Лабораторная работа №2

Настройка DNS-сервера BIND (прямая и обратная зона)

Владимир Базлов

10 ноября 2025

Российский университет дружбы народов, Москва, Россия

Цель работы

Основная цель

Получить практические навыки настройки **DNS-сервера BIND**, создания прямой и обратной зон
и проверки работы DNS-разрешения в локальной сети.

Что использовалось

Инструменты

- **Rocky Linux** – серверная ОС
- **BIND (named)** – DNS-сервер
- **Vagrant + VirtualBox** – виртуальная лаборатория
- **nmcli, dig, host** – сетевые утилиты

Анализ конфигурации

Основные конфигурационные файлы

- /etc/resolv.conf
- /etc/named.conf
- /var/named/named.ca
- /var/named/named.localhost

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

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

Проверка работы хоста как DNS-клиента

`dig www.yandex.ru`

→ запрос уходит на внешний DNS

`dig @127.0.0.1 www.yandex.ru`

→ ответ получает локальный сервер BIND

```
[root@server.vabazlov.net ~]# systemctl start named
[root@server.vabazlov.net ~]# systemctl enable named
Created symlink '/etc/systemd/system/multi-user.target.wants/named.service' → '/usr/lib/systemd/system/named.service'.
[root@server.vabazlov.net ~]# dig @127.0.0.1 www.yandex.ru
;; communications error to 127.0.0.1#53: timed out
;; 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: 25242
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 01839f3071098a87010000006911ec3f2ae95ffd03233598 (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       77.88.55.88
www.yandex.ru.          600     IN      A       77.88.44.55

;; Query time: 1315 msec
```

Настройка локального DNS

Назначение сервера DNS по умолчанию

Интерфейс `eth0` настроен на использование `127.0.0.1`

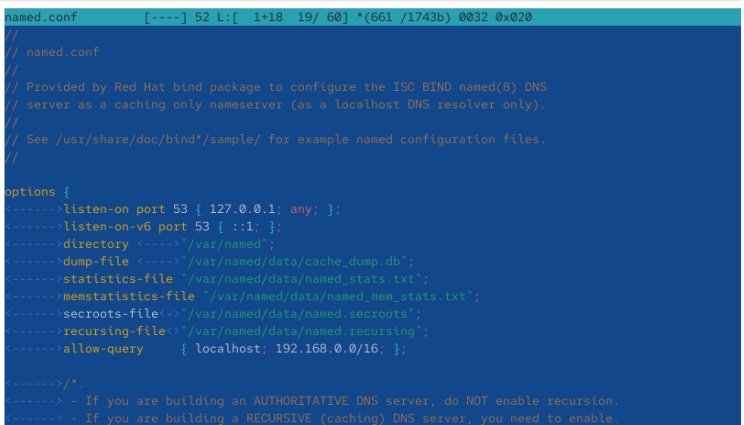
```
[root@server.vabazlov.net ~]#
[root@server.vabazlov.net ~]# nmcli connection edit eth0

===| 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, hostna
me, 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' (e292e83a-7750-4087-b4e1-a998fc55c0ea) successfully updated.
nmcli> quit
[root@server.vabazlov.net ~]# systemctl restart NetworkManager
[root@server.vabazlov.net ~]#
[root@server.vabazlov.net ~]# cat /etc/resolv.conf
# Generated by NetworkManager
search vabazlov.net
nameserver 127.0.0.1
[root@server.vabazlov.net ~]#
```

Рис. 3: Настройка DNS через nmcli

Разрешение запросов от внутренней сети

В `named.conf` изменено:

- `listen-on port 53 { any; };`
- `allow-query { localhost; 192.168.0.0/16; };`



Создание прямой и обратной зоны

Создание файла зон

```
vabazlov.net       [----]  0 L:[  1+27  28/ 29] *(699 / 700b) 0010 0x00A
// named.rfc1912.zones:
//
// Provided by Red Hat caching-nameserver package.
//
// ISC BIND named zone configuration for zones recommended by
// RFC 1912 section 4.1 : localhost TLDs and address zones
// and https://tools.ietf.org/html/rfc6303
// (c)2007 R W Franks
//.
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//
// Note: empty-zones-enable yes; option is default.
// If private ranges should be forwarded, add.
// disable-empty-zone "."; into options
//.

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

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

Прямая зона — vabazlov.net

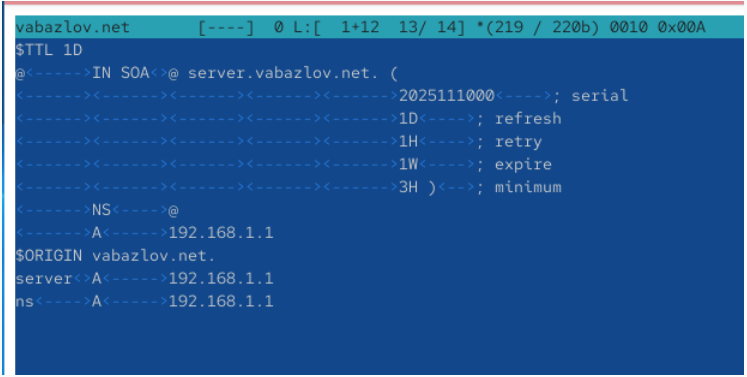


Рис. 6: Файл прямой зоны

Обратная зона — 1.168.192.in-addr.arpa

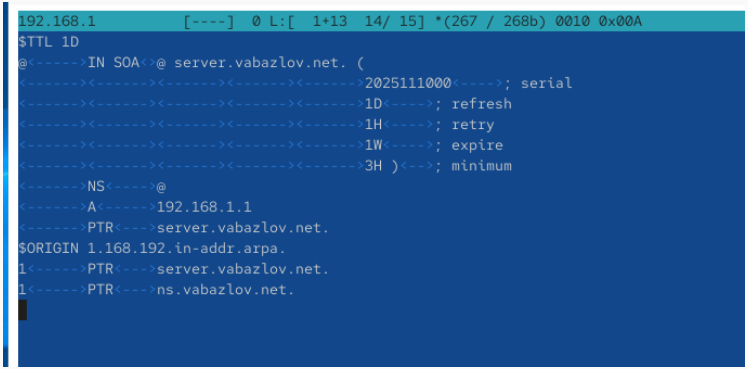


Рис. 7: Файл обратной зоны

Проверка работы DNS

Проверка A-записей и PTR-записей

```
[root@server.vabazlov.net rz]# dig ns.vabazlov.net

; <<>> DiG 9.18.33 <<>> ns.vabazlov.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54361
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: ecf04997e5304528010000006911f0d5c1558a64a43ef101 (good)
;; QUESTION SECTION:
;ns.vabazlov.net.               IN      A

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

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1) (UDP)
;; WHEN: Mon Nov 10 14:04:05 UTC 2025
;; MSG SIZE  rcvd: 88

[root@server.vabazlov.net rz]# host -l vabazlov.net
vabazlov.net name server vabazlov.net.
vabazlov.net has address 192.168.1.1
ns.vabazlov.net has address 192.168.1.1
server.vabazlov.net has address 192.168.1.1
```

Итоги работы

Вывод

В ходе выполнения лабораторной работы:

- установлен и настроен DNS-сервер **BIND**;
- созданы **прямая и обратная зоны**;
- настроен firewall и SELinux;
- выполнена проверка через `dig` и `host`;
- подготовлен **provisioning-скрипт** для автоматического развёртывания.

Настроенный DNS-сервер успешно разрешает доменные имена и может использоваться в локальной сети.