

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО

КАФЕДРА КОМПЬЮТЕРНЫХ СИСТЕМ И ПРОГРАММНЫХ ТЕХНОЛОГИЙ

Отчёт по лабораторной работе №3

Курс: «Администрирование компьютерных сетей»

Тема: «Администрирование сетевых сервисов»

Выполнил студент:

Бояркин Никита Сергеевич

Группа: 13541/3

Проверил:

Малышев Игорь Алексеевич

Санкт-Петербург
2018 г.

Содержание

| | | |
|----------|---------------------------------------------|----------|
| 1 | Лабораторная работа №3 | 2 |
| 1.1 | Цель работы | 2 |
| 1.2 | DHCP и удаленная загрузка | 2 |
| 1.2.1 | Структура сети | 2 |
| 1.2.2 | Настройка DHCP на FreeBSD | 3 |
| 1.2.3 | Настройка DHCP на Ubuntu | 3 |
| 1.2.4 | Настройка TFTP на Ubuntu | 4 |
| 1.2.5 | Тестирование | 5 |
| 1.3 | DNS сервисы | 6 |
| 1.3.1 | Структура сети | 6 |
| 1.3.2 | Настройка кэширующего DNS сервера | 6 |
| 1.3.3 | Тестирование | 7 |
| 1.3.4 | Настройка первичного DNS сервера | 8 |
| 1.3.5 | Настройка вторичного DNS сервера | 9 |
| 1.3.6 | Тестирование | 9 |
| 1.4 | Вывод | 9 |

Лабораторная работа №3

1.1 Цель работы

- Изучение состава и функциональных возможностей сетевых сервисов операционных систем.
- Разработка и настройка сервисов локальной сети.
- Разработка и настройка сервисов демилитаризованной зоны.
- Разработка и настройка сервисов пограничной зоны.

1.2 DHCP и удаленная загрузка

В данной работе используются два DHCP сервера, на FreeBSD и на Ubuntu, для двух бездисковых клиентов из разных сетей. Также на Ubuntu будет установлен TFTP сервер для удаленной загрузки ОС.

