


С.И. Богомолов

Программные средства систем связи

Лабораторный практикум №2

Министерство образования и науки РФ
Томский университет систем управления и радиоэлектроники

Радиотехнический факультет
Кафедра телекоммуникаций и основ радиотехники

«Утверждаю»
/ Зав. кафедрой ТОР
А.В. Пуговкин
 2010 г.

Лабораторный практикум №2
по дисциплине
«Программные средства систем связи»
для студентов радиотехнического факультета

Лабораторный практикум составил:
к.т.н., доцент С.И. Богомолов

Томск - 2010 г.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
Общие положения.....	3
Правила выполнения лабораторных работ.....	3
Содержание и оформление отчета	4
Защита работы.....	4
РАБОТА № 1	6
ИССЛЕДОВАНИЕ СЕТЕВЫХ КОМПОНЕНТОВ С ПОМОЩЬЮ ИМИТАТОРА NET-SIMULATOR	6
Краткие сведения о сетевом имитаторе Net-Simulator	6
Команды Net-Simulator.....	8
Основы работы с сетевым имитатором Net-Simulator.....	13
Предварительная подготовка.....	14
Контрольные вопросы и задания.....	15
Лабораторное задание	16
РАБОТА № 2	20
МОДЕЛИРОВАНИЕ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ С ПОМОЩЬЮ ИМИТАТОРА NET-SIMULATOR	20
Предварительная подготовка.....	20
Контрольные вопросы и задания.....	21
Лабораторное задание	23

ВВЕДЕНИЕ

ОБЩИЕ ПОЛОЖЕНИЯ

Лабораторный практикум по курсу «Программные средства систем связи» имеет целью закрепить и расширить теоретические знания студентов при изучении принципов построения и функционирования устройств и систем коммуникации сетей ЭВМ, ознакомить их с методиками исследования основных компонентов и процессов систем связи, в том числе, и с использованием моделирования устройств и систем связи.

Данный цикл лабораторного практикума, предназначенного для студентов специальности 210405, содержит описание следующих работ:

Исследование сетевых компонентов с помощью имитатора Net-Simulator.

Исследование сетей с помощью имитатора Net-Simulator.

Работа «Исследование сетевых компонентов с помощью имитатора Net-Simulator» является вводной и предназначена для получения первичных навыков работы с сетевым симулятором Net-Simulator. Последующие работы цикла ориентированы на исследование отдельных вопросов функционирования вычислительных сетей с помощью сетевого тренажера Net-Simulator.

ПРАВИЛА ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ

Перед выполнением работы студенты на этапе предварительной подготовки:

- а) изучают соответствующие разделы теоретического курса;
- б) знакомятся с описанием лабораторной работы и подготавливают шаблон отчета по лабораторной работе;
- в) выполняют необходимые предварительные расчёты, изложенные в разделе «Предварительная подготовка».

К выполнению лабораторных работ допускается только студенты, выполнившие требования предыдущего раздела, и подтвердившие свою подготовленность при собеседовании с преподавателем.

Лабораторные работы выполняются индивидуально фронтальным методом. При выполнении работ рекомендуется следовать методическим указаниям. Разрешается проведение допол-

нительных исследований (не в ущерб основному заданию).

В процессе выполнения работы составляется предварительный отчёт, который должен содержать таблицы и графики полученных экспериментально зависимостей.

Если при составлении предварительного отчёта выявится недостаточность или сомнительность полученных данных, то необходимо экспериментально получить недостающие данные и произвести проверку сомнительных результатов.

Работа считается выполненной после утверждения предварительного отчёта преподавателем.

Студенты, не выполнившие работу в часы занятий, выполняют её в специально отведённое время.

СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ ОТЧЕТА

Отчёт по выполненной работе составляется индивидуально каждым студентом.

Отчёт оформляется на листах формата А4 (достаточно электронной копии в редакторе OpenOffice.org Writer). У осей графиков должна быть проставлены числовые значения и единицы размерности.

Отчёт должен содержать:

- а) цель исследования;
- б) результаты расчётов, полученных на этапе предварительной подготовки;
- в) структурные схемы устройств и систем, характеристик которых исследуются в данной работе;
- г) структурные схемы измерительных установок для исследования характеристик устройств и систем связи;
- д) результаты исследований в виде таблиц, графиков и изображений, получаемых на экранах измерительных приборов;
- е) выводы, полученные на основании анализа расчётных и экспериментальных данных.

ЗАЩИТА РАБОТЫ

Лабораторная работа считается выполненной после защиты результатов работы.

При защите результатов работы студент должен представить оформленный отчёт, сохраненные результаты компьютерного

эксперимента и продемонстрировать свои знания в следующих разделах работы:

а) структурные схемы исследуемых устройств и систем и основы их функционирования;

б) структурные схемы измерительных установок для исследования характеристик устройств и систем связи и характеристики, которые могут быть получены с помощью данных установок;

в) результаты расчетов предварительной подготовки;

г) теоретическое обоснование полученных экспериментальных зависимостей.

