

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО

КАФЕДРА КОМПЬЮТЕРНЫХ СИСТЕМ И ПРОГРАММНЫХ ТЕХНОЛОГИЙ

Отчёт по лабораторной работе №1

Курс: «Администрирование компьютерных сетей»

Тема: «Виртуальное макетирование компьютерных сетей»

Выполнил студент:

Ерниязов Тимур Ертлеуевич

Группа: 13541/2

Проверил:

Малышев Игорь Алексеевич

Санкт-Петербург
2019 г.

Содержание

1	Лабораторная работа №1	2
1.1	Цели работы	2
1.2	Сведения о системе	2
1.3	Создание виртуальных машин	2
1.4	Структура сети	2
1.5	Настройки операционных систем	3
1.5.1	Windows 98	3
1.5.2	Windows XP	3
1.5.3	NetBSD	3
1.5.4	FreeBSD	4
1.6	Тестирование	4

Лабораторная работа №1

1.1 Цели работы

1. Изучить технологию виртуального макетирования компьютерных сетей в среде VMware Workstation.
2. Разработать и настроить полунатуральный эмулятор компьютерной сети.

1.2 Сведения о системе

Работа производилась на реальной системе, со следующими характеристиками:

Элемент	Значение
Имя ОС	Майкрософт Windows 10 Pro (Registered Trademark)
Версия	10.0.16299 Сборка 16299
RAM	16 ГБ
Процессор	Intel(R) Core(TM) i5-7300HQ CPU @ 2.50GHz, 2496 МГц

Для выполнения работы использовалась **VMware Workstation 12 pro (12.5.7 build-5813279)**

1.3 Создание виртуальных машин

С помощью средств **VMware** были созданы виртуальная машины, с использованием ниже представленных операционных систем, с соответствующим выделением оперативной памяти.

Название	Версия	Объем RAM
NetBSD	7.1.1 64-bit x86	256 MB
FreeBSD	11.1-RELEASE 64-bit x86	256 MB
Kali Linux	2017.2 64-bit x86	1.5 GB
Windows XP Professional	5.1.2600 SP3 Сборка 2600	512 MB
Windows 98	4.10.2222A	256 MB

1.4 Структура сети

Была создана ККС, состоящая из трех основных сегментов (**VMnet1, Vmnet2, VMnet3**) и одного вспомогательного (**VMnet4**). Каждый представитель подсетей (VMnet1, Vmnet2, VMnet3) имеет один сетевой адаптер, шлюз – два, а маршрутизатор – три сетевых адаптера.

Название сети	Адрес сети	Подключенные узлы	DHCP
VMnet1	192.168.40.0	Kali Linux, NetBSD, FreeBSD	-
VMnet2	192.168.80.0	FreeBSD, Windows XP	+
VMnet3	192.168.120.0	FreeBSD, Windows 98	-
VMnet4	192.168.32.0	NetBSD	+

Хост Win98 имеет статический адрес 192.168.120.15, хост Kali Linux также имеет статический адрес 192.168.40.32, а хост WinXP получает адрес 192.168.80.128 динамически с помощью виртуального сервера DHCP.

Маршрутизатору(FreeBSD) были назначены следующие адреса: 192.168.40.2 (для связи с сетью VMnet1), 192.168.80.2 (для связи с сетью VMnet2), 192.168.120.2 (для связи с сетью VMnet3).

Функциональное назначение шлюза (обеспечение взаимодействия ККС с внешними сетями) предполагает наличие какого-нибудь механизма сопряжения IP-адресов. Таким механизмом является служба NAT (преобразование сетевых адресов), подключённая к вспомогательной сети VMnet4, в которую (кроме устройства

NAT) входит DHCP-сервер и шлюз. Адрес «внешнего» сетевого адаптера шлюза назначается динамически (DHCP-сервером сети VMnet4) – 192.168.32.128, а адрес «внутреннего» сетевого адаптера шлюза (входящего в сеть VMnet1) статически – 192.168.40.57.

