Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

А. Л. Гурский, Н. А. Певнева

ТЕЛЕКОММУНИКАЦИОННЫЕ И ИНФОРМАЦИОННЫЕ СИСТЕМЫ И СЕТИ. ЛАБОРАТОРНЫЙ ПРАКТИКУМ

*Рекомендовано УМО по образованию
в области информатики и радиоэлектроники в качестве
учебно-методического пособия для специальности
1-54 01 04 «Метрологическое обеспечение информационных систем и сетей»*

Минск БГУИР 2012

УДК [621.391+004.7](076)
ББК 32.88я73+32.973.202я73
Г95

Р е ц е н з е н т ы:

кафедра системного анализа и компьютерного моделирования
Белорусского государственного университета
(протокол №10 от 27.03.2012 г.);

доцент кафедры телекоммуникаций и информационных технологий
Белорусского государственного университета,
кандидат технических наук В. И. Емельяненков

Гурский, А. Л.

Г95 Телекоммуникационные и информационные системы и сети. Лабораторный практикум : учеб.-метод. пособие / А. Л. Гурский, Н. А. Певнева. – Минск : БГУИР, 2012. – 96 с. : ил.

ISBN 978-985-488-879-8.

Содержит краткие теоретические сведения по основам построения и функционирования компьютерных сетей и работе с программой Paket Tracer 4.1 для симуляции работы сетей с оборудованием фирмы Cisco, задания к лабораторным работам по конфигурированию сетей на примерах коммутационного оборудования фирмы Cisco и список литературы.

Для студентов специальности 1-54 01 04 «Метрологическое обеспечение информационных систем и сетей»

УДК [621.391+004.7](076)
ББК 32.88я73+32.973.202я73

ISBN 978-985-488-879-8

СОДЕРЖАНИЕ

Лабораторная работа №1

ОЗНАКОМЛЕНИЕ С ПРОГРАММОЙ PACKET TRACER. МОДЕЛИРОВАНИЕ СЕТЕЙ

Цель работы: ознакомиться с программным пакетом Packet Tracer 4.1, приемами работы с ним, приобрести навыки построения простейших сетей.

1.1 Теоретическая часть. Интерфейс Packet Tracer

1.1.1 Главное окно программы

Пользовательский оконный интерфейс программы состоит из следующих основных элементов (рисунок 1.1):

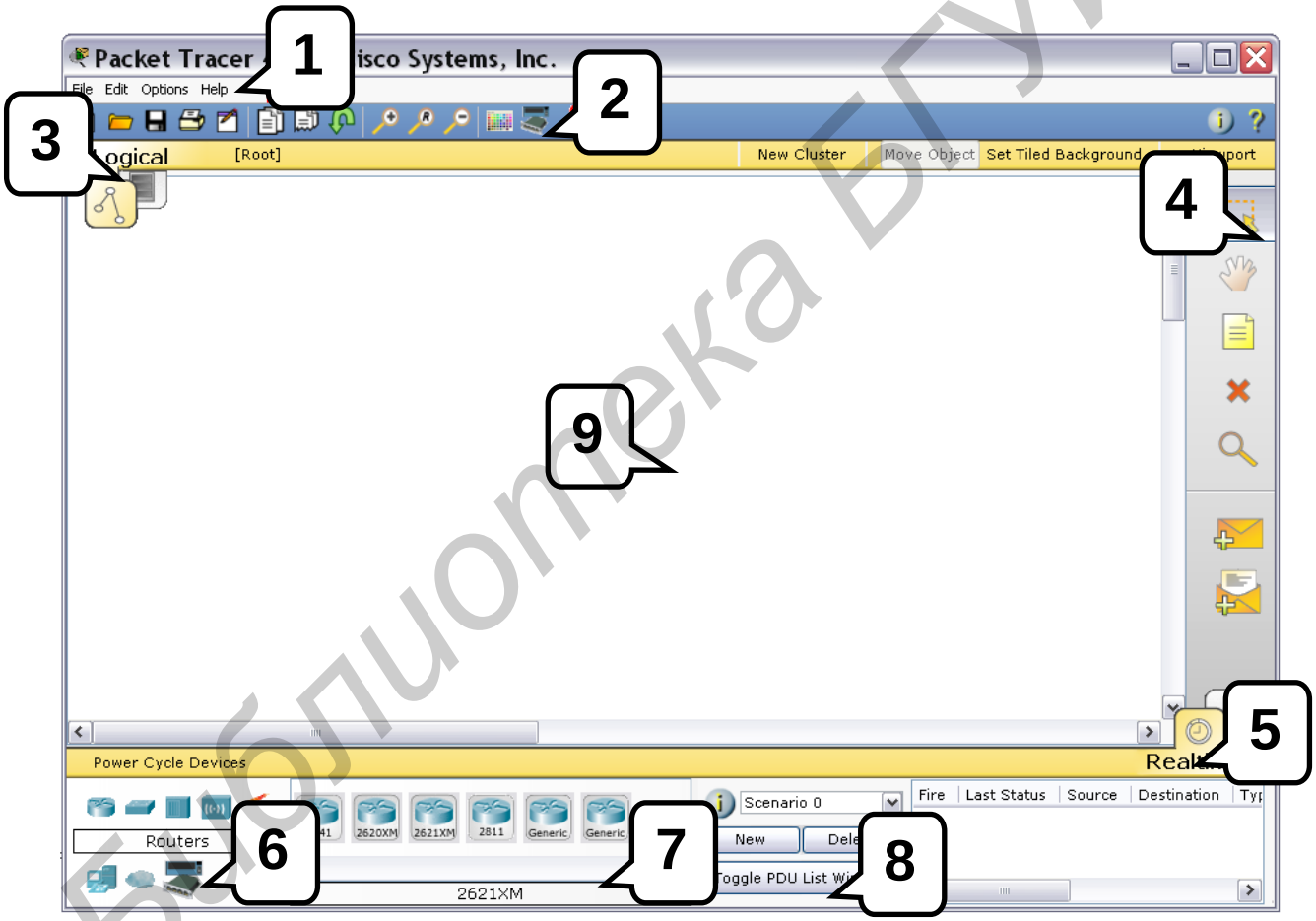


Рисунок 1.1 – Внешний вид интерфейса программы Packet Tracer

1 Главное меню программы:

– файл (File) – содержит операции открытия/сохранения документов;

– правка (Edit) – стандартные операции «копировать/вырезать, отменить/повторить»;

– настройки (Options) – работа с интерфейсом программы;

– помощь (Help) – справка об основных возможностях и командах программы.

2 Панель стандартных инструментов.

3 Переключатель между логической и физической организацией.

4 Панель инструментов, содержащая инструменты выделения, удаления, перемещения, масштабирования объектов, а также формирование произвольных пакетов.

5 Переключатель между режимом реального времени (Real-Time) и режимом симуляции.

6 Панель с группами сетевых устройств и линий связи.

7 Панель текущих групп сетевых устройств (коммутаторы, узлы, точки доступа, проводники и т. п.).

8 Панель создания пользовательских сценариев (окно наблюдения за пакетами визуального моделирования).

9 Рабочее пространство. Сюда методом «drag-and-drop» перетаскивается оборудование, из которого формируется сеть.

1.1.2 Оборудование и линии связи в Packet Tracer

1.1.2.1 Маршрутизаторы (Routers)

Маршрутизаторы используются, например, для поиска оптимального маршрута передачи данных на основании специальных алгоритмов маршрутизации, например выбор маршрута (пути) с наименьшим числом транзитных узлов. Могут решать и другие задачи. Работают на сетевом уровне модели OSI. Отличаются набором интерфейсов и возможностью установки плат расширения.

1.1.2.2 Коммутаторы (Switches)

Коммутаторы – это устройства, работающие на канальном уровне модели OSI и предназначенные для объединения нескольких узлов в пределах одного или нескольких сегментов сети. Коммутатор передает пакеты на основании внутренней таблицы – таблицы коммутации, следовательно, информация передается только на тот порт, к которому подключен сетевой интерфейс с адресом назначения, а не повторяется на всех портах (как в концентраторе).

1.1.2.3 Концентраторы (Hubs)

Устройство, объединяюшее сетевые узлы. В технологии Ethernet повторяет пакет, принятый на одном порту, на всех остальных портах.

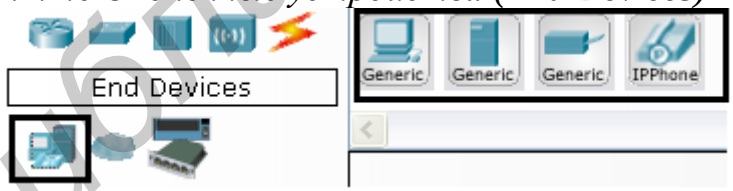
1.1.2.4 Беспроводные устройства (Wireless Devices)

Устройства, реализующие беспроводные технологии Wi-Fi и сети на их основе, включая точки доступа.

1.1.2.5 Линии связи (Connections)

С помощью этих компонентов узлы соединяются в единую схему. Среди этих элементов есть автоматическое соединение (1), консольный кабель (2), прямой патч-корд (компьютер – коммутатор, маршрутизатор – коммутатор) (3), кроссовый патч-корд (компьютер – компьютер, коммутатор – коммутатор, маршрутизатор – маршрутизатор, маршрутизатор – компьютер) (4), оптоволоконные линии и т. п.

1.1.2.6 Оконечные устройства (End Devices)



На этой консоли представлены непосредственно конечные узлы: хосты, серверы, принтеры, телефоны.

1.1.2.7 Эмуляция глобальных сетей (WAN Emulation)

На данной панели представлены элементы для эмуляции глобальной сети: модем DSL, «облако» и т. д.

1.1.2.8 Устройства, комплектуемые пользователем (Custom Made Devices)

Здесь размещаются пользовательские устройства, которые можно комплектовать самостоятельно и сохранять для последующей работы.

1.1.3 Физическое комплектование оборудования в Packet Tracer

Физическое комплектование заключается в дополнении устройства его модульными составляющими и последующей их настройке.

Для примера устанавливаем на рабочем поле маршрутизатор 1841. Кликом на его символьном изображении открываем его физическую конфигурацию (рисунок 1.2).

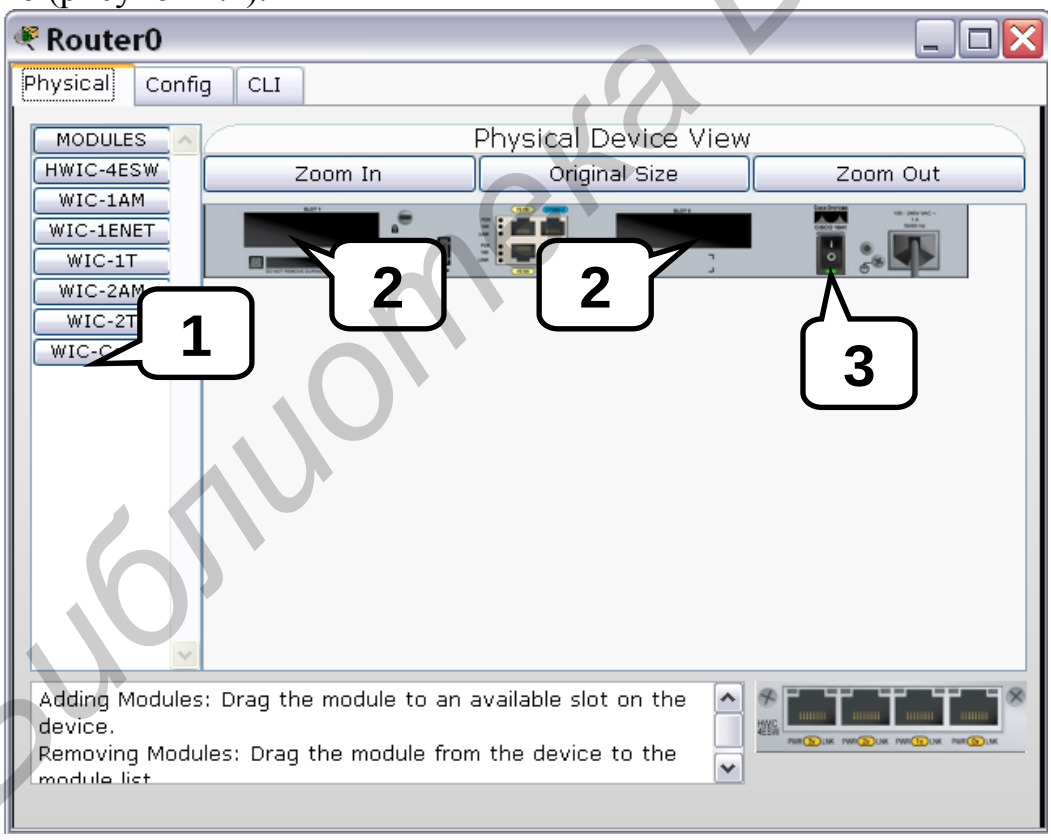


Рисунок 1.2 – Окно конфигурации маршрутизатора

Слева представлен список модулей (1), которыми можно укомплектовать данный маршрутизатор. В пустые слоты (2) можно вставить эти модули. Все манипуляции с маршрутизатором необходимо проводить при выключенном питании (кнопка 3).

Рассмотрим подробнее модули, показанные на рисунке 1.2.

HWIC-4ESW – высокопроизводительный модуль с четырьмя коммутационными портами Ethernet под разъем RJ-45. Позволяет сочетать возможности маршрутизатора и коммутатора.

WIC-1AM включает в себя два разъема RJ-11, используемых для подключения к телефонным сетям. Карта использует один порт для соединения с телефонной линией, другой может быть подключен к аналоговому телефону для звонков во время простоя модема.

WIC-1ENET – это однопортовая 10 Мб/с Ethernet карта для 10BASE-T Ethernet LAN.

WIC-1T предоставляет средства однопортового последовательного подключения к удаленным офисам или устаревшим сетевым устройствам, например SDLC концентраторам, системам сигнализации и устройствам packet over SONET (POS).

WIC-2AM содержит два разъема RJ-11, используемых для подключения к телефонным сетям. В WIC-2AM два модемных порта, что позволяет использовать оба канала для соединения одновременно.

WIC-2T-2-портовый синхронный/асинхронный последовательный сетевой модуль. Предоставляет гибкую поддержку многих протоколов с индивидуальной настройкой каждого порта в синхронный или асинхронный режим. При применении для синхронной/асинхронной поддержки порты представляют:

– низкоскоростную агрегацию (до 128 Кб/с);

– поддержку dial-up модемов;

– синхронные или асинхронные соединения с портами управления другого оборудования и передачу по устаревшим протоколам типа Bi-sync и SDLC.

WIC-Cover – стенка для WIC слота, необходима для защиты электронных компонентов и для улучшения циркуляции охлаждающего воздушного потока.

1.1.4 Моделирование сетей в Packet Tracer

В качестве примера промоделируем работу компьютера и сервера. Соединение производится кроссовым кабелем (рисунок 1.3).

Щелчком мыши по изображению компьютера заходим в окно PC0, выбираем вкладку Desktop, заходим в меню IP Configuration и в открывшемся окне вводим адреса, приведенные на рисунке 1.4.