РАБОТА № 1

ИССЛЕДОВАНИЕ СЕТЕВЫХ КОМПОНЕНТОВ С ПОМОЩЬЮ ИМИТАТОРА NET-SIMULATOR

Целью работы «Исследование сетевых компонентов с помощью имитатора Net-Simulator» является знакомство с сетевым имитатором Net-Simulator, предназначенным для имитационного моделирования процессов, происходящих в сетях связи, а также получения первичных навыков конфигурирования сетевых интерфейсов.

КРАТКИЕ СВЕДЕНИЯ О СЕТЕВОМ ИМИТАТОРЕ NET-SIMULATOR

Net-Simulator является программным продуктом, позволяющим осуществить имитационное моделирование сетей ЭВМ, имеет открытую лицензию и предназначен для учебных целей. Исходные материалы продукта размещены на сайте <http://sourceforge.net/projects/net-simulator/>. Копии основных документов этого проекта продублированы в каталоге S:\БогомоловСИ\Model\NetSim\Net-Simulator.

Net-Simulator позволяет строить виртуальные вычислительные сети из виртуальных сетевых устройств: маршрутизаторов, настольных компьютеров, концентраторов и т.п. Устройствами можно управлять при помощи интерфейса командной строки из виртуальных терминалов. В виртуальных сетях реализованы канальный и сетевой уровни в соответствии с ISO OSI, что достаточно для первичного обучения конфигурированию и поиску неисправностей в вычислительных сетях.

Проект является открытым. Программное обеспечение разрабатывается и распространяется в соответствии с положениями GNU GPL.

В Net-Simulator реализованы два уровня ISO OSI: канальный и сетевой, что позволяет решать следующие задачи: изучение принципов работы коммутаторов второго и третьего уровня, пассивных концентраторов; отработку практических навыков статической маршрутизации в IP-сетях и поиска неисправностей в IP-сетях.

Физическая природа сети не учитывается. Предполагается,

что пакеты канального уровня распространяются в среде аналогичной локальной сети на основе Ethernet. На канальном уровне используется простейший Ethernet-образный протокол, который предусматривает адресацию по 6-ти байтовым MAC-адресам. Уникальность MAC-адресов обеспечивает ядро Net-Simulator. Пакет канального протокола представляет собой объект Java и не имеет аналогов в реальных сетях.

На сетевом уровне используется ограниченная реализация IP в соответствии с RFC791. Для преобразования IP-адресов в MAC реализована служба ARP на основе широкоэвентельных запросов.

Для работы служебных утилит, таких как ping, используется ограниченная реализация ICMP в соответствии с RFC792.

В главном окне NET-Simulator отображается поле в которое можно добавлять различные сетевые устройства из меню Устройства. Поддерживаются следующие типы устройств:

Маршрутизатор. Коммутатор 3-го уровня с 8-ью интерфейсами и поддержкой IP4.

Настольный компьютер. Фактически маршрутизатор с одним интерфейсом.

Концентратор (Hub). Простейшее устройство ретранслирующее пакеты канального уровня на свои интерфейсы. Не имеет терминала и соответственно никак не управляется.

Коммутатор (Switch). Коммутатор 2-го уровня с 8-ью интерфейсами. Коммутирует пакеты канального уровня на основе таблиц MAC-адресов, по аналогии с известными алгоритмами используемыми в Ethernet-свитчах.

Устройства соединяются с помощью универсальной среды передачи данных, виртуального патчкорда. При прохождении пакета через патчкорд, он подсвечивается для визуального отслеживания активности в сети.

Вновь добавленные устройств появляются в верхнем левом углу, после чего их можно перетаскивать мышкой в удобное место. Вилки патчкордов «приклеиваются» к розеткам интерфейсов устройств. Нажатие правой кнопки мыши на устройстве открывает контекстное меню, которое позволяет просмотреть свойства, открыть терминал или удалить устройство. Двойной щелчок левой кнопкой мыши открывает терминал.

Проекты сохраняются в формате xml. Проекты можно сохранять в виде html-отчетов. Отчет состоит непосредственно из html-файла с детальным описанием проекта и одноименного файла со схемой виртуальной сети в формате png. Отчеты формируются путем конвертации исходного xml-файла проекта при помощи xsl-шаблона. По умолчанию используется шаблон `cfg/tohtml.xsl`. Изменяя шаблон можно добиться желаемого вида отчета.

Виртуальные устройства в Net-Simulator управляются при помощи интерфейса командной строки из виртуальных терминалов. Терминал устройства можно открыть двойным кликом на значке устройства или через контекстное меню. Поддерживается история команд, клавиши вверх/вниз позволяют просматривать историю команд.

Список команд доступных на данном устройстве можно посмотреть командой `help`. Курсивом здесь и далее выделены служебные слова и символы, набираемые на экранах виртуальных терминалов. Сочетание клавиш `Ctrl+L` очищает терминал. Краткая справка по любой команде выводится при вызове команды с опцией `-h`.