1.2.1 Структура сети

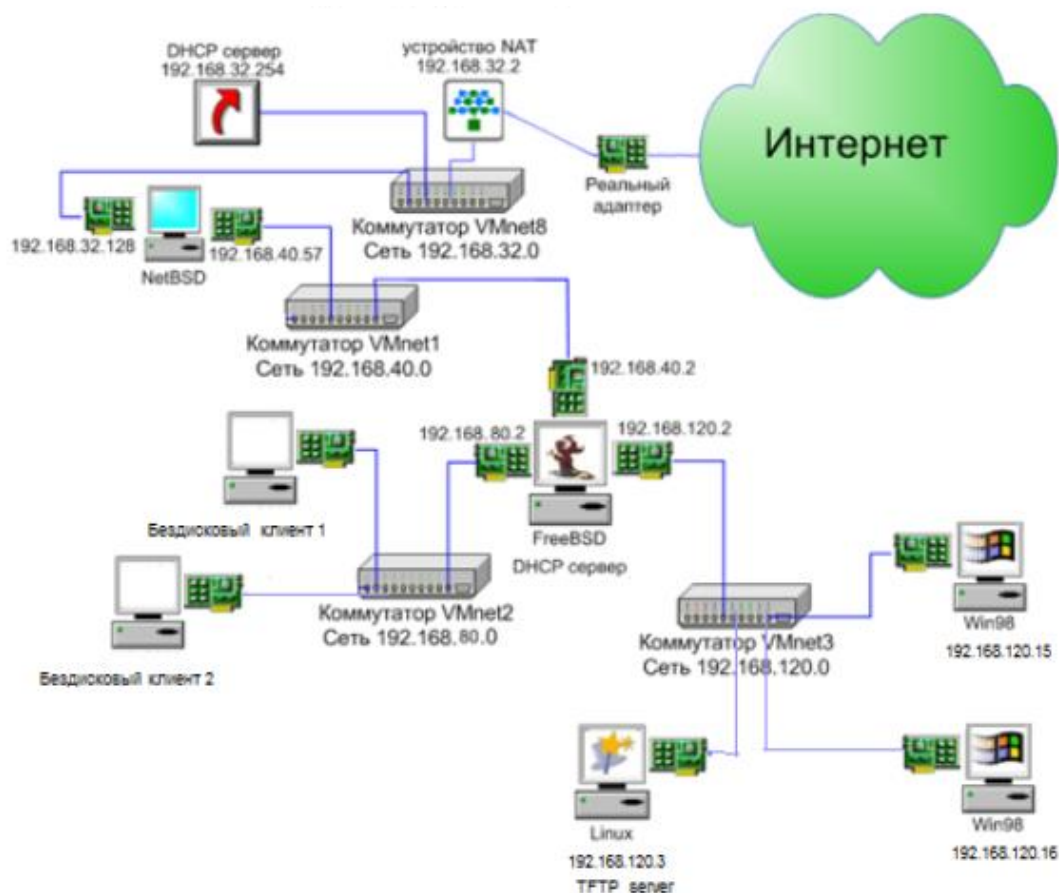


Рис. 1.1: Модифицированная структура сети

1.2.2 Настройка DHCP на FreeBSD

В первую очередь необходимо установить DHCP сервер из каталога `/usr/ports/net/isc-dhcp44-server`. После этого необходимо сконфигурировать DHCP сервер:

```
root@:~ # cat /etc/rc.conf
usbd_enable="YES"
gateway_enable="YES"
ipnat_enable="YES"
defaultrouter="192.168.40.57"
ifconfig_em1="inet 192.168.40.2 netmask 255.255.255.0"
ifconfig_em2="inet 192.168.80.2 netmask 255.255.255.0"
ifconfig_em3="inet 192.168.120.2 netmask 255.255.255.0"

dhcpcd_enable="YES"
dhcpcd_flags="-q"
dhcpcd_ifaces="em2"
```

Рис. 1.2: Конфигурация для DHCP сервера

```
root@:~ # cat /usr/local/etc/dhcpd.conf
option domain-name "example.org";
option domain-name-servers 192.168.32.2;

default-lease-time 600;
max-lease-time 7200;

subnet 192.168.80.0 netmask 255.255.255.0 {
    range 192.168.80.127 192.168.80.224;
    option routers 192.168.80.2;
    option root-path "192.168.120.3:/usr/tftpboot/";
    next-server 192.168.120.3;
    filename "gpxelinux.0";
}
```

Рис. 1.3: Конфигурация для DHCP сервера

Запуск DHCP сервера:

```
root@:~ # /usr/local/etc/rc.d/isc-dhcpd onestart
Starting dhcpd.
```

Рис. 1.4: Запуск службы DHCP

1.2.3 Настройка DHCP на Ubuntu

В первую очередь необходимо установить DHCP сервер командой:

```
1 sudo apt-get install isc-dhcp-server
```

После этого необходимо сконфигурировать DHCP сервер:

```
nikita@ubuntu:~$ cat /etc/dhcp/dhcpd.conf
option domain-name "example.org";
option domain-name-servers 192.168.32.2;

default-lease-time 600;
max-lease-time 7200;

subnet 192.168.120.0 netmask 255.255.255.0 {
    range 192.168.120.100 192.168.120.200;
    option routers 192.168.120.2;
    next-server 192.168.120.3;
    filename "gpxelinux.0";
}
```

Рис. 1.5: Конфигурация для DHCP сервера

```
nikita@ubuntu:/usr/tftpboot/pxelinux.cfg$ cat /etc/dhcp/dhcpd.conf
option domain-name "example.org";
option domain-name-servers 192.168.32.2;

default-lease-time 600;
max-lease-time 7200;

subnet 192.168.120.0 netmask 255.255.255.0 {
    range 192.168.120.100 192.168.120.200;
    option routers 192.168.120.2;
    next-server 192.168.120.3;
    filename "gpxelinux.0";
}
```