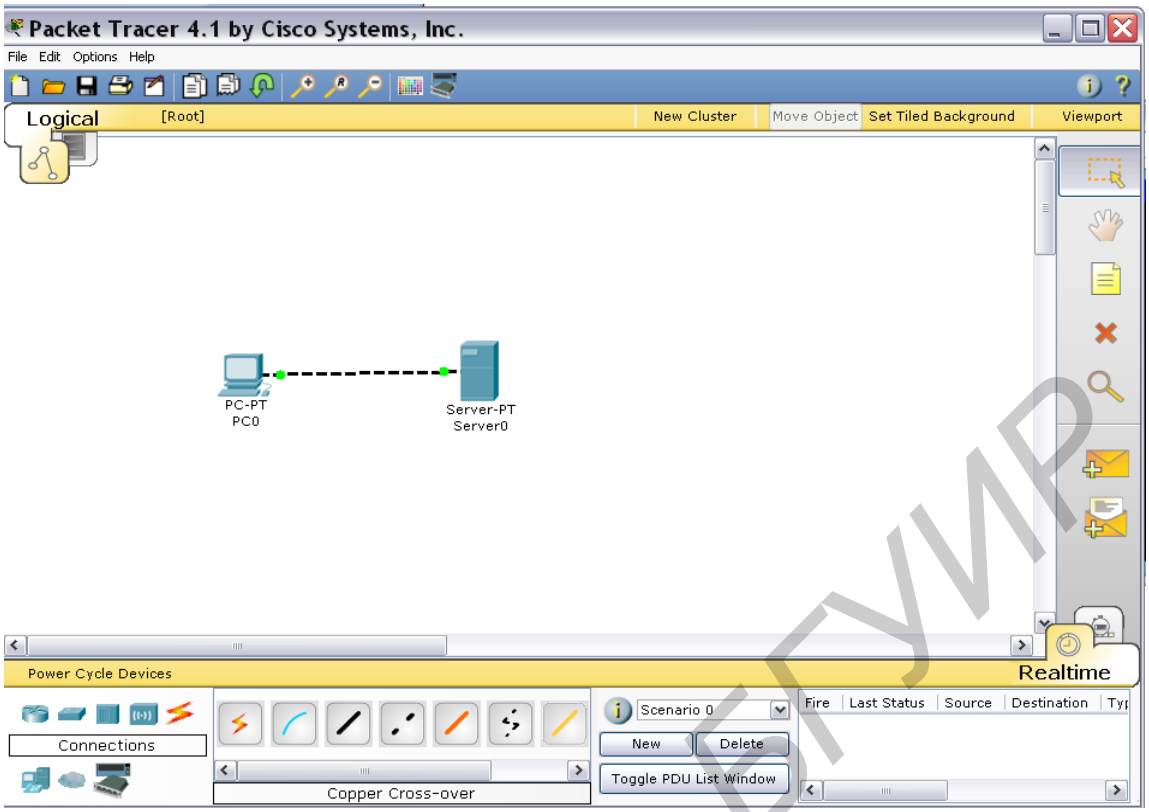


Рисунок 1.3 – Сеть, состоящая из компьютера и сервера

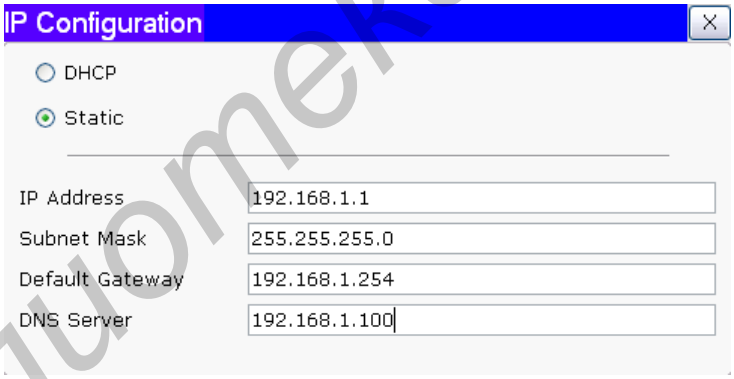


Рисунок 1.4 – Окно конфигурирования сетевого интерфейса компьютера

Теперь переходим к настройке сервера. Щелчком мыши по изображению сервера заходим в окно Server0, выбираем вкладку Config, в меню INTERFACE открываем закладку FastEthernet и вводим адрес, приведенный на рисунке 1.5.

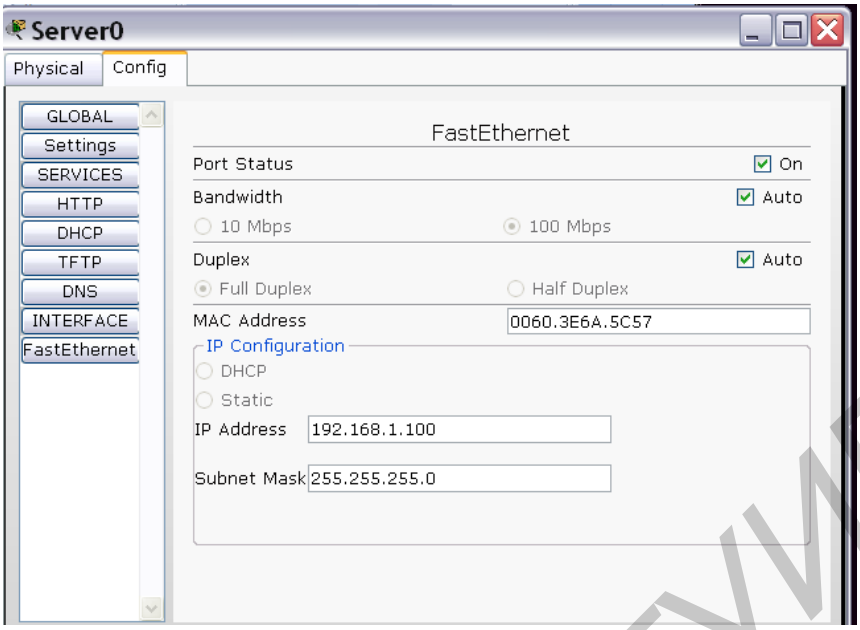


Рисунок 1.5 – Окно конфигурирования сетевого интерфейса сервера

Далее открываем закладку DNS и вводим данные, приведенные на рисунке 1.6.

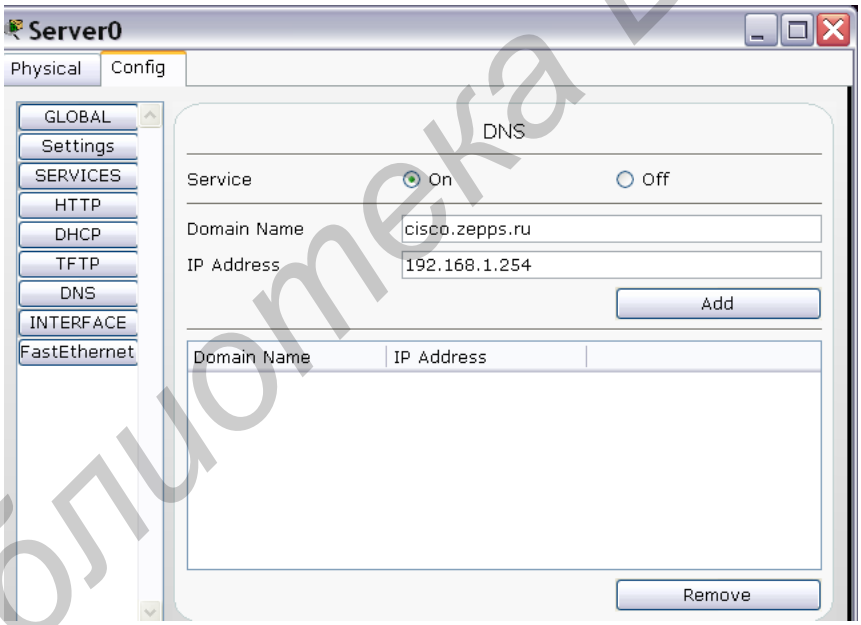


Рисунок 1.6 – Окно конфигурирования службы DNS на сервере

После этого нажимаем кнопку Add, закрываем окно сервера и возвращаемся в окно PC0 компьютера. Протестируем связность сети (связность означает физическое наличие соединений и возможность обмена информацией между всеми узлами сети) с помощью предназначенной для этого утилиты PING. Выбираем вкладку Desktop, заходим в меню Command Prompt и в открывшемся окне в командную строку вводим ping 192.168.1.100, нажимаем Enter. Результат выполнения команды представлен на рисунке 1.7.

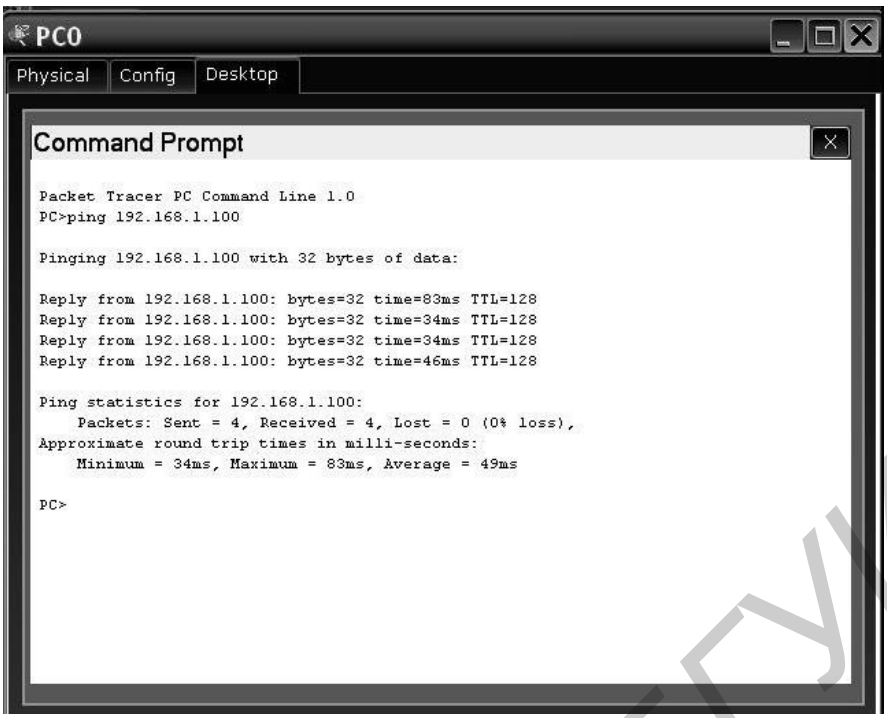


Рисунок 1.7 – Результат выполнения команды ping 192.168.1.100

Теперь соединим компьютер с сервером через коммутатор, при этом используем прямой (некроссированный) кабель, как показано на рисунке 1.8.

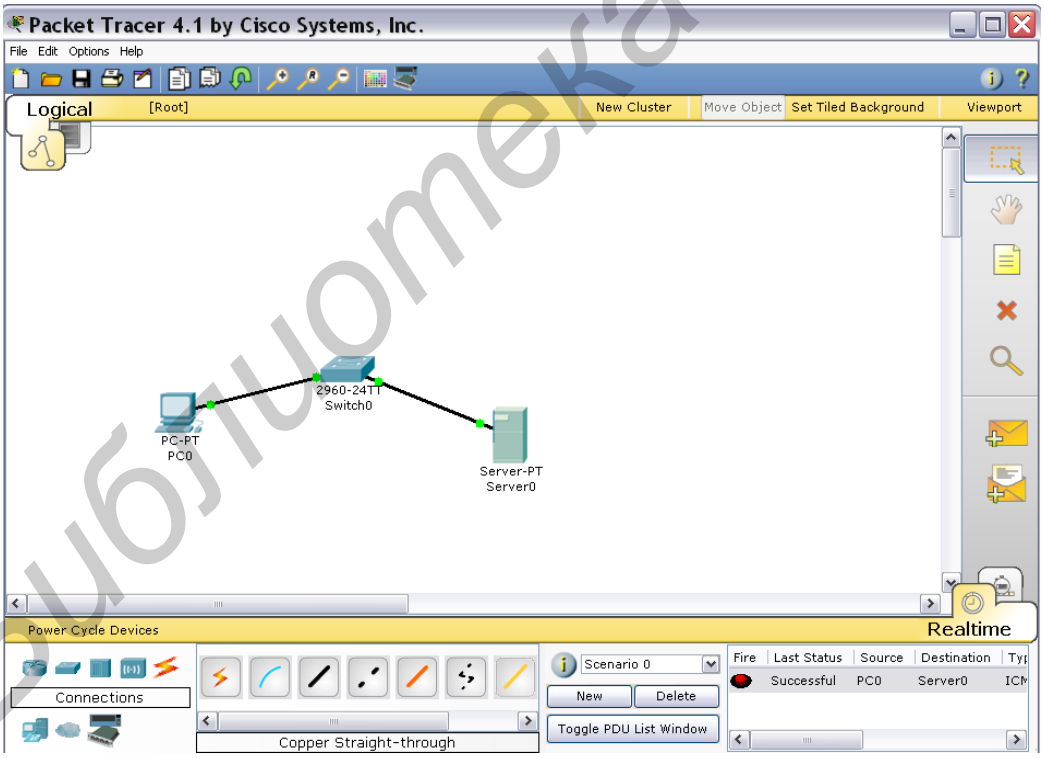


Рисунок 1.8 – Сеть, состоящая из компьютера, коммутатора и сервера

Необходимо подождать, пока соединительные точки загорятся зеленым цветом, после этого можно проверить прохождение информации через сеть. Для этого выбираем значок  с правой боковой панели окна программы и щелкаем им на изображениях компьютера и сервера, после чего в нижнем

правом углу рабочего окна в колонке Last Status должна появиться надпись об успешном завершении проводимой операции. Усложним схему, подключив через маршрутизатор еще один сервер, который будет олицетворять выход в Internet. Схема примет вид, показанный на рисунке 1.9.

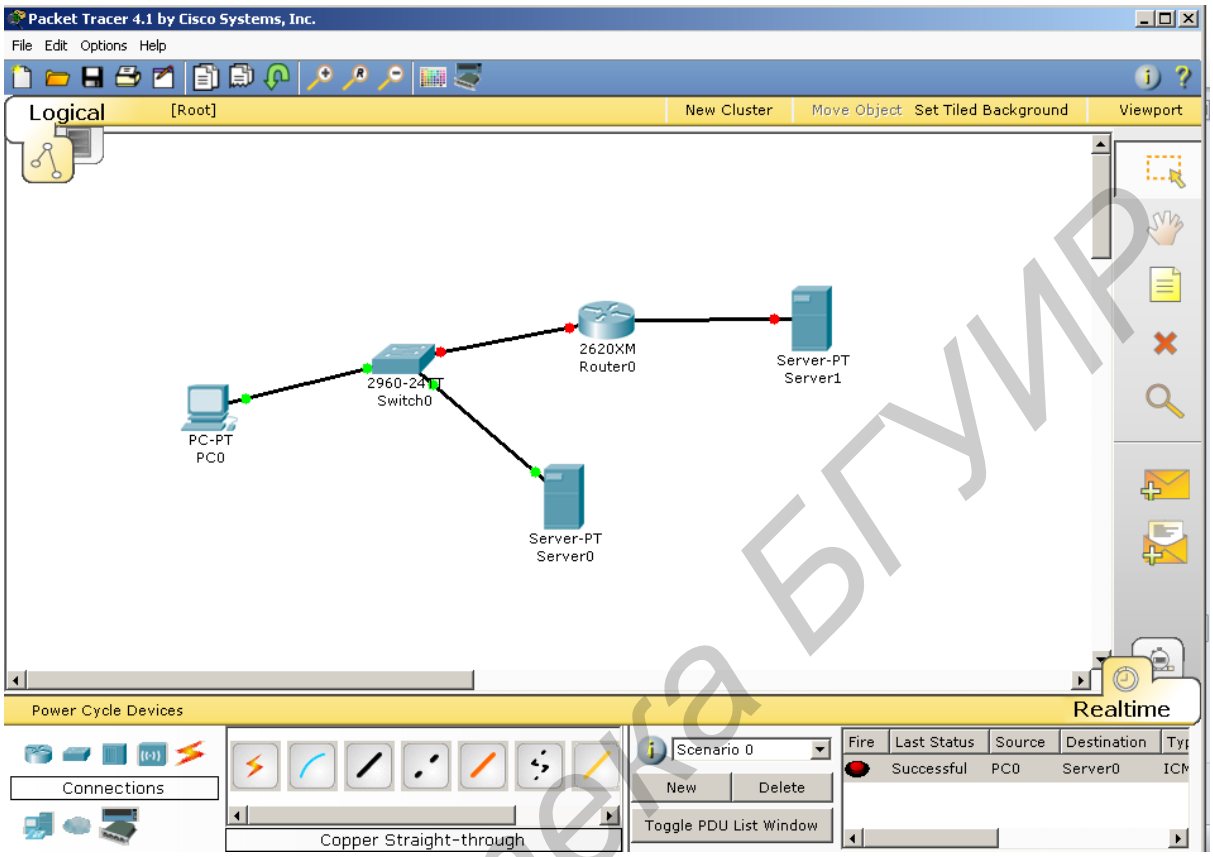


Рисунок 1.9 – Сеть, в которую добавлены маршрутизатор и сервер

На данной стадии необходимо заполнить поля адресов в новом сервере. Для этого щелкаем мышью по его изображению, во вкладке Config выбираем закладку FastEthernet и записываем адрес, представленный на рисунке 1.10.