КОМАНДЫ NET-SIMULATOR

В режиме командной строки пользователю доступны следующие команды: `help`; `route`; `ifconfig`; `ping`; `arp`; `mactable`; `help`.

help – выводит список доступных команд.

`help [-h]`

Опции	Описание
-------	----------

<code>-h</code>	краткая справка.
-----------------	------------------

Содержимое квадратных скобок является необязательным. На позициях угловых скобок размещают соответствующие значения.

route – позволяет управлять таблицей маршрутизации устройств поддерживающих протокол IP4.

`route [-h] [{-add|-del}] <target> [-netmask <address>] [-gw <address>] [-metric <M>] [-dev <If>]`

Опции	Описание
-------	----------

<code>-h</code>	краткая справка.
-----------------	------------------

<code>target</code>	адрес назначения. Назначением может быть под-
---------------------	---

сеть или отдельный узел в зависимости от значения маски подсети. Если маска равна 255.255.255.255 или отсутствует совсем, назначением будет узел, иначе назначением будет сеть.

-add добавляет новый маршрут в таблицу маршрутизации.

-del удаляет маршрут из таблицы маршрутизации.

-dev <If> принудительно присоединяет маршрут к определенному интерфейсу. If – имя интерфейса.

-gw <address> направляет пакеты по этому маршруту через заданный шлюз; address – адрес шлюза.

-netmask <address> маска подсети используемая совместно с адресом назначения при добавлении маршрута address – маска. Если маска не задана явно, подразумевается 255.255.255.255.

-metric <M> метрика, используемая в данном маршруте. M — целое число большее или равное нулю.

Если **route** вызывается без параметров, то команда выводит на экран таблицу маршрутизации:

```
=>route
```

```
IP routing table
```

Destination	Gateway	Netmask	Flags	Metric	Iface
10.0.0.0	*	255.0.0.0	U	1	eth0
11.0.0.0	10.0.0.10	255.0.0.0	UG	1	eth0
192.168.120.1	10.0.0.10	255.255.255.255	UGH	1	eth0

Если маршрут не использует шлюз, вместо адреса шлюза выводится *. Flags может содержать значение: U — маршрут активен, G — маршрут использует шлюз, H — назначением является узел.

Примеры:

```
=>route -add 192.168.120.0 -netmask 255.255.255.0 -dev eth0
```

```
=>route
```

```
IP routing table
```

Destination	Gateway	Netmask	Flags	Metric	Iface
192.168.120.0	*	255.255.255.0	U	1	eth0

```
=>
```

```
=>route -add 192.168.121.10 -gw 192.168.120.10
```

```
=>route
```

```
IP routing table
```

Destination	Gateway	Netmask	Flags	Metric	Iface
192.168.120.0	*	255.255.255.0	U	1	eth0
192.168.121.10	192.168.120.1	255.255.255.255	UGH	1	eth0

=>

ifconfig – конфигурирует сетевые интерфейсы.

ifconfig [-h] [-a] [<interface>] [<address>] [-broadcast <address>] [-netmask <address>] [-up|-down]

Опции Описание

-h краткая справка.

-a показывает информацию о всех интерфейсах. Если данная опция отсутствует, выводится информация только об активных интерфейсах.

interface конфигурирует или показывает информацию только о заданном интерфейсе.

address IP-адрес, присваиваемый интерфейсу.

-broadcast <address> широковещательный адрес, присваиваемый интерфейсу, address – широковещательный адрес.

-netmask <address> маска подсети, используемая совместно с адресом; address – маска. Если маска не задана явно, маска принимается равной стандартным значения для стандартных классов подсетей А, В и С.

-up активирует интерфейс. При активизации интерфейса для него автоматически добавляется соответствующий маршрут в таблице маршрутизации.

-down деактивирует интерфейс. При деактивации интерфейса соответствующий маршрут автоматически удаляется из таблицы маршрутизации.

Если **ifconfig** вызывается без параметров, то команда выводит на экран данные о состоянии всех активных интерфейсов:

=>**ifconfig**

```
eth0    Link encap:Ethernet HWaddr 0:0:0:0:CF:0
inet addr:192.168.120.1 Bcast:192.168.120.255 Mask:255.255.255.0
UP
RX packets:23 errors:0 dropped:0
TX packets:23 errors:0 dropped:0
RX bytes:0 TX bytes:0
```

HWaddr – уникальный 6-ти байтовый адрес интерфейса, аналогичный MAC-адресу в Ethernet сетях. Назначается автомати-

чески.