Рис. 1.6: Конфигурация для DHCP сервера

Запуск DHCP сервера:

```
nikita@ubuntu:~$ sudo /etc/init.d/isc-dhcp-server restart
[ ok ] Restarting isc-dhcp-server (via systemctl): isc-dhcp-server.service.
```

Рис. 1.7: Запуск службы DHCP

1.2.4 Настройка TFTP на Ubuntu

После установки TFTP необходимо создать каталог для удаленной загрузки:

```
nikita@ubuntu:~$ sudo mkdir -p /usr/tftpboot/images
nikita@ubuntu:~$ sudo mkdir /usr/tftpboot/pxelinux.cfg
nikita@ubuntu:~$ cat /etc/rc.conf
tftpd_enable="YES"
tftpd_flags="-p -s /usr/tftpboot -B 1024 --ipv4"
```

Рис. 1.8: Создание каталога удаленной загрузки и настройка rc.conf

После этого смонтируем образ Ubuntu в каталог удаленной загрузки:

```
nikita@ubuntu:/usr/tftpboot$ sudo mount -o loop /usr/tftpboot/ubuntu/ubuntu.iso
/usr/tftpboot/ubuntu/myubuntu/
mount: /dev/loop0 is write-protected, mounting read-only
```

Рис. 1.9: Монтирование образа в каталог удаленной загрузки

Добавим общий доступ к данному каталогу:

```
nikita@ubuntu:/usr/tftpboot$ cat /etc/exports | tail -n 1  
/usr/tftpboot/ubuntu/myubuntu *(ro, sync, no_wdelay, insecure_locks, no_root_squash,  
insecure)
```

Рис. 1.10: Внесение каталога удаленной загрузки в /etc/exports

Зададим главный конфигурационный файл TFTP:

```
nikita@ubuntu:/usr/tftpboot/pxelinux.cfg$ cat default  
default vesamenu.c32  
label Ubuntu 12  
kernel ubuntu/myubuntu/casper/vmlinuz.efi  
initrd ubuntu/myubuntu/casper/initrd.lz  
append root=/dev/nfs boot=casper netboot=nfs nfsroot=192.168.120.3:/usr/tftpboot  
/ubuntu/myubuntu
```

Рис. 1.11: Конфигурационный файл удаленной загрузки

1.2.5 Тестирование

В результате бездисковые клиенты могут автоматически подключаться к DHCP и устанавливать Ubuntu удаленно:



Рис. 1.12: Окно загрузки Ubuntu на бездисковом клиенте

1.3 DNS сервисы

Настроим кэширующий DNS сервер, а также связку из первичного и вторичного DNS серверов.