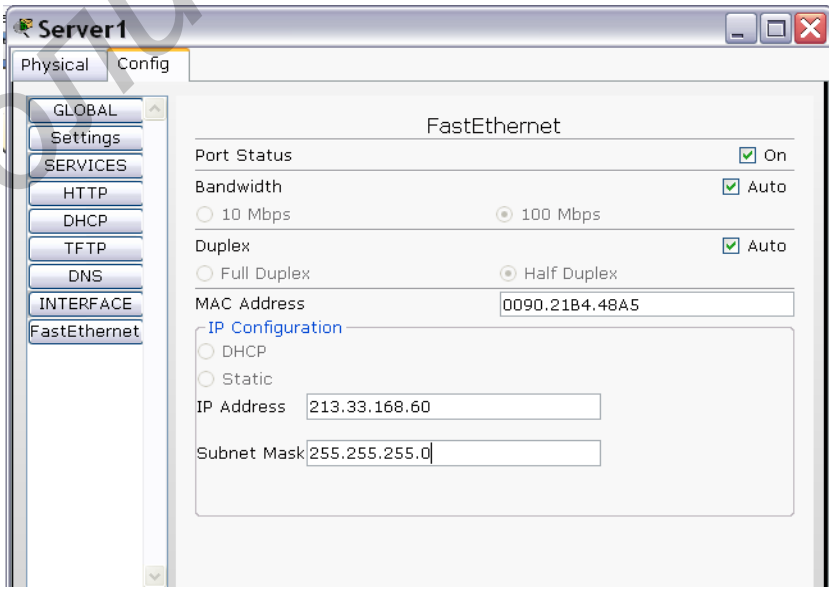


Рисунок 1.10 – Конфигурирование интерфейса FastEthernet сервера 1

Далее переходим на закладку Settings и заносим еще один адрес (рисунок 1.11).

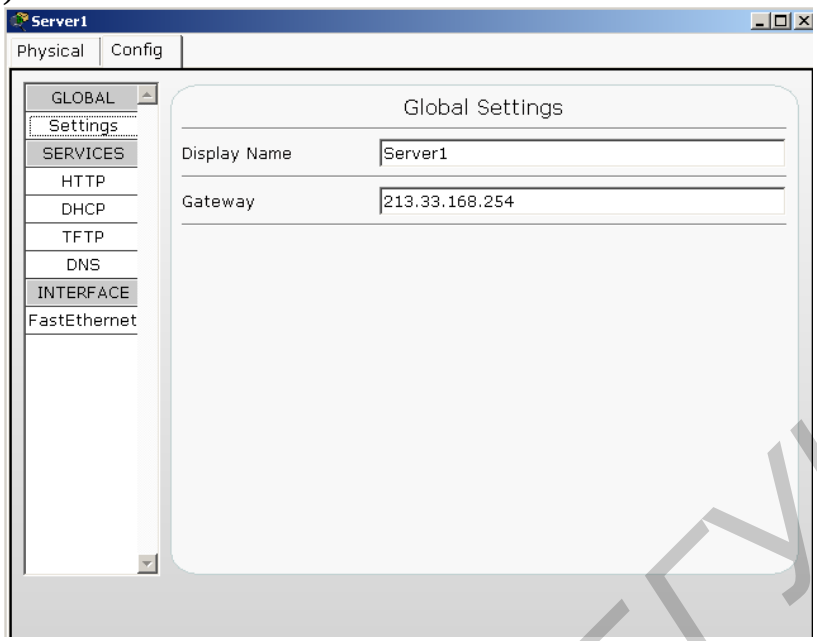


Рисунок 1.11 – Конфигурирование адреса шлюза на сервере 1

Имитируем содержимое простейшей веб-страницы. Во вкладке HTTP в строке <hr> вместо фразы Welcome to Packet Tracer 4.1, the best thing since..... Packet Tracer 4.0. пишем, например, Welcome to the Internet!!!, как показано на рисунке 1.12.

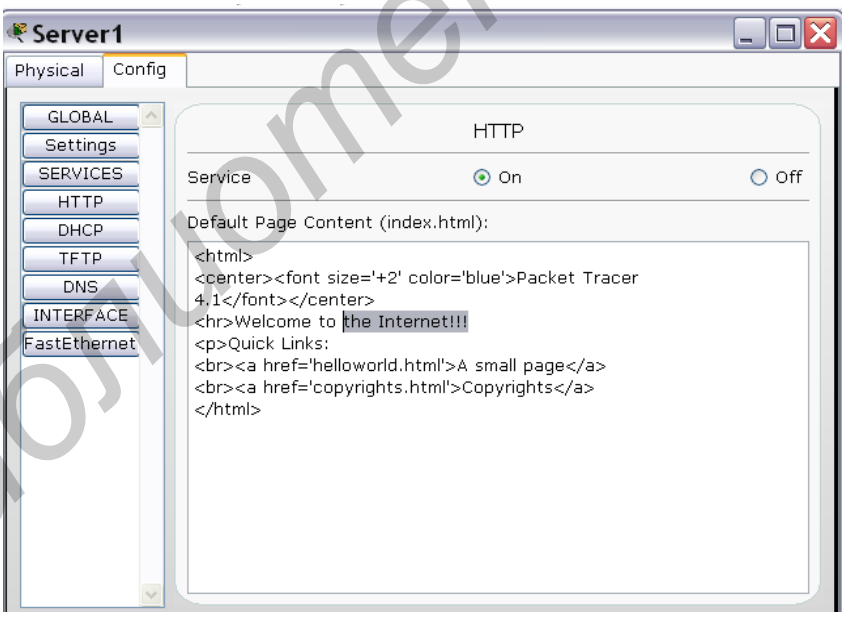


Рисунок 1.12 – Фрагмент гипертекстового документа для имитации простейшей веб-страницы

Теперь настроим маршрутизатор. Для этого щелкаем по его изображению клавишей мыши и переходим во вкладку CLI (command line interface, интерфейс командной строки). В командной строке Continue with configuration dialog? [yes/no]: пишем «no» и нажимаем клавишу Enter.

Приведем текст настройки маршрутизатора:

Router>ena
Router#conf ter

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#interface fastEthernet 0/0
Router(config-if)#ip address 192.168.1.254 255.255.255.0
Router(config-if)#no shutdown

LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up

Router(config-if)#description Interface_To_Local_Network
Router(config-if)#exit
Router(config)#interface fastEthernet 0/1
Router(config-if)#ip address 213.33.168.254 255.255.255.0
Router(config-if)#no shutdown

LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

Router(config-if)#description Interface_To_Internet
Router(config-if)#exit
Router(config)#exit

SYS-5-CONFIG_I: Configured from console by console

Router#copy running-config startup-config

Destination filename [startup-config]?
Building configuration...
[OK]
Router#

Возвращаемся к рабочему окну нашей сети. Обращаем внимание на то, что красным цветом подсвечивается связь между маршрутизатором и интернет-сервером. Для устранения этой неполадки меняем между ними прямой кабель на кроссовый.

Далее открываем окно Server0, вкладку Config, в закладке DNS в строку Domain Name вводим адрес server.internet.ru, а в строку IP Address вводим 213.33.168.60. Затем нажимаем кнопку Add, закрываем окно сервера и возвращаемся к рабочей зоне нашей сети.

Открываем окно PC0, вкладку Desktop, выбираем меню Web Browser, в строку URL вводим адрес server.internet.ru, нажимаем Enter. Результат работы отображен на рисунке 1.13.

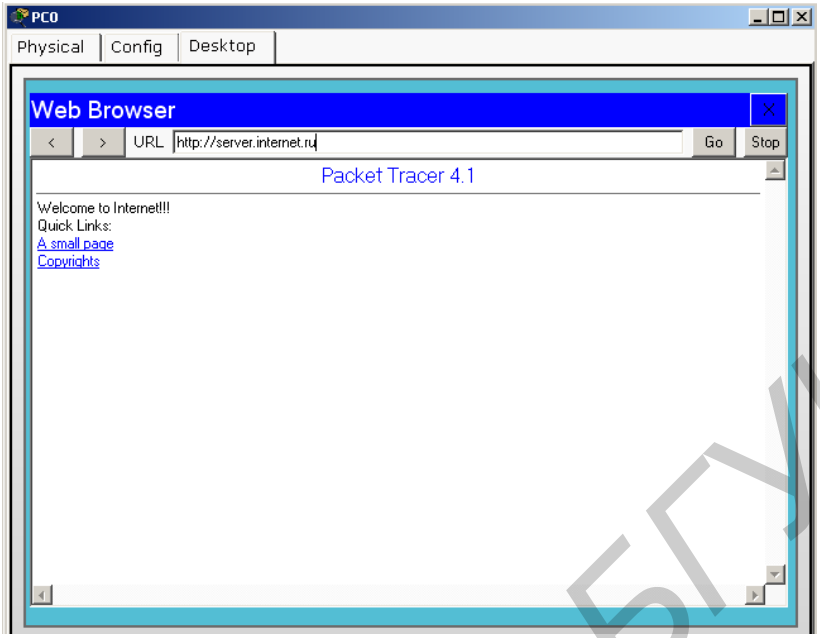


Рисунок 1.13 – Оконо имитатора веб-браузера в Packet Tracer

Чтобы еще раз убедиться, что информация проходит через всю сеть, воспользуемся в окне PC0 вкладкой Desktop, в которой выбираем меню Command Prompt. В командной строке вводим информацию tracert server.internet.ru, нажимаем Enter. Отображенный в окне результат (рисунок 1.14) свидетельствует об успешной работе сети.

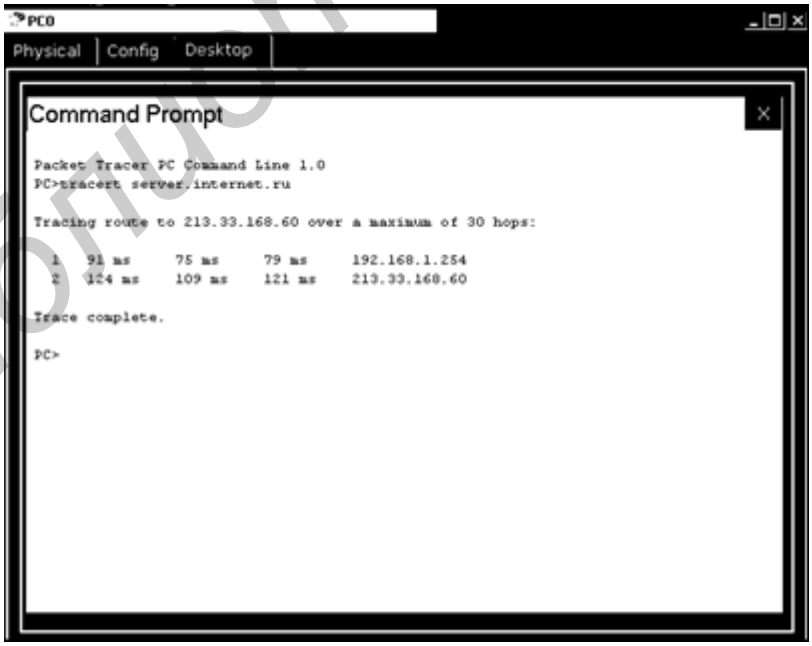


Рисунок 1.14 – Результат выполнения команды «tracert (trace route)»

Теперь соединим компьютер с маршрутизатором через консольный кабель, при этом компьютер используется в качестве терминала (консоли) маршрутизатора (рисунок 1.15).

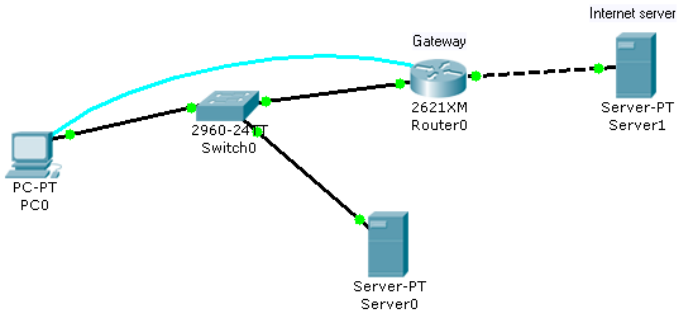


Рисунок 1.15 – Консольное подключение компьютера к маршрутизатору

Посмотрим, какую информацию выдает маршрутизатор о нашей сети. Для этого заходим в закладку CLI окна Router0 и в командной строке вводим информацию show ip interface brief. Наблюдаем следующий ответ:

Router#show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	192.168.1.254	YES	manual	up	up
FastEthernet0/1	213.33.168.254	YES	manual	up	up

Делаем вывод об исправности работы сети.

Чтобы сделать сложный запрос о работе сети, воспользуемся кнопкой  на правой боковой панели рабочего окна программы, появившимся указателем в виде белого конверта щелкаем по изображению компьютера. Запустим эквивалент утилиты PING. Для этого в открывшееся окно заносим следующие данные (рисунок 1.16).

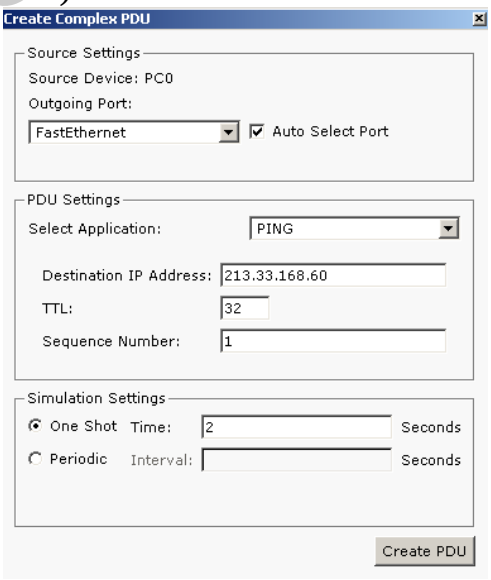
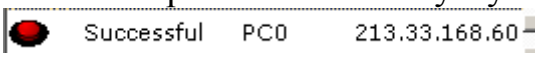


Рисунок 1.16 – Формирование ping-запроса в окне Create Complex PDU

В правом нижнем углу окна программы высветилась информация



«Зацикленный» пинг удобно применять при настройке сети, чтобы при любых изменениях можно было видеть, не пропала ли связность между хостами.

1.2 Задание к лабораторной работе

1.2.1 Ознакомиться с пользовательским интерфейсом программы Packet Tracer 4.1, ее службой справки, выполнить действия по пункту 1.1.4 и убедиться в работоспособности сети. Продемонстрировать работу сети преподавателю.

1.3 Содержание отчета

1 Цель работы.
2 Схема топологии сети.
3 Пример конфигурации компьютеров.
4 Вывод.

1.4 Контрольные вопросы

1 Укажите основные компоненты пользовательского интерфейса программы Packet Tracer и покажите их расположение в окне программы.
2 Какие типы соединений используются в программе Packet tracer?
3 Какие типы оборудования доступны в программе Packet Tracer?
4 Что нужно сделать для проверки прохождения информации от одного узла к другому?
5 Что такое связность сети?

Лабораторная работа №2

IP АДРЕСАЦИЯ. ДЕЛЕНИЕ СЕТЕЙ НА ПОДСЕТИ

Цель работы: изучить общие принципы IP-адресации и классы IP-адресов. Получить практический навык деления сетей на подсети.

2.1 Теоретическая часть

2.1.1 IP-адреса

Для того чтобы любые две системы могли взаимодействовать между собой, они должны иметь возможность однозначно идентифицировать друг друга. Несмотря на то что показанные адреса не являются фактическими сетевыми адресами, они демонстрируют концепцию группировки адресов. В повседневной жизни имена или номера (такие как номера телефонов) часто используются в качестве уникальных идентификаторов. Аналогично этому каждый компьютер в TCP/IP-сети обязан иметь как минимум один уникальный идентификатор или адрес. Такой адрес позволяет одному компьютеру в сети находить другой.