Примеры:

```
=>ifconfig eth0 192.168.120.1 -broadcast 192.168.120.255 -  
netmask 255.255.255.0 -up
```

```
=>ifconfig
```

```
eth0      Link encap:Ethernet  HWaddr 0:0:0:0:CF:0  
inet addr:192.168.120.1  Bcast:192.168.120.255 Mask:255.255.255.0  
UP  
RX packets:0 errors:0 dropped:0  
TX packets:0 errors:0 dropped:0  
RX bytes:0 TX bytes:0
```

ping – использует ICMP протокол чтобы проверить достижимость интерфейса удаленного узла. ping посылает удаленному узлу ICMP ECHO_REQUEST и ожидает в течении определенного промежутка времени ICMP ECHO_RESPONSE. В случае получения ответа выводит данные о прохождении ICMP-пакета по сети. (Отмена: «Ctrl-c»).

```
ping [-h] [-i <interval>] [-t <ttl>] <destination>
```

Опции Описание

-h краткая справка.

-i <interval> задает частоту ICMP-запросов; interval – интервал между запросами в секундах. По умолчанию отсылается один пакет в секунду.

-t <ttl> задает значение атрибута Time to Live в генерируемых IP-пакетах. ttl – целое число от 0 до 255. По умолчанию TTL равно 64.

destination IP-адрес исследуемого узла

Примеры:

```
=>ping 192.168.120.1
```

```
PING 192.168.120.1
```

```
64 bytes from 192.168.120.1: icmp_seq=0 ttl=62 time=477 ms
```

```
64 bytes from 192.168.120.1: icmp_seq=1 ttl=62 time=435 ms
```

```
64 bytes from 192.168.120.1: icmp_seq=2 ttl=62 time=234 ms
```

```
64 bytes from 192.168.120.1: icmp_seq=3 ttl=62 time=48 ms
```

```
64 bytes from 192.168.120.1: icmp_seq=4 ttl=62 time=87 ms
```

```
64 bytes from 192.168.120.1: icmp_seq=5 ttl=62 time=56 ms
```

ping выводит результат исследования удаленного узла в следующем формате:

64 bytes from 192.168.120.1 – размер полученного ответа и адрес источника ответа. В NET-Simulator размер пакета имеет условное значение и всегда равен 64B.

icmp_seq=0 – номер пакета. Каждый запрос содержит свой номер, как правило формируется инкрементно. ping выводит номер пакета из каждого полученного ответа.

ttl=62 – значение TTL из полученного ответа. time=48 ms — время прохождения пакетом полного маршрута (туда и обратно, round-trip time) в миллисекундах.

arp – показывает ARP-таблицу устройства. Кроме того опция -r позволяет сформировать запрос для определения MAC-адреса по явно заданному IP-адресу. Эта функция обычно отсутствует в реальных устройствах, в NET-Simulator она добавлена для наглядности при изучении протоколов канального и сетевого уровня.

```
arp [-h] [-r <IP-address> <interface>]
```

Опции Описание

-h краткая справка.

-r <IP-address> <interface> Прежде чем вывести ARP-таблицу, предпринимает попытку найти MAC-адрес по явно заданному IP-адресу; IP-address - IP-адрес, для которого определяется MAC-адрес. interface имя интерфейса в сети подсоединенной к которому будет происходить поиск.

Если arp вызывается без параметров, то команда выводит на экран ARP-таблицу:

```
=>arp
```

Address	HWaddress	iface
10.0.0.10	0:0:0:0:BC:0	eth0
10.0.0.11	0:0:0:0:1F:2	eth0

Примеры:

```
=>arp -r 192.168.120.12 eth1
```

```
=>arp
```

Address	HWaddress	iface
10.0.0.10	0:0:0:0:BC:0	eth0
10.0.0.11	0:0:0:0:1F:2	eth0
192.168.120.12	0:0:0:0:12:1	eth1

mactable – показывает таблицу MAC-адресов коммутаторов второго уровня.

mactable [-h]

Опции	Описание
-h	краткая справка.

Примеры:

=>*mactable*

MACAddress	port
0:0:0:0:B3:0	0
0:0:0:0:2F:2	2
0:0:0:0:03:0	3

Где port — номер порта на коммутаторе. Нумерация портов идет по порядку начиная с нуля.

ОСНОВЫ РАБОТЫ С СЕТЕВЫМ ИМИТАТОРОМ NET-SIMULATOR

Запуск программы Net-Simulator в среде Linux может быть выполнен из главного меню (кнопка К / Образование / Разное / Net-Simulator). При этом раскрываются два окна: окно истории функционирования симулятора и окно графического представления сети. На раскрытом рабочем окне программы Net-Simulator вкладка «Проект» содержит набор традиционных директив для работы с файлами. Вкладка «Устройства» содержит перечень сетевых компонентов, доступных для использования при моделировании сетей. Вкладка «Сервис» открывает дополнительные возможности представления анализируемого проекта. Дополнительные сведения о функционировании симулятора размещены в каталоге S:\ БогомоловСИ\ Model\ NetSim\ Net-Simulator.

Для составления новой схемы сети следует указать на пункт меню «Создать» вкладки «Проект». Сетевые компоненты (компьютеры, концентраторы, коммутаторы и т.д.), необходимые для построения схемы сети, выбирают с помощью вкладки «Устройства» с дальнейшим размещением на рабочем поле проекта.