1.3.1 Структура сети

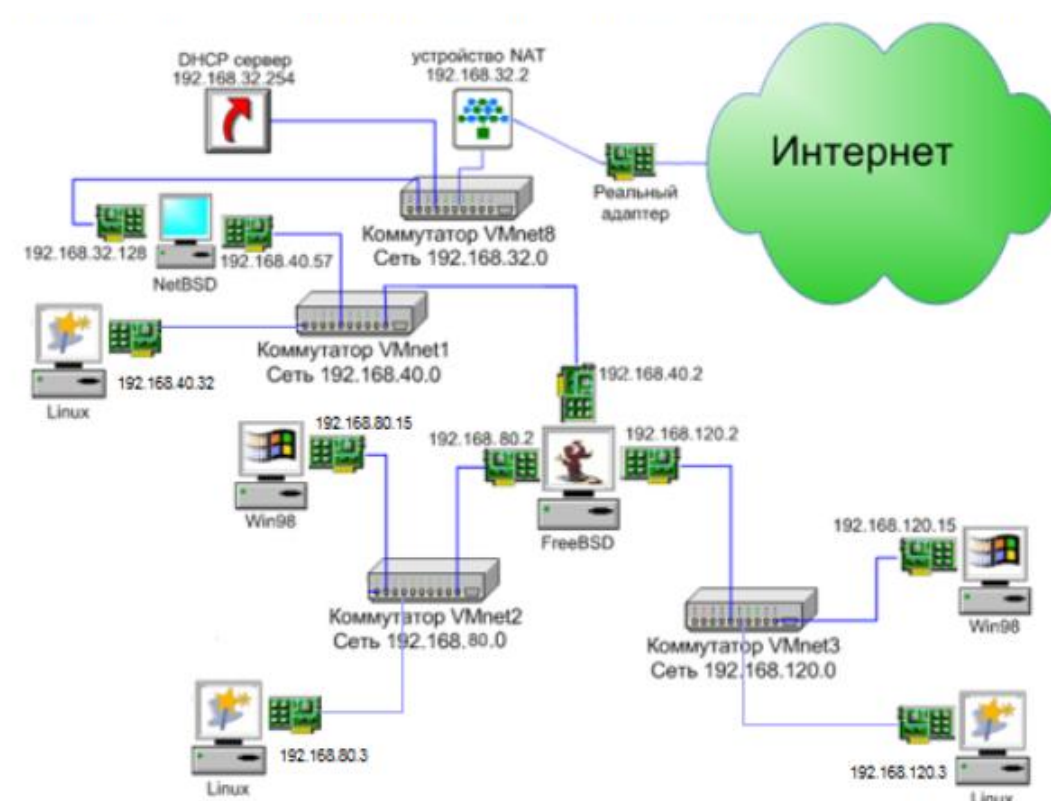


Рис. 1.13: Модифицированная структура сети

1.3.2 Настройка кэширующего DNS сервера

Настроим кэширующий DNS сервер на машине с адресом 192.168.40.32. Установим пакет bind9 для создания DNS сервера:

```
1 sudo apt-get install bind9
```

Настроим DNS сервер:

```
nikita@ubuntu:~$ cat /etc/bind/named.conf.options
options {
    directory "/var/cache/bind";

    forwarders {
        8.8.8.8;
    };

    dnssec-validation auto;

    auth-nxdomain no;    # conform to RFC1035
    listen-on-v6 { any; };
};
```

Рис. 1.14: Настройка кэширующего DNS сервера

Настроим клиент для данного сервера:

```
nikita@ubuntu:~$ cat /etc/NetworkManager/NetworkManager.conf
[main]
plugins=ifupdown,keyfile,ofono
# dns=dnsmasq

[ifupdown]
managed=false
```

Рис. 1.15: Настройка клиента для использования собственного DNS сервера

1.3.3 Тестирование

Проверим работоспособность DNS сервера с помощью утилиты nslookup:

```
nikita@ubuntu:~$ nslookup google.com
Server:          192.168.40.32
Address:         192.168.40.32#53

Non-authoritative answer:
Name:   google.com
Address: 173.194.222.100
```

Рис. 1.16: Проверка работы DNS сервера

Проверим способность к кэшированию: замерим время запроса в первый раз и во второй раз:

```
nikita@ubuntu:~$ dig google.ru

;; Query time: 59 msec
;; SERVER: 192.168.40.32#53(192.168.40.32)
;; WHEN: Mon May 21 18:47:20 PDT 2018
;; MSG SIZE rcvd: 265
```

Рис. 1.17: Первое обращение к DNS серверу

```
nikita@ubuntu:~$ dig google.ru

;; Query time: 0 msec
;; SERVER: 192.168.40.32#53(192.168.40.32)
;; WHEN: Mon May 21 18:47:31 PDT 2018
;; MSG SIZE rcvd: 265
```

Рис. 1.18: Второе обращение к DNS серверу

Время второго запроса практически равно нулю, что означает, что кэширование работает.