Компьютеры хранят IP-адрес в виде 32-битной последовательности единиц и нулей (рисунок 2.1). Для простоты использования IP-адрес обычно записывается в виде четырех десятичных номеров, разделенных точками. Предположим, адрес одного из компьютеров – 192.168.1.2. Второй компьютер может иметь адрес 128.10.2.1. Такой способ написания адреса называется **точечно-десятичным форматом.** В таком виде каждый IP-адрес состоит из четырех частей, разделенных точками. Каждая из частей называется **октетом,** поскольку состоит из восьми двоичных цифр. Октет эквивалентен байту. Например, адресу 192.168.1.8 соответствует запись 11000000.10101000.00000001.00001000 в двоичном представлении. Человек легче воспринимает точечно-десятичный формат, чем двоичные ноли и единицы. Этот формат помогает также избежать ошибок из-за перестановки цифр, что часто случается при использовании двоичных номеров.

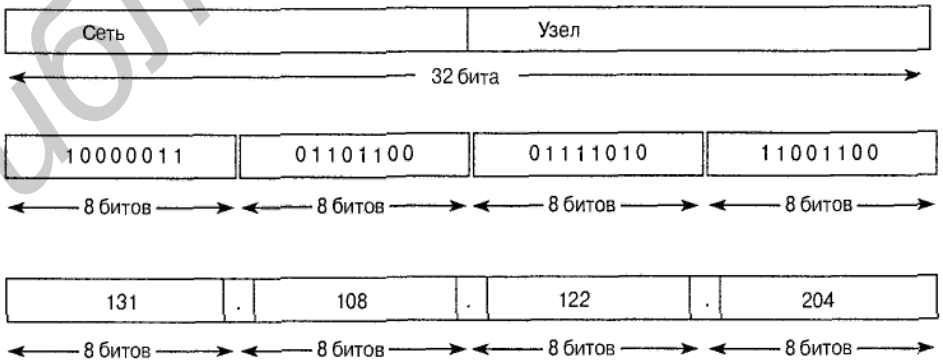


Рисунок 2.1 – Формат IP адреса

Точечно-десятичный формат позволяет намного быстрее различить цифровые составляющие адреса (см. рисунок 2.1). И двоичный, и десятичный номера на рисунке соответствуют одному и тому же адресу, но в десятичном

формате он выглядит намного проще и, несомненно, короче. Ошибки – одна из наиболее общих проблем при работе с двоичными адресами. В длинных последовательностях из нолей и единиц легко ошибиться, поменять цифры местами или что-либо пропустить. Иными словами, намного проще увидеть связь между такими двумя номерами:
192.168.1.8 и 192.168.1.9,
чем распознать ту же связь в двоичных эквивалентах тех же адресов:
11000000.10101000.00000001.00001000 и
11000000.10101000.00000001.00001001.

Глядя на двоичную форму записи двух адресов, практически невозможно понять, что представляют из себя последовательные номера узлов.

2.1.2 Классы IP-адресов

В общем случае IP-адрес состоит из номера сети (левая часть) и номера узла в этой сети (правая часть). В зависимости от того, где проходит граница между номером сети и номером узла, адреса делятся на классы.

Адреса класса А (рисунок 2.2) предназначены для очень больших сетей. В адресе класса А используется только первый октет в качестве идентификатора сети. Оставшиеся три октета выделены для перечисления адресов узлов.

Рисунок 2.2 – Начальные биты, образующие классы IP-адресов

Первый бит в адресе класса А всегда равен 0. Учитывая это, наименьшее допустимое число будет равно 00000000 (десятичный 0), а наибольшее – 01111111 (десятичное число 127). Следует заметить, что оба номера, 0 и 127, являются зарезервированными и не могут быть использованы в качестве сетевых адресов. Любые адреса, начинающиеся с числа в диапазоне от 1 до 126 в первом октете, являются адресами класса А (рисунок 2.3).

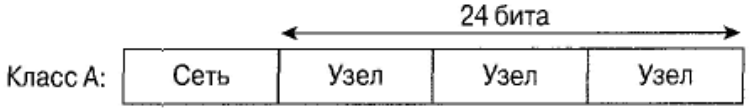


Рисунок 2.3 – Адреса класса А

Сеть с номером 127.0.0.0 зарезервирована для обратного петлевого (loopback) тестирования (маршрутизаторы или локальные узлы могут использовать его для передачи пакетов самим себе). Этот адрес носит название «адрес обратной петли». Следовательно, такой адрес не может быть присвоен сети.

Адреса класса В используются для сетей среднего и крупного размера (рисунок 2.4). В IP-адресе класса В используются два первых октета для сетевого адреса. Оставшиеся два октета представляют адрес узла.

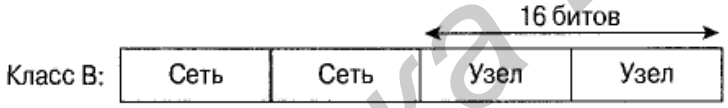


Рисунок 2.4 – Адреса класса В

Первые два бита первого октета всегда равны 10, оставшиеся 6 битов могут содержать любые комбинации нулей и единиц. Таким образом, наименьшее число, которое может быть использовано для адресов этого класса, равно 10000000 (десятичное 128), и наибольшее — 10111111 (десятичное значение равно 191). Любые адреса, содержащие в первом октете числа от 128 до 191, являются адресами класса В.

Адреса класса С (рисунок 2.5) – это наиболее часто используемые из исходных классов адресов. Данный класс адреса предназначен для использования в малых сетях.

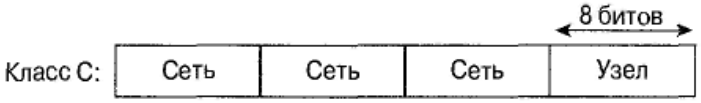


Рисунок 2.5 – Адреса класса С

Адрес этого класса начинается с двоичной комбинации 110. Таким образом, наименьшее доступное число – 11000000 (десятичное 192), а наибольшее – 11011111 (десятичное значение 223). Если адрес в первом октете содержит числа от 192 до 223, значит, он относится к классу С.

Адреса класса D (рисунок 2.6) были созданы для реализации в IP-адресах механизма групповой рассылки. **Групповым адресом (multicast address)** называется уникальный сетевой адрес, используемый для отправки па-

кетов, содержащих адрес рассматриваемого класса в поле получателя, предопределенным группам сетевых устройств. Таким образом, одна сетевая станция может передавать один поток данных нескольким получателям.

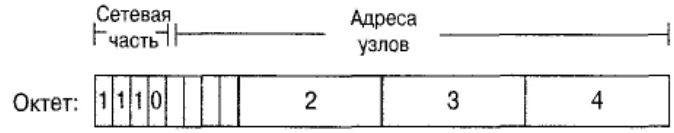


Рисунок 2.6 –Адреса класса D

Диапазон адресов класса D, так же, как и других классов, определенным образом ограничен. Первые четыре бита адреса класса D должны быть равны 1110. Следовательно, первый октет адресов этого класса может принимать значения от 11100000 до 11101111 или, в десятичной записи, от 224 до 239. Групповой IP-адрес, первый октет которого начинается с чисел в диапазоне от 224 до 239, является адресом класса D.

Адреса класса E (рисунок 2.7) также были описаны в стандартах и выделены в отдельный блок. Однако они были зарезервированы проблемной группой проектирования Internet (Internet Engineering Task Force – IETF) для собственных исследовательских нужд. В результате адреса класса E никогда не использовались в сети Internet. Первые четыре бита адресов класса E всегда содержат 1. Следовательно, значение первого октета находится в диапазоне от 11110000 до 11111111 или от 240 до 255 – в десятичном виде.

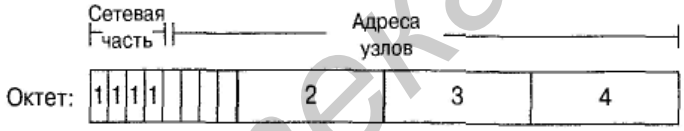


Рисунок 2.7 – Адреса класса E

Диапазоны значений первого октета в IP-адресах для каждого из классов приведены в таблице 2.1.

Таблица 2.1 – Классы IP адресов. Диапазон значений первого октета

Класс IP-адреса	Диапазон IP-адресов	
КлассA	от 1 до 126	от 00000001 до 01111111
Класс B	от 128 до 191	от 10000000 до 10111111
КлассC	от 192 до 223	от 11000000 до 11011111
Класс D	от 224 до 239	от 11100000 до 11101111
Класс E	от 240 до 255	от 11110000 до 11111111

2.1.3 Зарезервированные IP-адреса

Некоторые адреса являются зарезервированными и не могут быть присвоены сетевым устройствам. К ним относятся следующие:

– сетевые адреса, идентифицирующие саму сеть;

– как следует из названия, **широковещательный адрес** используется для широковещательной рассылки всем сетевым устройствам в данной сети.

IP-адрес, у которого все биты, отведенные под адрес узла, заполнены нулями, зарезервирован под **адрес сети** (рисунок 2.8). Показанный адрес класса B имеет нули во всех битах, отведенных под адрес узла. Таким образом, в примере для сети класса A число 113.0.0.0 является адресом сети, содержащей узел 113.1.2.3. Маршрутизатор использует IP-адрес сети при пересылке данных через сеть Internet. Примером для сети класса B может служить адрес 176.10.0.0, показанный на рисунке 2.7.

Рисунок 2.8 – Структура адреса сети

Для адреса сети класса B, записанного в виде чисел в точечно-десятичном формате, первые два октета стандартно идентифицируют сеть. Последние два октета содержат нули, поскольку именно эти 16 битов являются той частью адреса, которая отведена для идентификации подключенных к сети устройств. Такой адрес называется одноадресатным (unicast), где «uni» обозначает «один». Одноадресатный адрес указывает только на один узел во всей сети. IP-адрес из рассмотренного выше примера (176.10.0.0) зарезервирован в качестве адреса сети и ни при каких условиях не может быть использован в качестве адреса подключенного к сети устройства. Примером IP-адреса сетевого устройства в сети 176.10.0.0 может быть 176.10.16.1. В данном примере 176.10 является сетевой частью адреса, а 16.1 – это часть, обозначающая узел.

Для передачи данных всем узлам в сети требуется широковещательный адрес. Широковещательная рассылка используется, когда отправитель пересылает данные всем устройствам в сети (рисунок 2.9).

Рисунок 2.9 – Широковещательный адрес

Адрес класса B, который показан на рисунке 2.9 внизу, является широковещательным для данной сети. Когда пакеты будут получены в соответствии с широковещательным адресом получателя, данные будут обработаны на каждом из компьютеров. Чтобы быть уверенным в том, что все устройства в сети получили и обработали пакеты широковещательной рассылки, отправитель должен использовать специальный IP-адрес, который будет понят и правильно обработан остальными устройствами. В широковещательных IP-адресах все биты, отведенные под адрес узла (поле узла), равны единице.

Для сети с адресом 176.10.0.0, в котором последние 16 битов формируют поле узла (или отведенную для узла часть адреса), адресом широковещательной рассылки, по которому пакеты будут отправлены всем сетевым устройствам, является адрес 176.10.255.255 (поскольку десятичное число 255 соответствует двоичному октету 11111111).

2.1.4 Открытые и частные адреса

Стабильное функционирование сети Internet зависит от уникальности используемых в сети публичных адресов. Наличие дублирующихся адресов могло бы привести к нестабильности работы сети Internet и дополнительной нагрузке на устройства из-за доставки пакетов сетям, использующим дублирующиеся адреса.

Открытые IP-адреса уникальны. Не существует двух устройств с одинаковыми IP-адресами, которые были бы подключены к открытой сети, поскольку такие адреса используются в глобальном масштабе и подчиняются стандарту. Все компьютеры, подключенные к сети Internet, следуют такому требованию. Открытые IP-адреса должны выделяться поставщиками услуг Internet (Internet Service Provider – ISP) или регистрироваться за определенную плату.

Вследствие быстрого роста сети Internet количество незанятых IP-адресов уменьшается, и в настоящее время ресурс централизованно выделяемых адресов формата IPv4 (4-байтных) исчерпан. Поэтому появляются новые схемы адресации, такие как бесклассовая междоменная маршрутизация (Classless InterDomain Routing – CIDR) и IPv6, призванные помочь решить проблему исчерпания адресного пространства. Технологии CIDR и IPv6 подробно будут рассмотрены ниже. Чтобы частично решить проблему нехватки адресного пространства, был разработан альтернативный вариант – использование частных IP-адресов (таблица 2.2). Как уже говорилось, узлы в сети Internet должны иметь глобально-уникальные адреса. Однако частные сети, не подключенные к открытой сети, могут использовать любые действительные адреса, которые должны быть уникальны только внутри локальной сети. Многие частные сети используются совместно с открытыми сетями, поэтому использование выбранных произвольно адресов настоятельно не рекомендуется, поскольку однажды частная сеть может оказаться подключенной к глобальной сети Internet.

В спецификации RFC 1918 выделены три блока IP-адресов (один адрес класса A, серия адресов класса B и набор адресов класса C) для внутреннего использования в частных сетях. Адреса из этих диапазонов не передаются магистральными маршрутизаторами сети Internet, и пакеты с адресами из частных сетей немедленно будут отброшены такими устройствами.

В том случае, когда нужно выбрать схему адресации для внутренней сети предприятия или домашней сети, можно использовать диапазоны адресов, перечисленные в таблице 2.2, вместо глобально уникальных. Частные IP-адреса могут использоваться совместно с публичными для внутренних соединений, что позволяет экономить открытые уникальные адреса.

Таблица 2.2 – Частные IP-адреса

Класс IP-адреса	Диапазон адресов (для внутреннего использования)
Класс A	от 10.0.0.0 до 10.255.255.255
Класс B	от 172.16.0.0 до 176.31.255.255
Класс C	от 192.168.0.0 до 192.168.255.255

При подключении сети предприятия, в которой используются частные адреса, к сети Internet необходимо обеспечить преобразование частных адресов в открытые. Такой процесс называется трансляцией сетевых адресов (Network Address Translation – NAT) и обычно выполняется маршрутизатором.

2.1.5 Подсети

Еще одним способом экономии IP-адресов, который используется наряду с уже упомянутыми выше технологиями CIDR, адресацией IPv6 и частными адресами, является **механизм использования подсетей (subnetting).** Этот метод позволяет разбивать полные классовые блоки сетевых адресов на меньшие и помогает избежать полного исчерпания IP-адресов. На рисунке 2.10 показана сеть класса B (131.108.0.0), которая разбита на три подсети.

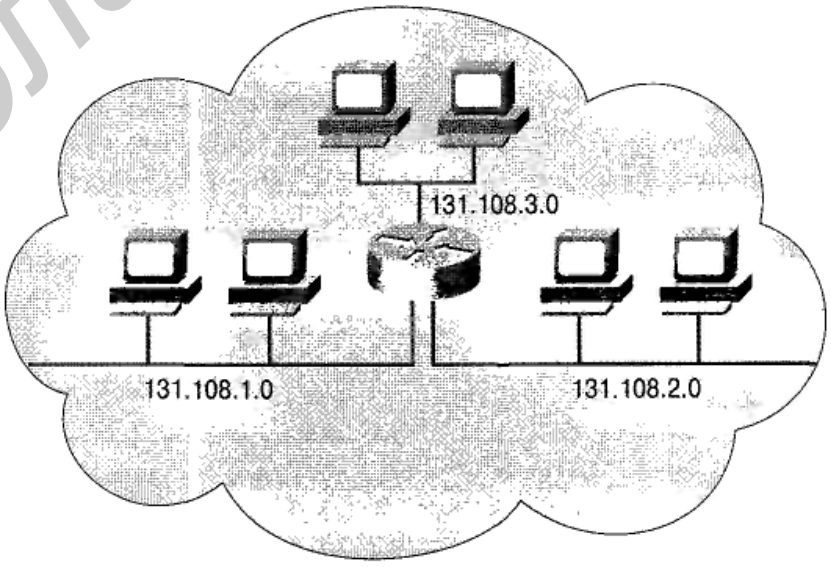


Рисунок 2.10 – Схема сети с использованием подсетей

Небольшие сети требуется разбивать на более мелкие подсети достаточно редко, но в случае использования больших блоков адресов и очень крупных сетей такое деление необходимо. Согласно определению создание в сети подсетей означает использование маски подсети для разделения ее на более мелкие, более эффективные, легче управляемые сегменты. Такая схема похожа на используемую в телефонных сетях нумерацию, где номер состоит из телефонного кода страны, кода региона или города и телефона конечного абонента. Такие компоненты телефонных систем сравнимы с соответствующими элементами в IP-сетях – адресами сетей, подсетей и отдельных узлов.