Для соединения сетевых компонентов используют пункт меню «Среда передачи (кабель)» вкладки «Устройства». Для «подключения» кабеля к устройству следует «разъем» кабеля подтащить к изображению «гнезда» прибора. При удачном «подключении» кабеля на приборе будет подсвечен индикатор соединения соответствующего порта.

Для конфигурирования устройств следует открыть виртуальный терминал двойным «кликом» на графическом изображении соответствующего узла либо выбором пункта «Терминал» при нажатии правой кнопки мыши на выбранном устройстве. Перечень команд, доступных для имитации работы выбранного устройства, может быть получен при вводе на терминале команды *help*.

Моделирование сетевых процессов основано на имитации передачи дейтаграмм IP в сетях IEEE 802, поэтому каждому узлу сети должен быть присвоен IP адрес (как индивидуальный, так и широковещательный) и маска адреса. MAC адрес каждому узлу присваивается программой автоматически. Номера сетевых интерфейсов каждого узла начинаются с нуля, например, eth0. В устройствах, имеющих несколько портов, их нумерация на графических образах выполняется слева направо.

Для формирования отчета результаты моделирования в виде копий экрана (протоколы терминалов, схемы сетей) последовательно переносятся на графический редактор, например, KolourPaint {кнопка К (аналог кнопки ПУСК в среде Windows) / ГРАФИКА /}, с последующим выбором нужных сегментов экрана и переносом их в текстовый редактор, например, Write.

ПРЕДВАРИТЕЛЬНАЯ ПОДГОТОВКА

Изучить принципы функционирования сетевого уровня модели OSI, типы адресов сетей TCP/IP, классификацию IP-адресов, назначение масок в IP-адресации, ознакомиться с работой сетевого оборудования (сетевая карта, концентратор, коммутатор, маршрутизатор), с протоколами ARP и ICMP и утилитой ping по материалам лекций и литературных источников, рекомендуемых, в том числе, и для самостоятельного изучения.

Ознакомиться с описанием сетевого имитатора Net-Simulator и его основных компонентов по материалам данного практикума. Изучить раздел «Основы работы с сетевым имитатором Net-Simulator» методических указаний по данной лабораторной работе и подготовить необходимые схемы моделей сетей, рассмотренные в данном разделе.

Подготовить схему ЛВС, состоящую из 3 компьютеров по технологии «звезда». В качестве IP-номера сети использовать

адреса, зарезервированные для автономных локальных сетей класса С вида 192.168.XYZ.0, где XY – порядковый номер студента по списку, Z – произвольная цифра от 0 до 9. Номер узлов в сети выбирается произвольно.

Изменить схему сети, заменив концентратор коммутатором. При этом изменить номер сети одного из узлов путем изменения цифры в позиции Z.

Изменить модернизированную схему объединенной сети, вернув на место концентратор и введя маршрутизатор для логической структуризации сети. Портам маршрутизатора присвоить адреса, соответствующие используемым номерам сетей.

Дополнить схему объединенной сети введением подсетей. Для этого заменить концентратор вторым маршрутизатором, а сеть из двух компьютеров разбить на две подсети с использованием масок. Для формирования номеров подсетей использовать четыре бита младшего байта адреса. В качестве номеров подсетей использовать порядковый номер студента, и число, на единицу большее.

Результатами предварительной подготовки по данной работе являются разработанные схемы сетей с указанием IP-адресов всех сетевых компонентов.

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Какие типы адресов используются в сетях TCP/IP?
2. Что такое локальный адрес и как он используется?
3. Назначение IP-адреса.
4. С какой целью введены символьные имена узлов сети?
5. Каким образом и для чего IP-адреса разбиты на классы?
6. Что такое групповой адрес и как он используется?
7. Что такое широковещательная рассылка?
8. Какая информация содержится в адресном поле IP пакета для выполнения широковещательной рассылки в удаленной сети?
9. Какая информация содержится в адресном поле IP пакета для выполнения широковещательной рассылки в локальной сети?
10. Какие ограничения накладываются на выбор IP-адресов?
11. Что понимается под термином loopback?

12. Для каких целей используется адрес сети 127.0.0.0?
13. Какую функцию выполняет маска адреса?
14. Указать маску для сетей класса А (в двоичном и десятичном формате).
15. Привести маску для сетей класса В (в двоичном и десятичном формате).
16. Привести маску для сетей класса С.
17. Указать маску для сетей класса С (в двоичном и десятичном формате).
18. Что следует понимать под адресами класса D?
19. Что такое таблица маршрутизации?
20. Основные компоненты таблицы маршрутизации.
21. Какие сетевые компоненты используют таблицу маршрутизации?
22. Какие сетевые компоненты не используют таблицу маршрутизации?
23. Что представляет собой проект Net-Simulator?
24. Какие компоненты сети могут быть смоделированы в Net-Simulator?
25. Какие команды используются в Net-Simulator?
26. Какие команды используются для контроля конфигурации сетевых интерфейсов?
27. Какие команды используются для контроля таблицы маршрутизации?
28. Для каких целей используется команда ping?
29. Какие протоколы используются при применении команды ping?