1.3.4 Настройка первичного DNS сервера

Настроим первичный DNS сервер на машине с адресом 192.168.40.32:

```
nikita@ubuntu:~$ cat /etc/bind/named.conf.local
zone "example.com" {
    type master;
    file "/etc/bind/db.example.com";
    allow-transfer { 192.168.40.0/24;192.168.80.0/24;192.168.120.0/24; };
};

zone "40.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.192";
    allow-transfer { 192.168.40.0/24;192.168.80.0/24;192.168.120.0/24; };
};
```

Рис. 1.19: Конфигурация первичного DNS сервера

```
nikita@ubuntu:~$ cat /etc/bind/db.example.com
;
; BIND data file for example.com
;
$TTL      604800
@         IN      SOA      example.com.  root.example.com. (
                        220420131      ; Serial
                        604800          ; Refresh
                        86400           ; Retry
                        2419200         ; Expire
                        604800 )        ; Negative Cache TTL
;
IN        A       192.168.40.32
;
@         IN      NS       ns.example.com.
@         IN      A        192.168.40.32
@         IN      AAAA     ::1
ns        IN      A        192.168.40.32
```

Рис. 1.20: Конфигурация первичного DNS сервера

```
nikita@ubuntu:~$ cat /etc/bind/db.192
;
; BIND reverse data file for local 192.168.40.XXX net
;
$TTL 604800
@         IN      SOA      ns.example.com. root.example.com. (
                        220420132      ; Serial
                        604800          ; Refresh
                        86400           ; Retry
                        2419200         ; Expire
                        604800 )        ; Negative Cache TTL
;
@         IN      NS       ns.
32        IN      PTR      ns.example.com.
```

Рис. 1.21: Конфигурация первичного DNS сервера

1.3.5 Настройка вторичного DNS сервера

Настроим вторичный DNS сервер на машине с адресом 192.168.120.3.

```
nikita@ubuntu:~$ cat /etc/bind/named.conf.local
zone "example.com" {
    type slave;
    file "db.example.com";
    masters { 192.168.40.32; };
};

zone "40.168.192.in-addr.arpa" {
    type slave;
    file "db.192";
    masters { 192.168.40.32; };
};
```

Рис. 1.22: Конфигурация вторичного DNS сервера

1.3.6 Тестирование

Проверим работу связки первичного и вторичного DNS серверов с помощью syslog:

```
nikita@ubuntu:~$ cat /var/log/syslog | grep example.com
May 21 22:39:45 ubuntu named[1071]: zone example.com/IN: refresh: failure trying master 192.168.40.32#53 (source 0.0.0.0#6
May 21 22:39:47 ubuntu named[1071]: zone example.com/IN: Transfer started.
May 21 22:39:47 ubuntu named[1071]: transfer of 'example.com/IN' from 192.168.40.32#53: connected using 192.168.120.3#4046
May 21 22:39:47 ubuntu named[1071]: zone example.com/IN: transferred serial 220420131
May 21 22:39:47 ubuntu named[1071]: transfer of 'example.com/IN' from 192.168.40.32#53: Transfer status: success
May 21 22:39:47 ubuntu named[1071]: transfer of 'example.com/IN' from 192.168.40.32#53: Transfer completed: 1 messages, 7
cs (202000 bytes/sec)
May 21 22:39:47 ubuntu named[1071]: zone example.com/IN: sending notifies (serial 220420131)
May 21 22:40:46 ubuntu named[2448]: zone example.com/IN: loaded serial 220420131
May 21 22:40:46 ubuntu named[2448]: zone example.com/IN: sending notifies (serial 220420131)
```

Рис. 1.23: Результат работы связки первичного и вторичного DNS серверов

1.4 Вывод

Сервера DHCP и удаленная загрузка значительно упрощают работу с сетью для пользователей, что делает эти технологии очень полезными в корпоративных сетях. Кроме того удаленная загрузка позволяет поддерживать одинаковую версию ПО для всех пользователей.

Кэширующий DNS сервер чрезвычайно эффективен при частом доступе к одинаковым ресурсам, как чаще всего и происходит при использовании интернет-ресурсов человеком.