Наиболее важный вопрос, на который необходимо дать ответ, связан с определением нужного количества подсетей и допустимым количеством узлов, которые могут входить в каждую из полученных в процессе разбиения сетей. Благодаря использованию механизма подсетей, можно создать гибкую структуру сети, которая не будет ограничиваться масками или рамками стандартных сетей классов A, B и C.

Адреса подсетей состоят из сетевой части классов A, B или C, поля подсети и поля адреса узла. Указанные поля формируются из исходного адреса всей сети. Умение определить, каким образом разделить исходное поле адреса узла на поля адреса подсети и адреса узла, дает сетевым администраторам определенную свободу при выборе схемы адресации.

Чтобы создать подсеть, сетевой администратор заимствует биты из поля адресов узлов исходного адреса всей сети и назначает их в качестве адреса подсети. Минимальное число битов, которое может быть заимствовано, – два. Если использовать всего 1 бит, то после разбиения будет получен только один сетевой адрес (.0 – адрес сети) и один широковещательный (.255). Максимальное число битов, которые разрешено заимствовать, может быть любым (в рамках максимальной длины узловой части адреса), при условии, что останутся незадействованными не менее двух битов для адресов узлов.

Чтобы выделить подсеть, биты сетевого узла должны быть переназначены как сетевые биты посредством деления **октета (или октетов)** сетевого узла на части. Хотя такой механизм называют заимствованием битов, но более точным термином будет **аренда битов**, хотя последний используется очень редко. Процесс деления всегда начинается с крайнего левого бита узла, положение которого зависит от класса IP-адреса.

Помимо повышения управляемости создание подсетей позволяет ограничить широковещательные рассылки и реализовать механизм низкоуровневой безопасности в локальной сети. Безопасность при использовании подсетей в локальных сетях реализуется благодаря тому, что доступ в другие подсети организуется через маршрутизаторы. Маршрутизатор может быть настроен таким образом, чтобы разрешить или запретить доступ к подсети на основе различных критериев, реализуя таким образом политику безопасности. Использование механизма выделения подсетей может принести дополнительные доходы за счет продажи или передачи в аренду ранее не использовавшихся IP-адресов.

На рисунке 2.11 показано, как в среде с многочисленными сетями каждая из них подключена к сети Internet посредством единой точки доступа – общего маршрутизатора. Подробности и детали организации внутренней сети несущественны для сети Internet. С использованием подсетей можно организовать частную сеть, в которой внутренние устройства будут заниматься доставкой данных пользователей. Таким образом, задача устройств сети Internet состоит только в том, как доставить данные сетевому маршрутизатору-шлюзу, посредством которого частная сеть подключена к глобальной. Внутри частной сети узловая часть IP-адреса может быть разделена на части для создания подсетей.

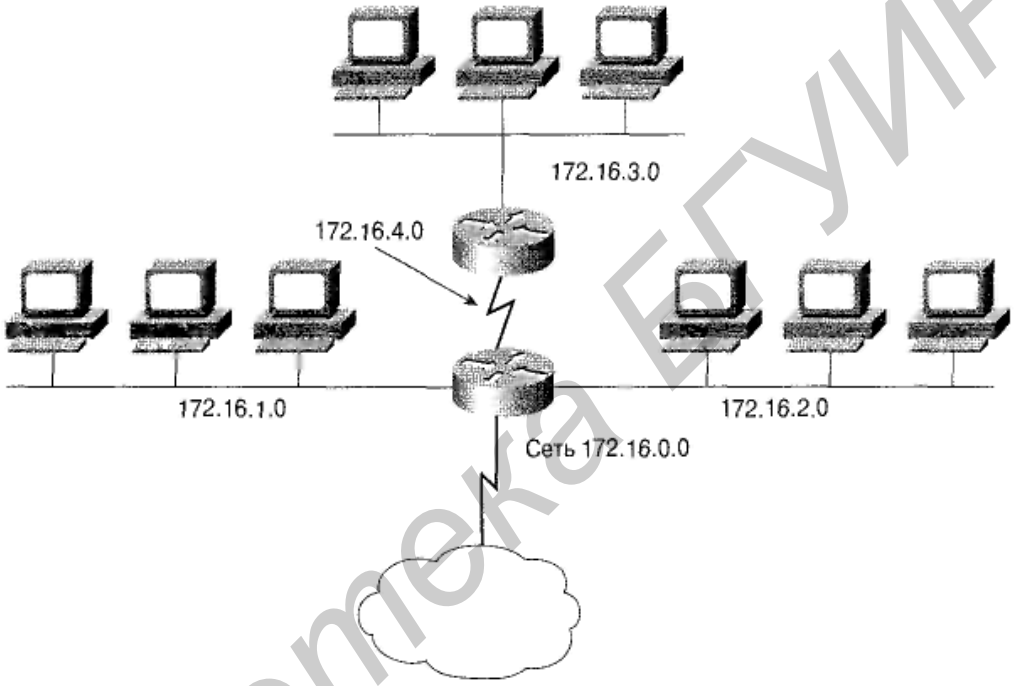


Рисунок 2.11 – Подсети

Поскольку *адрес подсети* формируется из узловой части адреса класса A, B или C, он назначается локально, обычно местным сетевым администратором. Кроме того, как и остальные части IP-адреса, каждый адрес подсети должен быть уникальным внутри области их использования (рисунок 2.12).

Использование подсетей часто бывает необходимо при объединении локальных сетей с целью создания единой распределенной сети. Например, при объединении двух локальных сетей, расположенных в географически удаленных точках, можно назначить уникальные подсети каждой из локальных сетей и каналу распределенной сети между ними. В таком случае могут быть использованы два маршрутизатора (по одному в каждой из сетей) для маршрутизации пакетов между локальными сетями (подсетями).

Другой важной причиной использования подсетей является необходимость в уменьшении размеров широковещательных доменов. Широковещательные пакеты рассылаются всем узлам в сети или подсети. Когда широковещательный трафик начинает расходовать значительную часть доступной полосы пропускания, можно уменьшить размеры широковещательного домена.

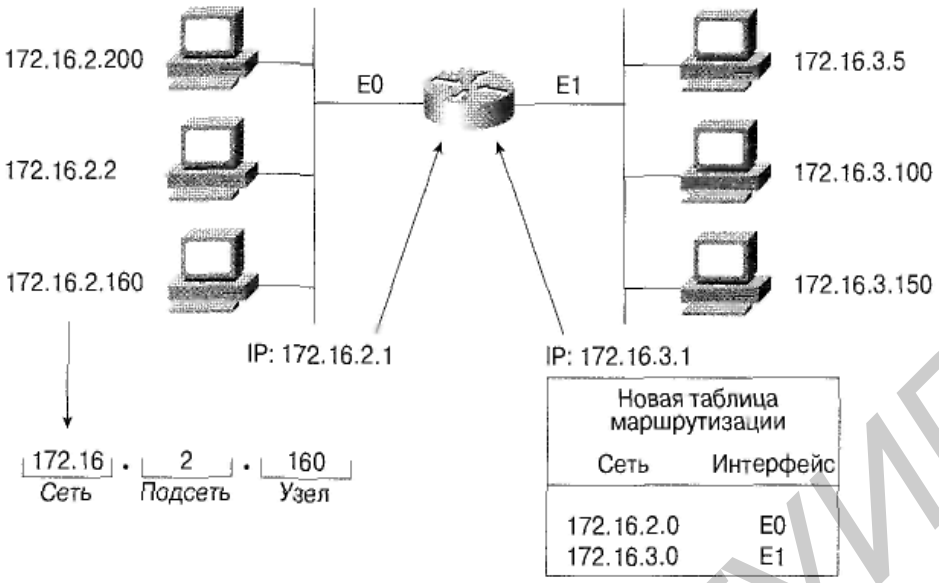


Рисунок 2.12 – Адреса подсетей

Внешний мир «видит» локальную сеть как единую сеть, ничего не зная о ее внутренней структуре. Такой подход позволяет уменьшить таблицы маршрутизации и эффективно их использовать. Получив локальный адрес узла 192.168.10.14, внешний мир за пределами локальной сети использует только объявленный основной сетевой адрес 192.168.10.0. Причина этого в том, что локальный адрес 192.168.10.14 действителен только в пределах локальной сети 192.168.10.0. В других местах он работать не будет.

Адрес подсети включает сетевую часть адреса классов A, B и C плюс поле подсети и поле узла. Эти поля создаются на основе оригинального IP-адреса заимствованием битов из узловой части адреса и присоединением к исходной сетевой части адреса. Возможность деления оригинальной узловой части адреса на новые подсети и адреса узлов предоставляет гибкость в выборе схемы адресации для сетевых администраторов. Это означает, что у сетевого администратора есть более широкий выбор при выборе схемы адресации как изначально, так и при расширении сети.

2.1.6 Назначение маски подсети

Выбор необходимого количества битов для создания подсети зависит от требуемого максимального количества узлов в подсети. Чтобы вычислить результат заимствования определенного количества узловых битов для создания подсети, необходимо иметь базовые знания из области двоичной математики и помнить битовые значения в каждой из позиций октета, как показано в таблице 2.3. Независимо от класса IP-адреса, последние 2 бита в последнем октете никогда не могут быть использованы для формирования подсети. Они называются *наименее значимыми битами.* Заимствование всех доступных битов, за исключением двух последних, позволяет создать подсеть, которая содержит только два узла. Такой способ используется на практике

для экономии адресов при адресации последовательных связей между маршрутизаторами. Однако для работающих локальных сетей это вызвало бы недопустимые расходы на оборудование.

Таблица 2.3 – Расчет подсети: два формата маски подсети

Формат с обратной косой чертой	/25	/26	/27	/28	/29	/30	—	—
Маска	128	192	224	240	248	252	254	255
Бит	1	2	3	4	5	6	7	8
Значение	128	64	32	16	8	4	2	1

Чтобы создать **маску подсети**, дающую маршрутизатору информацию, необходимую для вычисления адреса подсети, которой принадлежит конкретный узел, необходимо выбрать столбец из таблицы с нужным количеством битов и в качестве значения маски воспользоваться числом строкой выше из того же столбца, как показано в таблице 2.3. Это значение получено в результате сложения двоичных значений для знакомест используемых битов. Как видно из таблицы 2.3, если заимствованы 3 бита, маска подсети для сети класса С будет равна 255.255.255.224. При использовании формата записи маски с обратной косой чертой он может быть представлен как «/27». Число, указанное после символа обратной косой черты, представляет собой количество битов, составляющих адрес сети, плюс биты, использующиеся для маски подсети.

Чтобы определить требуемое количество битов, разработчик сети должен рассчитать, какое максимальное число узлов будет в подсети, и общее количество подсетей. В качестве примера предположим, что необходимо разместить по 30 узлов в пяти подсетях. Чтобы определить необходимое количество битов для переназначения, воспользуемся строкой «Количество используемых узлов» (таблица 2.4). Так, для использования 30-ти узлов требуются 3 бита.

Таблица 2.4 – Расчет подсети: подсети и узлы

Формат с обратной косой чертой	/25	/26	/27	/28	/29	/30	---	---
Маска	128	192	224	240	248	252	254	255
Бит	1	2	3	4	5	6	7	8
Значение	128	64	32	16	8	4	2	1
Всего подсетей		4	8	16	32	64		
Доступные подсети		2	6	14	30	62		
Всего узлов		64	32	16	8	4		
Количество используемых узлов		62	30	14	6	2		

Таким образом, будет создано шесть подсетей, что также удовлетворяет указанным выше требованиям. Следует помнить, что разница в количестве доступных узлов и полном количестве возникает из-за того, что первый доступный адрес является идентификатором сети, а последний – ее широковещательным адресом. Классовая маршрутизация не предоставляет механизм использования соответствующих подсетей, в то время как при бесклассовой

маршрутизации множество таких «потерянных» адресов доступно для использования, как показано в таблице 2.4. Глядя на таблицу, можно также оценить, какое количество подсетей и узлов будет потеряно, если бесклассовая маршрутизация не используется.

2.1.7 Создание подсети

Для создания подсети необходимо расширить часть адреса, с которой оперируют маршрутизаторы. В сети Internet устройства оперируют с сетью как с единым целым согласно классам адресов A, B или C, которые задаются 8, 16 или 24 битами в маске (т. е. номером сети). Поле подсети описывает дополнительные биты, давая возможность локальным маршрутизаторам оперировать разными подсетями внутри единой, большой сети.

В маске подсети используется тот же формат, что и в IP-адресе. Иными словами, маска подсети состоит из четырех октетов, а длина ее составляет 32 бита. Сетевая часть маски подсети, как и часть, определяющая подсеть, состоит из всех единиц, а узловая ее часть заполнена нулем. Стандартно, если ни один бит не заимствован для разбиения сети на подсети, маска для сети класса B выглядит как 255.255.0.0. Если заимствованы 8 битов, соответствующая маска будет иметь вид 255.255.255.0, как показано на рисунках 2.13 и 2.14. Поскольку в адресе класса B выделены два октета под адреса узлов, для задания маски подсети может быть заимствовано не более 14 битов. В сети класса C используются только 8 битов для поля узла. Следовательно, для задания маски подсети может быть заимствовано не более 6 битов.

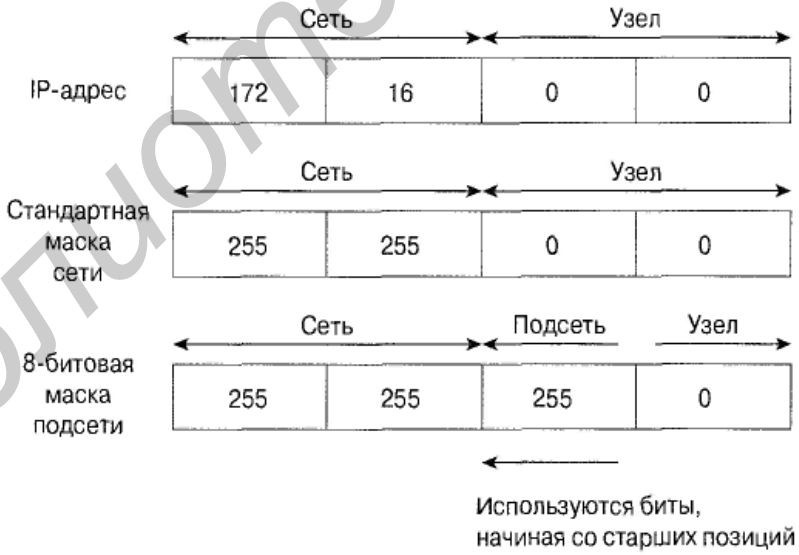


Рисунок 2.13 – Адреса сети и узла

Поле подсети всегда следует непосредственно за номером сети. Такое требование означает, что заимствовать можно первые *n* битов из стандартного поля узлов, где *n* – необходимая длина поля создаваемой подсети. Маска подсети является инструментом, который помогает маршрутизатору в определении сетевой (и используемой маршрутизатором) части адреса и его узловой части.

128	64	32	16	8	4	2	1		
1	0	0	0	0	0	0	0	=	128
1	1	0	0	0	0	0	0	=	192
1	1	1	0	0	0	0	0	=	224
1	1	1	1	0	0	0	0	=	240
1	1	1	1	1	0	0	0	=	248
1	1	1	1	1	1	0	0	=	252
1	1	1	1	1	1	1	0	=	254
1	1	1	1	1	1	1	1	=	255

Рисунок 2.14 – Схема двоичных преобразований

2.2 Задание к лабораторной работе

Разделите сеть (рисунок 2.15) на подсети так, чтобы:
– каждая подсеть в сети 172.16.0.0 /16 содержала до 1000 хостов;
– каждая подсеть в сети 172.17.0.0 /16 содержала до 80 хостов;
– в сети 172.18.0.0 /16 было минимум 19 подсетей;
– в сети 172.19.0.0 /16 было минимум 4 подсети;
Настройте PC0: IP-адрес 172.16.3.5 с вычисленной маской.
Настройте PC1: IP-адрес 172.17.0.90 с вычисленной маской.
Настройте PC2: IP-адрес 172.18.0.2 с вычисленной маской.
Настройте PC3: IP-адрес 172.19.0.9 с вычисленной маской.
Проверьте связность сети.

Указание: при выполнении работы руководствуйтесь пунктом 1.1.4 лабораторной работы №1.