ЛАБОРАТОРНОЕ ЗАДАНИЕ

1. Ознакомиться с описанием сетевого имитатора Net-Simulator и его основных компонентов по материалам данного практикума и рекомендуемых в нем литературных источников (каталог \Lab3\Method\). Создать папку с названием Lab_3 на сервере X. Промежуточные и окончательные результаты данной работы сохранять в этом каталоге.

2. В среде Linux запустить пакет Net-Simulator (кнопка К / Образовательные / Net-Simulator). Ознакомиться с пакетом Net-Simulator с помощью его справочной системы.

3. Выбрать пункт «Проект» главного меню рабочего окна программы Net-Simulator.; в ниспадающем подменю выбрать пункт «Создать».

4. Используя схему сети, подготовленную в результате предварительной подготовки, построить модель ЛВС, состоящую из 3 компьютеров и концентратора.

5. Открыть терминал одного из компьютеров. Проверить функционирование команд *help* и *echo*. Результаты выполнения команд сохранить в отчете.

6. Открыть виртуальный терминал каждого компьютера и с помощью команды *ifconfig* ввести параметры сетевых интерфейсов всех узлов сети. С помощью команды *route* проконтролировать таблицы маршрутизации всех компьютеров.

7. С помощью команды *arp* сформировать запросы на соседние узлы сети. С помощью команды *ifconfig* проконтролировать состояние интерфейсов всех узлов сети. Обратит внимание на количество переданных и принятых пакетов на каждом узле. Провести анализ перемещения пакетов по сети, а также количества широковещательных пакетов и пакетов с конкретным адресом назначения на основании их времени кругооборота (пакеты широковещательной рассылки удобно контролировать на тех узлах, с которыми на данный момент не проводился сеанс обмена сообщениями).

8. С помощью команды *ping* проверить достижимость любого из узлов сети. С помощью команды *ifconfig* проконтролировать состояние интерфейсов всех узлов сети. Обратит внимание на количество переданных и принятых пакетов.. Провести анализ перемещения пакетов по сети, а также количества широковещательных пакетов и пакетов с конкретным адресом назначения на основании их времени кругооборота.

9. Повторить п. 8 лабораторного задания для всех компьютеров данной сети. Выявить закономерности формирования широковещательных пакетов.

10. Выполнить команду *ping* для проверки достижимости узла назначения сети с тем же самым адресом, что и адрес источника. Объяснить функционирование системы в этой ситуации.

11. Повторить п. 8 лабораторного задания для проверки достижимости узлов сети с адресами, которые отсутствуют в данной

сети. При этом в качестве адресов несуществующих узлов выбирать номера, которые могут как относиться к данной сети, так и находиться за ее пределами. Обосновать полученные результаты.

12. Увеличивая параметр *i* команды *ping* добиться, чтобы каждый отправляемый ICMP пакет требовал рассылки ARP пакетов. Оценить продолжительность хранения адресов *timeout* ARP таблиц до их удаления.

13. Используя схему, подготовленную в результате предварительной подготовки, построить модель сети, состоящую из 3 компьютеров и коммутатора.

14. Открыть терминал одного из двух компьютеров, принадлежащих к одной сети. С помощью команды *ping* проверить достижимость второго узла сети. С помощью команды *ifconfig* проконтролировать состояние интерфейсов всех узлов сети.

15. С помощью команды *ping* проверить достижимость компьютера, находящегося в другой сети. С помощью команды *ifconfig* проконтролировать состояние интерфейсов всех узлов сети. Провести анализ перемещаемых пакетов. Объяснить полученные результаты.

16. С помощью команды *route* ввести в таблицу маршрутизации информацию о маршруте, обеспечивающем передачу пакетов во вторую сеть.

17. Повторить п. 14 задания. С помощью команды *route* ввести в таблицу маршрутизации компьютера-получателя информацию о маршруте, обеспечивающем передачу пакетов во вторую (для данного компьютера) сеть. Снова повторить п. 14. Объяснить полученные результаты.

18. Используя схему, подготовленную в результате предварительной подготовки, построить модель сетей, объединенных с помощью маршрутизатора.

19. Повторить п.п. 13...16 для одного из двух компьютеров, размещенных в одной сети.

20. Используя схему, подготовленную в результате предварительной подготовки, построить модель объединенной сети, с разбиением на подсети помощью маршрутизатора.

21. С помощью команды *ifconfig* ввести параметры сетевых интерфейсов всех узлов сети. С помощью команды *route* ввести параметры таблицы маршрутизации, обеспечивающих достижи-

мость любого узла сети с любого компьютера.

22. С помощью команды *ping* проверить достижимость всех узлов спроектированной сети для всех компьютеров.

Литература

1. Олифер В.Г., Олифер Н.А. Компьютерные сети. С.-Петербург, изд-во «Питер».2007. - 957с.

