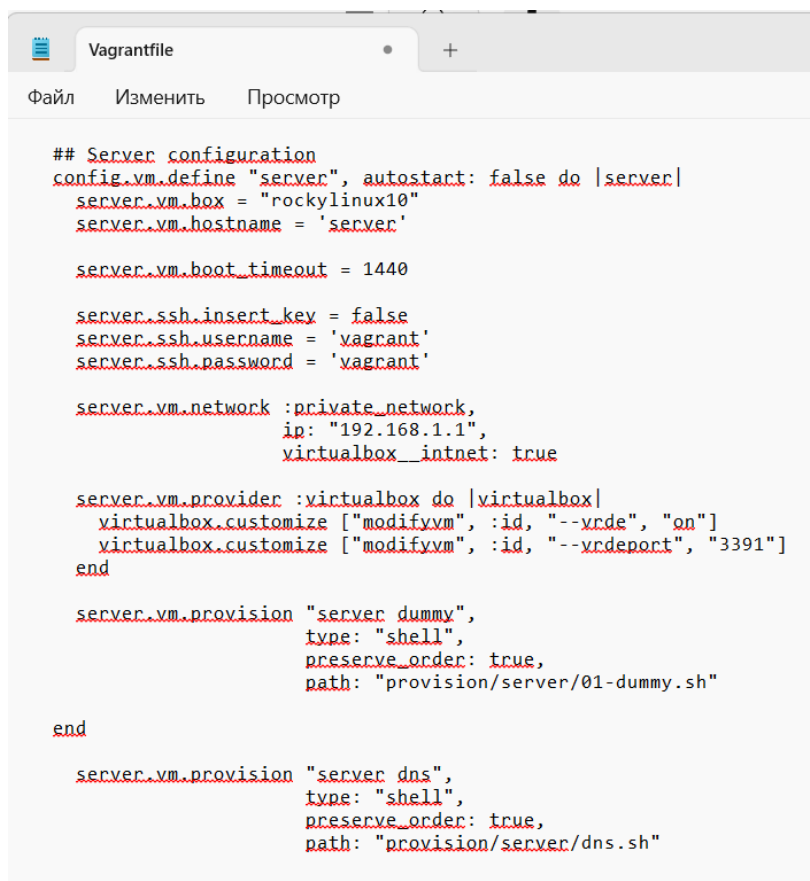
Рисунок 4.38: Редактирование исполняемого файла dns.sh

Для отработки созданного скрипта во время загрузки виртуальной машины server в конфигурационном файле Vagrantfile добавляю в разделе конфигурации для сервера:

```

server.vm.provision «server dns»,
type: «shell»,
preserve_order: true,
path: «provision/server/dns.sh» (рис. 4.39).

```



The image shows a code editor window titled 'Vagrantfile'. The editor has a menu bar with 'Файл', 'Изменить', and 'Просмотр'. The code is written in Ruby and defines a virtual machine named 'server'. The configuration includes the VM box name ('rockylinux10'), hostname ('server'), boot timeout (1440), SSH settings (username 'vagrant', password 'vagrant'), network settings (private network, IP '192.168.1.1', virtualbox____intnet: true), provider ('virtualbox'), and two provision scripts: 'server dummy' and 'server dns'.

```
## Server configuration
config.vm.define "server", autostart: false do |server|
  server.vm.box = "rockylinux10"
  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.provider :virtualbox do |virtualbox|
    virtualbox.customize ["modifyvm", :id, "--vrdm", "on"]
    virtualbox.customize ["modifyvm", :id, "--vrdmport", "3391"]
  end

  server.vm.provision "server dummy",
    type: "shell",
    preserve_order: true,
    path: "provision/server/01-dummy.sh"

end

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

Рисунок 4.39: Редактирование Vagrantfile

5 Выводы

В ходе выполнения лабораторной работы №2 я приобрела практические навыки по установке и конфигурированию DNS-сервера, усвоила принцип работы системы доменных имён.

Список литературы

1. Barr D. Common DNS Operational and Configuration Errors: RFC / RFC Editor. —02/1996. — DOI: 10.17487/rfc1912.
2. Security-Enhanced Linux. Linux с улучшенной безопасностью: руководство пользователя / М. McAllister, S. Radvan, D. Walsh, D. Grift, E. Paris, J. Morris. — URL: https://docs-old.fedoraproject.org/ru-RU/Fedora/13/html/Security-Enhanced_Linux/index.html (дата обр.13.09.2021).
3. Systemd. — 2015. — URL: <https://wiki.archlinux.org/index.php/Systemd> (visited on 09/13/2021).
4. Костромин В. А. Утилита lsof — инструмент администратора. — URL: <http://ruslinux.net/kos.php?name=/papers/lsof/lsof.html> (дата обр. 13.09.2021).
5. Поттеринг Л. Systemd для администраторов: цикл статей. — 2010. — URL: <http://wiki.opennet.ru/Systemd> (дата обр. 13.09.2021).
6. Сайт проекта NetworkManager. — URL: <https://wiki.gnome.org/Projects/NetworkManager> (visited on 09/13/2021).
7. Сайт проекта nmcli. — URL: <https://developer.gnome.org/NetworkManager/stable/nmcli.html> (visited on 09/13/2021).