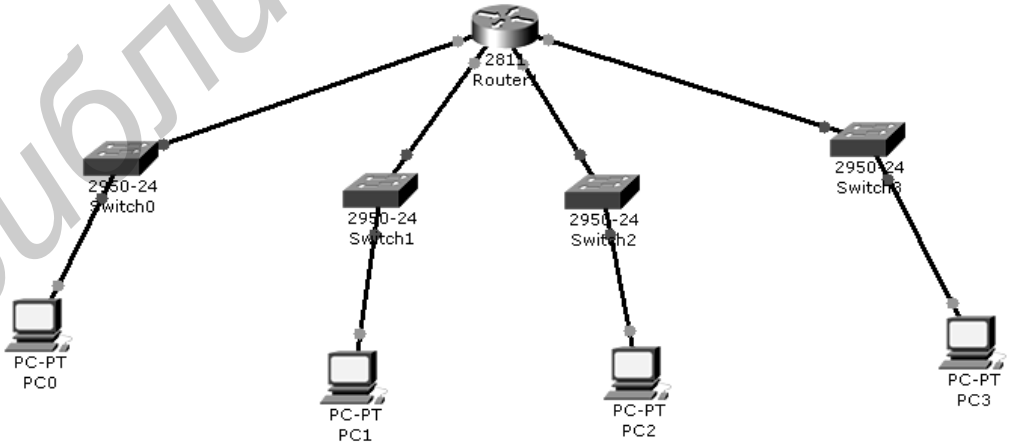


Рисунок 2.15 – Схема сети

2.3 Содержание отчета

1 Цель работы.
2 Схема сети.
3 Описание деления сети на подсети.
4 Выводы.

2.4 Контрольные вопросы

1 Что такое IP-адреса? Из каких частей они состоят?
2 Назовите классы IP-адресов.
3 Сколько может быть создано сетей класса A?
4 Приведите примеры частных IP-адресов в сетях классов A, B, C.
5 Каково назначение маски подсети?
6 Какие есть форматы записи масок подсети?

Лабораторная работа №3

ЭТАЛОННАЯ МОДЕЛЬ ВЗАИМОДЕЙСТВИЯ ОТКРЫТЫХ СИСТЕМ. ОБОРУДОВАНИЕ ПЕРВОГО И ВТОРОГО УРОВНЯ

Цель работы: изучить работу устройств первого и второго уровней.

3.1 Теоретическая часть

3.1.1 Использование уровней для описания процесса обмена данными в сети

Трудность реализации сетевого проекта состоит в том, что это достаточно сложный процесс. Он становится особенно трудным, если смотреть на него как на единое целое. Решение этой проблемы состоит в разделении всей системы сетевой коммуникации на ряд уровней. При этом каждый уровень отвечает за определенную часть процесса коммуникации и взаимодействует только с уровнем, находящимся непосредственно под ним, и с уровнем над ним. Такое взаимодействие строго определяет назначение каждого уровня. Основной сетевой моделью, использующей уровни, является эталонная модель взаимодействия открытых систем (Open System Interconnection – OSI).

3.1.2 Эталонная модель OSI

Эталонная модель OSI является первичной моделью, используемой в качестве основы для сетевых коммуникаций. Она определяет сетевые функции, выполняемые каждым ее уровнем. Кроме того, модель OSI описывает, каким образом информация или пакеты данных перемещается от программ-приложений (таких как электронные таблицы или текстовые процессоры) по сетевой передающей среде (такой как провода) к другим программам-приложениям, работающим на другом компьютере этой сети, даже если отправитель и получатель используют разные виды передающих сред.

Эталонная модель OSI содержит семь пронумерованных уровней, каждый из которых выполняет свои особые функции в сети.

Уровень 1 – физический уровень.

Уровень 2 – канальный уровень.

Уровень 3 – сетевой уровень.

Уровень 4 – транспортный уровень.

Уровень 5 – сеансовый уровень.

Уровень 6 – уровень представления данных.

Уровень 7 – уровень приложений.

Деление сетевого взаимодействия на уровни обеспечивает следующие преимущества:

– процесс сетевой коммуникации подразделяется на меньшие и более простые этапы;

– стандартизируются сетевые компоненты, что позволяет использовать и поддерживать в сети оборудование разных производителей;

– деление процесса обмена данными на уровни позволяет осуществлять связь между различными типами аппаратного и программного обеспечения;

– изменения на одном уровне не влияют на функционирование других уровней, что позволяет быстрее разрабатывать новые программные и аппаратные продукты;

– коммуникация в сети подразделяется на компоненты меньшего размера, что облегчает их изучение.

3.1.3 Функции уровней эталонной модели OSI

Уровень 1: физический уровень. Физический уровень (physical layer) определяет электрические, процедурные и функциональные спецификации для активизации, поддержки и отключения физических каналов между оконечными системами. Спецификациями физического уровня определяются уровни напряжений, синхронизация изменений напряжения, физическая скорость передачи данных, максимальная дальность передачи, физические соединения и другие аналогичные параметры.

Уровень 2: канальный уровень. Канальный уровень (data link layer) обеспечивает надежную передачу данных по физическому каналу. При этом канальный уровень решает задачи физической (в противоположность логической) адресации, анализа сетевой топологии, доступа к сети, уведомления об ошибках, упорядоченной доставки кадров (фреймов, пакетов) и управления потоками.

Уровень 3: сетевой уровень. Сетевой уровень (network layer) является комплексным уровнем, обеспечивающим выбор маршрута и соединение между собой двух рабочих станций, которые могут быть расположены в географически удаленных друг от друга сетях. Кроме того, сетевой уровень решает вопросы логической адресации. Примерами протоколов третьего уровня могут служить межсетевой протокол (Internet-protocol, IP), протокол межсетевого пакетного обмена (Internetwork Packet Exchange — IPX) и протокол AppleTalk.

Уровень 4: транспортный уровень. Транспортный уровень (transport layer) сегментирует данные передающей станции и вновь собирает их в одно целое на принимающей стороне. Границу между транспортным уровнем и уровнем сеанса связи можно рассматривать как границу между протоколами приложений и протоколами передачи данных. Транспортный уровень пытается обеспечить службу передачи данных таким образом, чтобы скрыть от верхних уровней детали процесса передачи данных. В частности, задачей транспортного уровня является обеспечение надежности передачи данных между двумя рабочими станциями. При обеспечении службы связи транспортный уровень устанавливает, поддерживает и соответствующим образом ликвидирует виртуальные каналы. Для обеспечения надежности транспортной службы используются выявление ошибок при передаче и управление информационными потоками. Примерами протоколов четвертого уровня мо-

гут служить протокол управления передачей (Transmission Control Protocol – TCP), протокол пользовательских дейтаграмм (User Datagram Protocol – UDP) и протокол последовательного обмена пакетами (Sequenced Packet Exchange – SPX).

Уровень 5: сеансовый уровень. Как показывает само название этого уровня, сеансовый уровень (session layer) устанавливает сеанс связи между двумя рабочими станциями, управляет им и разрывает его. Сеансовый уровень предоставляет свои службы уровню представления данных. Он также синхронизирует диалог между уровнями представления двух систем и управляет обменом данными. Кроме своей основной постоянной функции – управления, уровень сеанса связи обеспечивает эффективную передачу данных, требуемый класс обслуживания и рассылку экстренных сообщений о наличии проблем на сеансовом уровне, уровне представления данных или уровне приложений. Примерами протоколов пятого уровня могут служить сетевая файловая система (Network File System – NFS), система X-Windows и протокол сеанса AppleTalk (AppleTalk Session Protocol –ASP).

Уровень 6: уровень представления данных. Задача уровня представления данных (presentation layer) состоит в том, чтобы информация уровня приложений, которую посылает одна система (отправитель), могла быть прочитана уровнем приложений другой системы (получателя). При необходимости уровень представления преобразует данные в один из многочисленных существующих форматов, который поддерживается обеими системами. Другой важной задачей этого уровня является шифрование и расшифровка данных. Типовыми графическими стандартами шестого уровня являются стандарты PICT, TIFF и JPEG. Примерами стандартов шестого уровня эталонной модели, описывающих формат представления звука и видео, являются стандарты MIDI и MPEG.

Уровень 7: уровень приложений. Уровень приложений (application layer) является ближайшим к пользователю и предоставляет службы его приложениям. От других уровней он отличается тем, что не предоставляет служб другим уровням; вместо этого он предоставляет службы только приложениям, которые находятся вне рамок эталонной модели OSI. Примерами таких приложений могут служить электронные таблицы (например программа Excel) или текстовые процессоры (например программа Word). Уровень приложений определяет доступность партнеров по сеансу связи друг для друга, а также синхронизирует связь и устанавливает соглашение о процедурах восстановления данных в случае ошибок и процедурах контроля целостности данных. Примерами приложений седьмого уровня могут служить программы, работающие по протоколам Telnet и HTTP.

Чтобы передавать данные от отправителя получателю, необходимо установить соединение между однотипными уровнями сети – одноранговая связь. Во время этого процесса протоколы одного и того же уровня обеих систем обмениваются информацией, называемой протокольными единицами обмена (Protocol Data Unit – PDU).

Пакеты данных создаются станцией-отправителем, а затем передаются в пункт назначения. Функционирование каждого уровня зависит от службы, предоставляемой уровнем модели OSI, лежащим непосредственно под ним. Для предоставления такой службы нижний уровень использует инкапсуляцию, которая заключается в размещении модуля PDU находящегося над ним уровня в поле данных своего модуля PDU. После этого каждый уровень может добавить заголовки, которые требуются ему для выполнения своих функций. По мере того как данные передаются по уровням модели OSI, к ним добавляются дополнительные заголовки. Модуль данных протокола (PDU) на четвертом уровне называется сегментом (segment).

Сетевой уровень предоставляет службы транспортному уровню. Он обеспечивает передачу данных по объединенной сети путем инкапсуляции (пункт 3.1.4) данных транспортного уровня и добавления заголовка, в результате чего создается пакет (packer), являющийся модулем PDU третьего уровня. Заголовок пакета содержит информацию, требуемую для передачи пакета по сети, такую, в частности, как логические адреса отправителя и получателя. Канальный уровень предоставляет службу сетевому уровню. Он инкапсулирует информацию сетевого уровня во фрейм (frame), являющийся модулем PDU второго уровня. Заголовок фрейма содержит физический адрес, требуемый для выполнения канальным уровнем своих функций, а концевик (trailer) содержит контрольную последовательность фрейма (Frame Check Sequence – FCS), которая используется для проверки того, не был ли поврежден фрейм в процессе передачи. Получившийся модуль данных передается вниз, на физический уровень. Физический уровень предоставляет службу канальному уровню. Физический уровень кодирует фрейм канального уровня, превращая его в последовательность нулей и единиц (в биты) для передачи по сетевой среде (обычно медному проводу) на первом уровне.

3.1.4 Инкапсуляция

Процесс инкапсуляции включает несколько этапов.

Этап1. Первоначальное формирование данных. Когда пользователь отправляет сообщение по электронной почте, символы его письма преобразуются в данные, которые могут быть переданы по объединенной сети.

Этап 2. Упаковка данных для сквозной передачи по сети. На этом этапе данные упаковываются для передачи по сети. Используя сегменты, транспортная функция обеспечивает рабочим станции на обоих концах системы электронной почты возможность осуществлять надежную связь.

Этап 3. Добавление в заголовок сетевого адреса. Данные помещаются в пакеты или дейтаграммы, содержащие сетевой заголовок, в котором расположены логические адреса источника и получателя. Эти адреса используются сетевыми устройствами для пересылки пакета по сети в соответствии с выбранным маршрутом.

Этап 4. Добавление локального адреса в заголовок канального уровня. Каждое сетевое устройство должно поместить пакет сетевого уровня во фрейм канального уровня. Преобразование пакета во фрейм позволяет осуществить соединение со следующим лежащим на данном маршруте непосредственно подсоединенным сетевым устройством. Каждому устройству на выбранном в сети маршруте необходимо выполнить такое преобразование пакета во фрейм для соединения со следующим устройством.

Этап 5. Преобразование в биты для передачи по сети. Функция синхронизации позволяет устройствам различать передаваемые биты при их передаче по сети. На протяжении используемого маршрута физическая среда объединенной сети может меняться. Например, электронное сообщение может исходить из локальной сети (Local Area Network, LAN), пересечь территориальную магистраль и выйти в канал распределенной сети (Wide Area Network, WAN), перед тем как достичь пункта назначения или другой удаленной локальной сети. Заголовки и концевики добавляются по мере того, как данные перемещаются по уровням эталонной модели OSI.

3.1.5 Декапсуляция

Когда удаленное устройство получает последовательность битов, его физический уровень передает эти биты на канальный уровень для последующей обработки. Канальный уровень выполняет следующие действия:

Этап 1. Выполняется проверка, соответствует ли MAC-адрес пункта назначения адресу этой станции и не является ли он широковещательным адресом. Если ни одно из условий не выполняется, фрейм отбрасывается.

Этап 2. Если данные содержат ошибки, они могут быть отброшены; в этом случае канальный уровень может запросить повторную передачу данных.

Этап 3. Канальный уровень удаляет заголовок канального уровня и концевик, а затем передает оставшиеся данные на сетевой уровень, основываясь на управляющей информации, содержащейся в заголовке канального уровня.

3.1.6 Оборудование первого уровня

Одной из первых задач, которая стоит перед любой технологией транспортировки данных, является их передача на максимально большое расстояние. Физическая среда накладывает на этот процесс свое ограничение – рано или поздно мощность сигнала падает и прием становится невозможным. При этом не имеет значения абсолютное значение амплитуды – для распознавания важно соотношение сигнал/шум. Привычное для аналоговых систем усиление не годится для высокочастотных цифровых сигналов. Разумеется, при его использовании какой-то небольшой эффект может быть достигнут, но с увеличением расстояния искажения быстро нарушат целостность данных. В

таких ситуациях применяют не усиление, а повторение сигнала. При этом устройство на входе должно принимать сигнал, далее распознавать его первоначальный вид и генерировать на выходе его точное подобие.

Первоначально в сетях стандарта Ethernet использовался коаксиальный кабель с топологией «шина», и нужно было соединять между собой всего несколько протяженных сегментов. Для этого обычно использовались **повторители (repeater)**, имевшие два порта. Несколько позже появились многопортовые устройства, называемые концентраторами (hub).

Концентратор (hub) – многопортовый повторитель, соединяющий несколько физических сегментов. В то время как типичный повторитель имеет только 2 порта, концентратор обычно имеет от 4-х до 24-х портов. Чаще всего используются в сетях 10BASE-T и 100BASE-T, хотя могут использоваться и в других типах сетей. Использование концентратора преобразует сетевую топологию из шинной, в которой каждое устройство непосредственно подсоединено к обшей шине, в звездообразную, оставляя неизменной логическую топологию. Увеличивает надежность сети. Передаваемый сигнал поступает во все элементы сети.

Крупные сети не могут быть построены на основе одной разделяемой среды передачи, т. к. невозможно перераспределение трафика (англ. traffic – уличное движение, здесь имеется в виду движение данных) между различными частями сети и практически все современные технологии ограничивают количество узлов, которое может быть подсоединено к разделяемой среде.

Подсоединенные к концентраторам устройства остаются в одной и той же разделяемой среде, и при одновременной передаче данных в сеть несколькими узлами возникают коллизии. Кроме того, концентраторы могут работать только в полудуплексном режиме, т. е. в каждый конкретный момент они могут либо только передавать, либо только получать данные.

Концентраторы принадлежат к одному из трех указанных типов:

– активный концентратор должен быть подключен к источнику внешнего питания, поскольку ему нужна энергия для усиления входящего сигнала перед передачей его на внешние порты;

– интеллектуальный концентратор иногда называют «умным» (smart hubs). В целом, такие устройства функционируют как обычные концентраторы, однако имеют встроенный микропроцессор и обладают возможностями диагностики. Они дороже обычных концентраторов, однако полезны в аварийных ситуациях;

– пассивный концентратор выступает исключительно в качестве точки физического соединения устройств. Такой концентратор не проверяет проходящий через него трафик и не выполняет никаких действий с потоками данных; он не усиливает и не очищает сигнал. Пассивный концентратор только предоставляет доступ к общей шине и поэтому не требует наличия источника питания.