2 Козлов В.Г., Семигук Е.С., Богомолов С.И. Программные средства систем связи: Учебное пособие. – Томск: ТМЦДО, 2008. – 162 с.

3. <http://sourceforge.net/projects/net-simulator/>.

РАБОТА № 2

МОДЕЛИРОВАНИЕ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ С ПОМОЩЬЮ ИМИТАТОРА NET-SIMULATOR

Целью работы «Моделирование вычислительных сетей с помощью имитатора Net-Simulator» является ознакомление с популярными протоколами стека TCP/IP (ARP, ICMP, IP, UDP), освоение принципов маршрутизации и построения составных сетей, а также закрепление навыков конфигурирования сетевых интерфейсов.

ПРЕДВАРИТЕЛЬНАЯ ПОДГОТОВКА

Ознакомиться с протоколами ARP, ICMP, IP, UDP по материалам лекций и литературных источников, рекомендуемых для изучения данной дисциплины. Копии версий некоторых рекомендаций (Request for Comments) продублированы в каталоге S:\БогомоловСИ\Model\NetSim\Net-Simulator\RFC.

Освоить принципы функционирования сетевого уровня модели OSI, типы адресов сетей TCP/IP, классификацию IP-адресов, назначение масок в IP-адресации, освоить принципы маршрутизации и распределения адресного пространства в составных сетях по материалам лекций и других источников.

В процессе подготовки к лабораторной работе выполнить 3 задания по соответствующему варианту. Полное описание заданий приведено в разделе «Контрольные работы» учебного пособия Козлова В.Г., Семигук Е.С. «Программные средства систем связи» [2]. Там же приведены методические указания по выполнению этих заданий.

Первое задание включает составление таблицы маршрутизации для одного из маршрутизаторов составной сети, схема которой для каждого варианта приведена в пособии [2]. В процессе выполнения задания следует обратить внимание на то, что не все номера сетей указаны на схемах явно.

Второе задание предполагает обратную задачу: построение возможного варианта схемы сети по таблице маршрутизации одного из ее узлов. После выполнения этого задания для проверки рекомендуется составить таблицу маршрутизации для этого же узла спроектированной сети и сравнить ее с исходной.

Третье задание состоит из предварительного построения схемы составной сети по описанию сетевых интерфейсов ее узлов. В последующем, по построенной схеме планируется определить кратчайшее расстояние между заданными узлами.

Результатами предварительной подготовки по данной работе являются разработанные схемы сетей с указанием адресов всех сетевых компонентов, а также таблицы маршрутизации, содержащие необходимые сведения о возможных маршрутах пересылки пакетов.

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Какие ограничения следует учитывать при выборе номеров сетей (подсетей)?
2. Какие ограничения следует учитывать при выборе номеров узлов в сети?
3. Приведите маску, использующуюся по умолчанию для сетей класса А (В или С) в двоичном и десятичном представлении. (А.В.С)
4. Планируется, что предприятие будет расширяться и количество ПК в дальнейшем будет расти. Для упрощения администрирования расширяющейся сети предполагается поделить ее на n подсетей. Определить маску, которую необходимо использовать для получения требуемого количества подсетей. В ответе приведите количество помеченных бит в маске (помеченными считаются биты с единичным значением) и саму маску в двоичном и десятичном представлении ($n=2, 3, 6, 11, 14, 22, 28$).
5. Имеется IP-адрес класса С (выбрать из задания предварительной подготовки лабораторной работы 1 своего варианта) и маска подсети (выбрать из таблицы). Укажите IP-адрес сети, IP-адрес подсети и IP-адрес конечного узла.

Таблица

№ варианта	№ варианта	№ варианта	Маска подсети
1	8	15	255.255.255.128
2	9	16	255.255.255.192
3	10	17	255.255.255.224
4	11	18	255.255.255.240
5	12	19	255.255.255.248

6	13	20	255.255.255.252
7	14	21	255.255.255.254

6. Планируется, что предприятие будет расширяться и количество ПК в дальнейшем будет расти. Для упрощения администрирования расширяющейся сети предполагается поделить ее на n подсетей. Определить возможное количество непомеченных бит в маске и узлов в каждой подсети; ($n=2, 3, 6, 11, 14, 22, 28$).
7. В сети 172.16.0.0 необходимо выделить подсети так, чтобы к каждой подсети можно было подключить до 600 хостов. Выберите такую маску подсети, чтобы допустить рост числа подсетей в будущем. (ответ приведите в десятичном и двоичном исчислении).
8. Сеть 172.16.0.0 содержит 8 подсетей. Вам необходимо подключить к подсети максимально возможное число хостов. Какую маску подсети следует выбрать? (ответ приведите в десятичном и двоичном исчислении).
9. В сети 192.168.55.0 необходимо выделить максимальное число подсетей так, чтобы к каждой подсети можно было подключить 25 хостов. Какую маску подсети следует выбрать? (ответ приведите в десятичном и двоичном исчислении).
10. Ваша сеть класса А содержит 60 подсетей. В следующие два года вам необходимо организовать еще 40 подсетей, причем так, чтобы к каждой из них можно было подключить максимальное число хостов. Какую маску подсети следует выбрать? (ответ приведите в десятичном и двоичном исчислении).
11. У вас имеется сеть класса С с адресом 192.168.19.0, содержащая четыре подсети. Вам необходимо установить максимально возможное число хостов на сегменте. Какую маску подсети следует выбрать? (ответ приведите в десятичном и двоичном исчислении).
12. У вас есть сеть класса В, разделенная на 30 подсетей. Вы хотите добавить 25 новых подсетей в ближайшие два года. При этом вам потребуется подключить к каждому сегменту до 600 хостов. Какую маску подсети следует