1.5 Настройки операционных систем

1.5.1 Windows 98

В свойствах TCP/IP были заданы:

1. IP-адрес = 192.168.120.15
2. Маска подсети = 255.255.255.0
3. Шлюз = 192.168.120.2

1.5.2 Windows XP

В свойствах TCP/IP были заданы:

1. DNS-сервер = 192.168.80.254
2. Шлюз = 192.168.80.2

1.5.3 NetBSD

1. Узнать названия сетевых адаптеров с помощью команды `ifconfig`, в моем случае это `pcn0`, `pcn1`.
2. Разрешаем ip forwarding добавляя в файл `/etc/sysctl.conf`:

```
1 net.inet.ip.forwarding=1
```

3. Внести следующие настройки в файл `/etc/rc.conf`:

- (a) Указание шлюза по умолчанию:

```
1 defaultroute=192.168.32.2
```

- (b) Задаем ip адрес и сетевую маску для одного из интерфейсов:

```
1 ifconfig_pcn0=inet 192.168.40.57 netmask 255.255.255.0
```

- (c) Разрешаем настройку по DHCP.

```
1 dhclient=yes
2 dhclient_flags=pcn1
3 ifconfig_pcn1=DHCP
```

- (d) Разрешаем запуск NAT:

```
1 ipnat=yes
```

4. Задаем правила NAT в файле `/etc/ipnat.conf`:

```
1 map pcn1 192.168.40.0/24 -> 0/32 portmap tcp/udp 40000:60000
2 map pcn1 192.168.40.0/24 -> 0/32
```

5. В консоли прописываем следующие команды:

```
1 route add -net 192.168.80.0 -netmask 255.255.255.0 192.168.40.2
2 route add -net 192.168.120.0 -netmask 255.255.255.0 192.168.40.2
```

Так как например Windows 98 находится в другом широковещательном домене, были добавлены маршруты, чтобы NetBSD знал куда отвечать.

1.5.4 FreeBSD

1. Узнать названия сетевых адаптеров с помощью команды `ifconfig`, в моем случае это `em0`, `em1`, `em2`.
2. Внести следующие настройки в файл `/etc/rc.conf`:

```
1 gateway_enable="YES"
```

- (a) Разрешаем ip forwarding при помощи команды:

```
1 defaultrouter=192.168.40.57
```

- (b) Задаем ip адрес и сетевую маску для всех интерфейсов:

```
1 ifconfig_em0=inet 192.168.40.2 netmask 255.255.255.0
2 ifconfig_em1=inet 192.168.80.2 netmask 255.255.255.0
3 ifconfig_em2=inet 192.168.120.2 netmask 255.255.255.0
```

- (c) Разрешаем запуск NAT:

```
1 ipnat_enable="YES"
```

3. После этого необходимо задать правила NAT для сопряжения адресов. Это делается путем редактирования файла `/etc/ipnat.rules`:

```
1 map em0 192.168.80.0/24 -> 0.0.0.0/32 portmap tcp/udp 40000:60000
2 map em0 192.168.80.0/24 -> 0.0.0.0/32
3 map em0 192.168.120.0/24 -> 0.0.0.0/32 portmap tcp/udp 40000:60000
4 map em0 192.168.120.0/24 -> 0.0.0.0/32
```

Эти строки позволяет корректно обрабатывать tcp, udp, icmp пакеты.

1.6 Тестирование

Тестирование заключалось в проверке возможности выхода в интернет из каждой системы, путем отправки ping на адрес 8.8.8.8 (публичный DNS Google). Во всех системах данная команда отработала корректно, что говорит о правильно настроенной ККС.

Вывод

В данной работе была рассмотрена эмуляция корпоративной компьютерной сети(ККС), которая содержит три основных и один вспомогательный сегмент сети. Средствами VMWare были созданы:

- Виртуальные машины, с различными представителями операционных систем.
- Виртуальные сети(с различными параметрами).
- Адаптеры для виртуальных машин.

Это позволило эмулировать заданную ККС, в которой использовались:

- Статическая адресация;
- Динамическое выделение IP адреса;
- Статическая и динамическая маршрутизация.

Также имелась возможность выхода в сеть "Интернет" из каждой операционной системы.

На мой взгляд **VMware Workstation** в большей степени подходит для визуализации какой-либо требуемой операционной системы, так как нередко случаи необходимости использования платформозависимого программного обеспечения. Макетирование сетей в данной программе является не лучшим решением, ввиду отсутствия какого-либо визуального редактора. Для подобных целей, лучше использовать специализированное ПО, например **Graphical Network Simulator 3**.