3.1.7 Оборудование второго уровня

Мосты(Bridge), как и повторители, принимают данные на входящий порт и передают на исходящий с восстановленными уровнем и формой сигнала (рисунок 3.1). Мост принимает входящий кадр в свой буфер, определяет его целостность и адрес (MAC) назначения. При этом каждая половина моста, анализируя поле адреса отправителя, ведет таблицу Ethernet-адресов узлов, находящихся на своей стороне. На другую сторону моста передаются только кадры широковещательной рассылки (Broadcast) и кадры, не имеющие получателя на своей стороне. Таким образом, коллизии не транслируются (как это происходит в повторителях).

Рисунок 3.1 – Соединение двух сегментов сети
с помощью моста в программной среде

Буферизация данных перед их отправкой (store-and-forward) приводит к возникновению большей по сравнению с концентраторами задержки, что несколько снижает скорость работы сети. С другой стороны, количество устройств, которые разделяют между собой физическую среду, снижается. В результате обычно реальная скорость передачи данных возрастает.

Мосты не могут выполнять фрагментацию и повторную сборку пакетов более высокого (сетевого) уровня. Это свойство вызывает важное, но не заметное на первый взгляд следствие. Многие модели мостов имеют ограничение по размеру передаваемого кадра, слишком большой может быть отброшен как поврежденный.

Коммутатор(switch) представляет собой сетевое устройство второго уровня, которое выполняет функции точки концентрации для соединения между собой рабочих станций, серверов, маршрутизаторов, концентраторов и других коммутаторов. Коммутаторы можно рассматривать как многопортовые мосты, которые являются стандартными для современных технологий локальных сетей Ethernet, использующих звездообразную топологию (рисунок 3.2). Коммутаторы обеспечивают выделенные виртуальные каналы типа «точка – точка» между каждыми двумя подсоединенными сетевыми устройствами, поэтому при одновременной передаче коллизий не происходит. Коммутаторы могут работать в дуплексном режиме; это означает, что они могут одновременно получать данные и отправлять их. Понимание характера работы коммутаторов является важным фактором для поддержки работы сети.

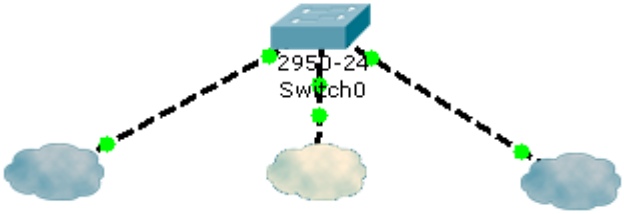


Рисунок 3.2 – Соединение сегментов сети с помощью коммутатора
в программной среде

3.2 Задание к лабораторной работе

Провести первичную настройку компьютеров сети, состоящей из 6 компьютеров PC0, PC1, PC2, PC3, PC4, PC5, соединенных коммутатором Switch0 и концентратором Hub0 через мост Bridge0. В сети есть выход в Internet через маршрутизатор Router0. Логическая схема сети представлена на рисунке 3.3. Протестировать работоспособность сети. Продемонстрировать результат преподавателю.

Указание: при выполнении работы руководствуйтесь пунктом 1.1.4 лабораторной работы №1.

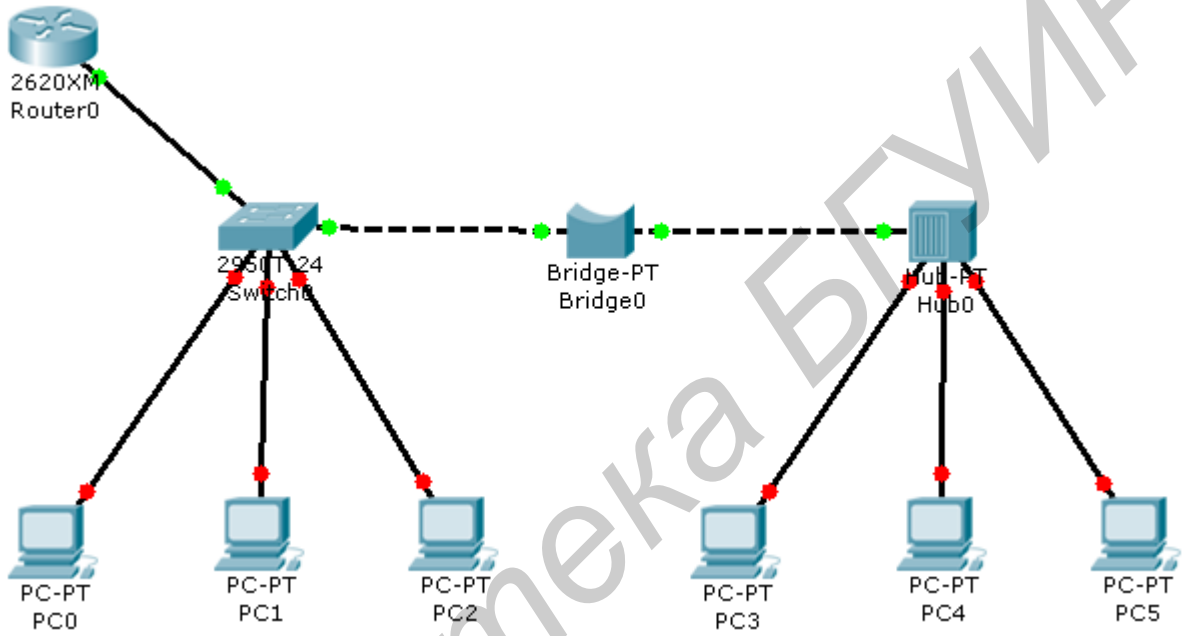


Рисунок 3.3 – Логическая схема учебной сети

3.3 Содержание отчета

1 Цель работы.
2 Схема топологии сети.
3 Таблица конфигурации компьютеров.
4 Вывод.

3.4 Контрольные вопросы

1 Сколько уровней имеет эталонная модели OSI?
2 Назовите уровни модели OSI и их функции.
3 Что такое инкапсуляция и декапсуляция?
4 Какие этапы включают процессы инкапсуляции и декапсуляции?
5 К оборудованию какого уровня относятся концентраторы?

Лабораторная работа №4

МАРШРУТИЗАТОРЫ И ИХ КОНФИГУРИРОВАНИЕ

Цель работы: ознакомиться с маршрутизаторами и получить практические навыки их конфигурирования

4.1 Теоретическая часть

4.1.1 Маршрутизаторы

Сетевой уровень (network layer) – это третий уровень эталонной модели взаимодействия открытых систем OSI. На этом уровне обеспечивается выбор маршрута и соединение между собой двух рабочих станций, которые могут быть расположены в географически удаленных друг от друга сетях. Кроме того, сетевой уровень решает вопросы логической адресации (например в сети Internet – по IP-адресу). Для реализации данной процедуры используется *маршрутизаторы (Router).*

Маршрутизатор может рассматриваться как специализированный тип компьютера. Он содержит те же компоненты, что и обычный персональный настольный компьютер. В нем есть центральный процессор (CPU), память, системная шина и интерфейсы ввода/вывода. Тем не менее любое маршрутизирующее устройство выполняет некоторые специфические функции, которые не присущи обычным настольным компьютерам. В качестве примера такой функции можно указать механизм объединения, например двух сетей, и метод определения маршрутов для потоков данных между такими непосредственно подключенными к устройству сетями. Отметим, что обычный компьютер с несколькими сетевыми интерфейсами способен работать как маршрутизатор, если установить на него специальное программное обеспечение (ПО).

Так же, как для работы обычного компьютера, для работы маршрутизатора необходима операционная система (например Cisco IOS, Internetwork Operating System – межсетевая операционная система корпорации Cisco). Операционная система в маршрутизаторе используется для интерпретации конфигурационных файлов, в которых содержатся параметры и инструкции управления потоками исходящих и входящих данных. Используя специальные протоколы и таблицы маршрутизации, маршрутизаторы позволяют определить маршрут для передачи данных, по которому информация будет доставлена быстрее всего. Путь выбирается на основании действующих конфигурационных файлов и информации от протоколов маршрутизации. Файлы конфигурации содержат всю необходимую для работы устройства информацию. В маршрутизирующих устройствах корпорации Cisco, которые работают под управлением операционной системы Cisco IOS, существуют два конфигурационных файла: стартовый и рабочий, или текущий (startup-config и running-config).

В маршрутизаторах могут быть интерфейсы как для локальных, так и для распределенных сетей. Главным назначением маршрутизаторов является объединение сетей в единую распределенную сеть посредством соединений, например Frame Relay или выделенных линий, например T1. Их также можно использовать и просто для сегментации локальных сетей. Фактически для подключения других маршрутизаторов чаще всего используются интерфейсы распределенных сетей. Маршрутизаторы обмениваются информацией посредством распределенной сети. Эти устройства являются базовыми устройствами больших корпоративных сетей и сетей, которые входят в состав структуры Internet. Фактически маршрутизаторы редко выступают в качестве основных устройств крупных распределенных сетей (а именно – телекоммуникационных и телефонных), чаще они используются для подключения к сети Internet в небольших инфраструктурах. В сетях среднего и крупного размера маршрутизаторы зачастую используются для разделения широковещательных доменов, маршрутизации и т. п.

Маршрутизаторы обеспечивают две основные функции:

1) выбор оптимального маршрута для входящих пакетов данных;

2) передача пакетов соответствующим интерфейсам.

К основным компонентам маршрутизатора относятся: оперативная память (Random-Access Memory-RAM), энергонезависимая память (Non-Volatile Random-Access Memory-NVRAM), Flash-память, постоянное запоминающее устройство (Read-Only Memory-ROM) и интерфейсы.

Оперативная память (RAM/DRAM):

– используется для хранения таблиц маршрутизации;

– хранит кэш протокола ARP;

– содержит быстродействующий кэш;

– отвечает за буферизацию пакетов (разделяемая оперативная память);

– обеспечивает хранение пакетов;

– обеспечивает временную и рабочую память для файлов конфигурации маршрутизатора при включенном питании;

– содержимое RAM-памяти теряется после выключения питания или перезагрузки устройства.

Энергонезависимая память (NVRAM):

– содержит резервную, или стартовую, копию файла конфигурации;

– при перезагрузке или после выключения данные в этой памяти не стираются.

Flash-память:

– стираемая, перепрограммируемая память, которая обычно работает только в режиме чтения (EPROM);

– содержит образ операционной системы и микрокод;

– позволяет обновлять программное обеспечение без извлечения и перемещения чипа на процессоре;

– содержит данные, которые при перезагрузке или завершении работы маршрутизатора не уничтожаются;

– несколько версий операционной системы могут быть сохранены во Flash-памяти.

Постоянное запоминающее устройство (ROM):

– содержит код команд самотестирования при включении **питания (Power-On Self Test – POST)**;

– содержит программы начальной загрузки и основное программное обеспечение операционной системы;

– для обновления программного обеспечения в ПЗУ требуется замена подключаемого чипа на системной плате устройства.

Интерфейс:

– сетевое соединение, через которое пакеты данных передаются из маршрутизатора и поступают в устройство.

– размещается на системной плате или в отдельном модуле интерфейса.

4.1.2 Внешние порты маршрутизатора

В маршрутизаторах существует три типа разъемов: интерфейсы локальных сетей, интерфейсы распределенных сетей и порты управления.

Интерфейсы локальных сетей позволяют маршрутизатору соединяться со средой локальной сети (например одной из форм среды Ethernet).

Интерфейсы распределенных сетей (WAN) обеспечивают подключение к удаленным узлам или к сети Internet через сеть провайдера. В качестве портов распределенной сети могут выступать как простые последовательные соединения, так и любой другой тип интерфейса из множества существующих технологий распределенных сетей. При использовании некоторых типов интерфейсов распределенных сетей для подключения маршрутизатора к местной службе провайдера услуг необходимы дополнительные внешние устройства, такие как CSU/DSU. Некоторые разновидности соединений распределенных сетей позволяют подключить маршрутизатор непосредственно к оборудованию поставщика услуг (рисунок 4.1).

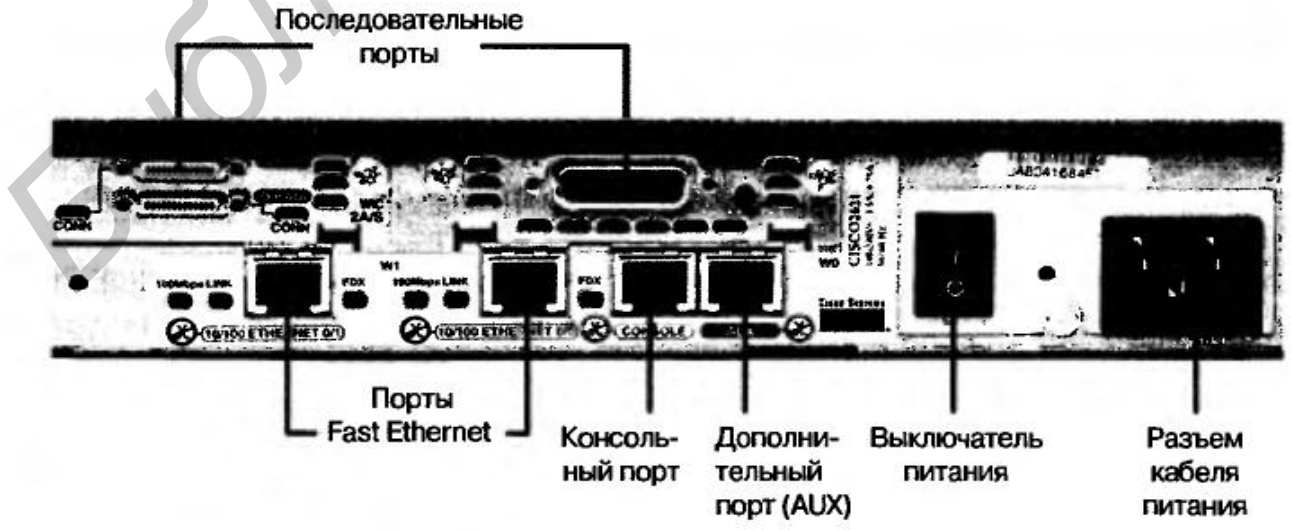


Рисунок 4.1 – Внешние порты маршрутизатора

Функции портов управления отличаются от функций других интерфейсов. Интерфейсы локальных и распределенных сетей обеспечивают сетевое взаимодействие, посредством которого передаются пакеты данных. Порт управления обеспечивает соединение, передающее текстовую информацию, которая используется для конфигурирования и исправления ошибок в работе устройства. Наиболее часто используемые интерфейсы управления – это консоль и вспомогательные порты (AUX). Они представляют собой последовательные асинхронные порты RS-232, которые подсоединяются к коммуникационным портам персонального компьютера, выступающего в роли терминала. Для начальной конфигурации маршрутизатора требуется использовать один из этих двух портов. На компьютере должна быть запущена программа эмуляции терминала, которая должна обеспечивать сеанс обмена текстовой информацией с маршрутизатором. Такой сеанс дает возможность управлять маршрутизатором.

4.1.3 Осуществление консольного соединения

При первом включении маршрутизатора в нем не настроены никакие сетевые параметры. Таким образом, устройство не способно взаимодействовать ни с каким сетями. Для подготовки маршрутизатора к запуску и настройке следует подсоединить к какому-либо порту управления маршрутизатора (интерфейс RS-232) терминал или компьютер, эмулирующий терминал. Таким образом администратор сможет передавать команды конфигурации маршрутизатору. После того как маршрутизатор будет настроен, его можно подключать к сети для дальнейшего конфигурирования, устранения неполадок, мониторинга и эксплуатации.

Кроме того, маршрутизатор может быть настроен таким образом, что появится возможность удаленно управлять им, если подсоединить модем к порту консоли или AUX-порту.

Для поиска и устранения неисправностей предпочтительнее использовать порт консоли, чем AUX-порт, поскольку порт консоли стандартно настроен на отображение параметров запуска маршрутизатора, его отладки и сообщений об ошибках. Если сетевые службы по какой-либо причине не запустились или в их работе произошла ошибка, то для устранения проблем можно использовать порт консоли. Чтобы подключиться к консольному порту, необходимо использовать консольный кабель и адаптер для разъема RJ-45 на разъем DB-95 для подключения к персональному компьютеру. В стандартной поставке любое устройство комплектуется необходимыми средствами для подключения к персональному компьютеру: кабелем и адаптером-переходником.