- выбрать? (ответ приведите в десятичном и двоичном исчислении).
13. Сеть 192.168.1.0 требуется разделить на 9 подсетей. При этом необходимо подключить к каждому сегменту максимально возможное число хостов. Какую маску подсети следует выбрать? (ответ приведите в десятичном и двоичном исчислении).
 14. У вас имеется сеть класса С с тремя подсетями. Вам необходимо добавить 2 новые подсети в ближайшие два года. Каждая сеть должна содержать 25 хостов. Какую маску подсети следует выбрать? (ответ приведите в десятичном и двоичном исчислении).
 15. В имеющемся у вас сетевом адресе класса С 192.168.88.0 необходимо выделить максимально возможное число подсетей, в каждой из которых должно быть до 12 хостов. Какую маску подсети следует выбрать? (ответ приведите в десятичном и двоичном исчислении).
 16. Вы выбрали маску подсети 255.255.255.248. Сколько подсетей и хостов вы получите?
 17. У вас есть сеть класса А и 22-битовая маска подсети. Сколько подсетей и хостов вы получите?
 18. У вас есть сеть класса А и 19-битовая маска подсети. Сколько подсетей и хостов вы получите?
 19. У вас есть сеть класса В и 10-битовая маска подсети. Сколько подсетей и хостов вы получите?
 20. У вас есть сеть класса С и 6-битовая маска подсети. Сколько подсетей и хостов вы получите?

ЛАБОРАТОРНОЕ ЗАДАНИЕ

1. Создать папку для хранения промежуточных и окончательных результатов данной работы на сервере X. В среде Linux запустить пакет Net-Simulator.

2. Построить модель сети, используя схему, подготовленную в результате выполнения первого задания предварительной подготовки. В данной схеме сети можно не подключать компьютеры к сетям, в состав которых входит более, чем один маршрутизатор.

4. Отрывать виртуальный терминал каждого компьютера и с

помощью команды *ifconfig* ввести параметры сетевых интерфейсов всех узлов сети.

5. С помощью команды *route* проконтролировать таблицы маршрутизации всех компьютеров. При необходимости дополнить таблицы маршрутами, обеспечивающими соединение маршрутизатора, определенного в задании, со всеми остальными узлами сети.

6. Открыть терминал маршрутизатора, заданного на этапе предварительной подготовки. С помощью команды *ping* проверить достижимость любого из узлов сети (всех сетей составной сети). В случае отсутствия связи с каким-либо узлом проверить параметры таблиц маршрутизации на промежуточных узлах.

Сравнить таблицу маршрутизации исходного маршрутизатора компьютерной модели с таблицей маршрутизацией узла, построенной в результате выполнения первого задания предварительной подготовки к работе. Сделать выводы.

7. Построить модель сети, используя схему, подготовленную в результате выполнения второго задания предварительной подготовки.

8. Отрыть виртуальный терминал каждого компьютера и с помощью команды *ifconfig* ввести параметры сетевых интерфейсов всех узлов сети.

С помощью команды *route* проконтролировать таблицы маршрутизации всех узлов. При необходимости дополнить таблицы маршрутами, обеспечивающими соединение всех узлов сети.

9. Сравнить таблицу маршрутизации одного из маршрутизаторов компьютерной модели с таблицей маршрутизацией, представленной во втором задании предварительной подготовки. При необходимости внести исправления в модель.

10. Построить модель сети, используя схему, подготовленную в результате выполнения третьего задания предварительной подготовки.

Отрыть виртуальный терминал каждого компьютера и с помощью команды *ifconfig* ввести параметры сетевых интерфейсов всех узлов сети.

С помощью команды *route* проконтролировать таблицы маршрутизации всех узлов. При необходимости дополнить таб-

лицы маршрутами, обеспечивающими соединение всех узлов сети.

11. С помощью команды *ping* проверить возможность соединения заданных узлов сети. Убедиться в правильности выбора наикратчайшего пути между указанными точками по схеме сети.

Литература

1. Олифер В.Г., Олифер Н.А. Компьютерные сети. С.-Петербург, изд-во «Питер».2007. - 957с.

2 Козлов В.Г., Семигук Е.С., Богомолов С.И. Программные средства систем связи: Учебное пособие. – Томск: ТМЦДО, 2008. – 162 с.