Программное обеспечение персонального компьютера или алфавитно-цифровой терминал должны поддерживать режим работы или эмуляцию режима vt100 (vt100 – это ранее широко используемый тип видеотерминала и соответствующий протокол связи с ним). Наиболее часто используется программный пакет эмуляции терминала HyperTerminal (рисунок 4.2), который входит в стандартную поставку операционной системы Windows.

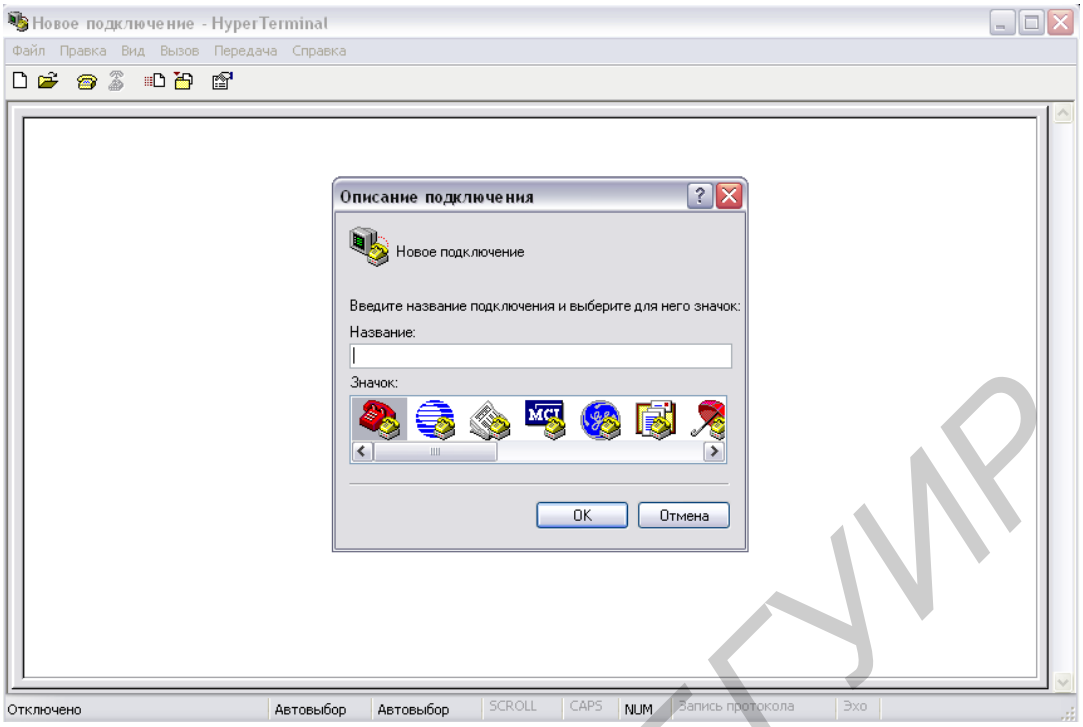


Рисунок 4.2 – Окно программного пакета HyperTerminal

Для подключения персонального компьютера к консольному порту следует выполнить следующее.

Этап 1. Осуществить для программы эмуляции терминала персонального компьютера следующие настройки:

– указать соответствующий COM-порт;

– установить скорость 9600 бод;

– установить режим 8 битов данных в посылке;

– указать отсутствие проверки четности (no parity);

– установить использование одного стопового бита;

– указать отсутствие механизма управления потоком.

Этап 2. Подключить разъем RJ-45 консольного кабеля к консольному порту.

Этап 3. Подключить второй разъем RJ-45 консольного кабеля к переходнику DB-9.

Этап 4. Подключить переходник DB-9 к COM-порту персонального или портативного компьютера.

4.1.4 Подключение к маршрутизатору через интерфейсы локальных сетей

К большинству сред локальных сетей маршрутизаторы подключаются посредством соединений Ethernet или Fast Ethernet. В данном случае маршрутизатор является узлом, который подключен к сети LAN посредством коммутатора или концентратора; для такого подключения используется кабель с прямой распайкой контактов. Интерфейс 10/100BASE-TX маршрутизатора должен быть подключен как минимум неэкранированной витой парой кате-

гории 5 (UTP) к любому другому устройству независимо от типа маршрутизатора, как показано на рисунке 4.3.

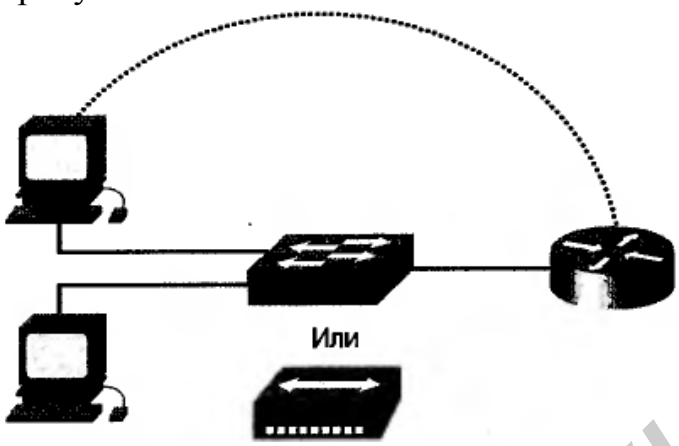


Рисунок 4.3 – Подключение маршрутизатора посредством неэкранированной витой пары

В некоторых случаях Ethernet-интерфейс устройства необходимо будет подключить напрямую к такому же интерфейсу маршрутизатора или непосредственно к сетевой плате персонального компьютера: в этом случае следует использовать перекрещенный кабель (crossover).

В любом соединении необходимо обращать внимание на тип интерфейса. Если для подключения будет использован «неправильный» интерфейс, то может пострадать как сам маршрутизатор, так и любое другое сетевое оборудование. Во многих портах или различных типах соединений используются одинаковые разъемы, например для Ethernet-, ISDN-BRI-, консольных, AUX-портов интегрированных CSU/DSU и Token Ring-портов используется один и тот же восьмиконтактный разъем RJ-45, RJ-48 или RJ-49. Чтобы администратор мог быстро и легко отличить один порт от другого, в устройствах принято использовать специальные разноцветные метки для разных типов интерфейсов.

4.1.5 Конфигурирование маршрутизатора

Чтобы получить доступ к маршрутизатору, необходимо иметь соответствующую учетную запись. После того как администратор получил доступ к интерфейсу устройства, он может войти в один из возможных режимов конфигурирования устройства. Средства интерфейса командной строки должны интерпретировать команды, вводимые в каждом из режимов, и выполнять соответствующие им операции.

Интерфейс командной строки имеет иерархическую структуру. Для выполнения различных задач эта структура требует перехода в различные режимы. Например, для настройки интерфейсов машрутизатора необходимо войти в режим конфигурирования интерфейсов. В режиме настройки интерфейса администратор может менять только настройки интерфейсов. В различных режимах маршрутизатора командная строка имеет различные метки

приглашения командной строки, что позволяет не путать режимы и использовать только команды, присущие текущему режиму.

Операционная система IOS обеспечивает работу интерпретатора команд (EXEC). Интерпретатор проверяет и выполняет все команды, введенные с консоли.

В целях безопасности в операционной системе IOS EXEC-сеансы разделены на два уровня доступа: пользовательский EXEC-режим и привилегированный EXEC-режим, которые еще называют режимами допуска.

В пользовательском режиме доступен ограниченный набор основных команд, которые позволяют отследить режимы работы маршрутизатора. Часто этот режим упоминается как режим просмотра. Пользовательский режим не допускает изменения файла конфигурации маршрутизатора. Этот режим предназначен для специалистов по информационным технологиям, технических специалистов и сотрудников, которым нужен доступ к устройству только для проведения мониторинга, но не нужны права на изменение конфигурации коммутатора, маршрутизатора и т. п. В командной строке этот режим идентифицируется символом «>».

Привилегированный режим доступа дает возможность использовать все команды маршрутизатора. Доступ к этому режиму только авторизированный, он может (и должен!) быть ограничен паролем и логином. Для выполнения команд настройки и управления маршрутизатором системному администратору необходимо войти в привилегированный режим. Доступ к режиму глобальной конфигурации и другим специальным режимам может быть получен только из привилегированного режима. В командной строке этот режим идентифицируется символом «#».

Для перехода в привилегированный режим необходимо ввести команду enable (или ее сокращенный вариант «en») в командной строке с приглашением, которое оканчивается символом «>». Если пароль был установлен, то для продолжения работы маршрутизатор его потребует. В целях безопасности сетевые устройства **не отображают вводимые символы пароля**. Если введенный пароль верен, то приглашение командной строки маршрутизатора меняется на «#», и маршрутизатор входит в привилегированный EXEC-режим.

Пример:

```
Router>enable
Password: (вводим, символы не отображаются)
Router#
```

или

```
Router>en
Password: (вводим, символы не отображаются)
Router#
```

Введя символ «?» в привилегированном режиме, вы увидите объемный список доступных команд; к некоторым из них можно получить доступ также из пользовательского режима.

Примеры команд непривилегированного режима:

enable – переключение режима;
ping –послать echo-сообщение;
show – показать текущую информацию о системе;
telnet – открытие telnet сессии;
traceroute – трассировка пути.
Примеры некоторых команд привилегированного режима:
clock – задать системное время;
configure – режим конфигурирования;
copy – копирование из файла в файл;
debug – функция отладки (debugging);
delete – удаление файла;
dir – просмотр файла в файловой системе;
disable – выход из привилегированного режима;
erase – очистка файловой системы;
reload – остановка и выполнение холодной перезагрузки;
telnet – открытие telnet сессии;
traceroute – трассировка пути;
undebug – отключении функции отладки;
vlan – настройка параметров Vlan;
write – запись текущей конфигурации в память или на tftp сервер.

4.1.6 Режимы конфигурации маршрутизатора

Чтобы изменить настройки, которые оказывают влияние на всю систему, нужно воспользоваться командами глобальной конфигурации. Для входа в режим глобальной конфигурации используется команда привилегированного режима configure. После ввода этой команды будет запрошен источник команд конфигурации: можно будет выбрать терминал, энергонезависимое ОЗУ или сеть. По умолчанию все команды конфигурации принимаются из консоли терминала, т. е. для выбора этого режима конфигурирования достаточно нажать клавишу <Enter>.

Ниже показаны некоторые команды маршрутизатора, доступ к которым можно получить из режима глобальной конфигурации:

access-list – добавить поле списка доступа;
banner – определить сообщение-приветствие при входе в систему;
boot – изменить параметры загрузки системы;
cdp – войти в режим подкоманд конфигурации CDP;
clock – конфигурация системных часов;
config-register – определить регистр конфигурации;
enable – изменить параметры пароля привилегированного режима;
end – выход из режима конфигурирования;
exit – выход из режима конфигурирования;
hostname – установка сетевого имени системы;
interface – выбор интерфейса для конфигурирования;

ip – вход в подрежим конфигурирования IP;
line – конфигурирование линии терминала;
no – отмена команды или переход к установкам по умолчанию;
router – разрешить процесс маршрутизации;
service – изменить праметры сетевой службы;
username – установить имя пользователя (логин).
Режимы конфигурирования, доступные в глобальном режиме:

Интерфейс	Router(config-if)#
Подынтерфейс	Router(config-subif)#
Контроллер	Router(conf ig-controller)#
список преобразования	Router(config-map-list)#
класс преобразования	Router(config-raap-class)#
линия	Router(config-line)#
маршрутизатор	Router(config-router)#
IPX- маршрутизатор	Router(config-ipx-router)#
преобразование маршрутизации	Router(config-route-map)#

Для возврата маршрутизатора в режим глобальной конфигурации из подрежима используется команда exit. Нажав сочетание клавиш <Ctrl>+<Z>, вы окончательно выйдете из режима конфигурации и попадете в привилегированный EXEC-режим.

В следующем примере проиллюстрировано переключение между различными режимами маршрутизатора.

```
Router# configure terminal
Router(config)#
! далее можно ввести нужные команды
Router(config)# exit
Router#

Router#configure terminal
Router(config)# router protocol
Router(config-router)#
! далее можно ввести нужные команды
Router(config-router)# exit
Router(config)#interface type port
Router(config-if)#
! далее можно ввести нужные команды
Router(config-if)# exit
Router(config)# exit
Router#
```

4.1.7 Настройка имени маршрутизатора

Одной из основных задач, которые необходимо решить при установке маршрутизатора, является задание его имени. Именование маршрутизатора позволяет повысить удобство администрирования сети. Имя маршрутизатора задается в режиме настройки глобальной конфигурации (global configuration mode), оно называется именем узла (hostname) и отображается в системном приглашении командной строки. Если пользователем не задано имя маршрутизатора, то по умолчанию используется имя Router.

Пример:

```
Router(config)#hostname  R1
R1(config)#
```

4.1.8 Команды группы «show»

В маршрутизаторе существует большое количество разновидностей команды show, которые позволяют получить информацию о режимах работы маршрутизатора, записанную в конфигурационных файлах; такие команды очень полезны при решении проблем в работе маршрутизатора. В каждом из режимов конфигурирования команда «show ?» отображает допустимые параметры соответствующей команды. Ниже в таблице 4.1 приведен список некоторых команд «show»:

Таблица 4.1 – Команда **show** и ее параметры

Команда	Описание
show interfaces	Отображает статистику обо всех интерфейсах маршрутизатора. Если пользователю необходимо проанализировать статистические данные конкретного интерфейса, он может указать в команде show interfaces номер соответствующего интерфейса. Например: **Router# show interfaces FastEthernet0/1**
show controllers serial	Отображает информацию об аппаратных средствах
show clock	Отображает время, которое установлено в маршрутизаторе
show hosts	Отображает список котированных имен узлов и адресов
show users	Отображает список пользователей, подключенных к маршрутизатору
show history	Отображает журнал введенных команд
show flash	Отображает информацию о Flash-памяти и о файлах операционной системы Cisco IOS, хранимых в ней
show version	Отображает информацию об образе операционной системы
show arp	Отображает ARP-таблицу маршрутизатора
show protocol	Отображает глобальное состояние и состояние интерфейсов любого настроенного протокола третьего уровня
show startup-configuration	Отображает конфигурацию, сохраненную в энергонезависимом ОЗУ (NVRAM)
show running-configuration	Отображает конфигурацию, которая в настоящее время находится в ОЗУ (RAM)

В нижеприведенных примерах проиллюстрировано использование команд «show protocol», «show version» и «show interfaces».

Результат выполнения команды «show protocol»:

```
Router#  show  protocols
Global values:
```

Internet Protocol routing is enabled
DECnet routing is enabled
XNS routing is enabled
Vines routing is enabled
AppleTalk routing is enabled
Novell routing is enabled
--More--
Ethemet0 is up, line protocol is up
Internet address is 183.8.126.2, subnet mask is 255.255.255.128
DECnet cost is 5
XNS address is 3010.aaOO.0400.0284
CLNS enabled
Vines metric is 32
AppleTalk address is 3012.93, zone ld-eO
Novell address is 3010.aa00.0400.0284
--More--

Использование команды «show version»

Router# show version

Cisco Internetwork Operating System Software
IOS (tm) 4500 Software (C4500-J-M). Version 12.1.5
Copyright (c) 1986-1996 by Cisco Systems, Inc.
Compiled Fri 28-Jun-96 16:32 by rbeach
Image text-base: 0x600088A0, data-base: 0x6076E000
RCM: System Bootstrap. Version 5.1(1) RELEASE SOFTWARE (fcl)
ROM: 4500-XBOOT Bootstrap Software, Version 10.1(1) RELEASE SOFTWARE (fcl)
router uptime is 1 week, 3 days, 32 minutes
System restarted by reload
System image file is c4500-j-mz, booted via tftp from 171.69.1.129
--More--

Использование команды «show interfaces»

Router# show interfaces

Serial0 is up, line protocol is up
Hardware is MK5025
Internet address is 183.8.64.129, subnet mask is 255.255.255.128
MTU 1500 bytes, BW 56 kbit, DLY 20000 usec, rely 255/255. load 9/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
Last input 0:00:00, output 0:00:01, output hang never
Last clearing of show interfaces counters never
Output queue 0/40, 0 drops, input queue 0/75, 0 drops
Five minute input rate 1000 bits/sec, 0 packets/sec
331885 packets input, 62400237 bytes, no buffer
Received 230457 broadcasts, 0 runts, 0 giants

3 input errors, 3 CRC, 0 frame, 0 overrun, Oignored, 0 abort
403591 packets output, 66717279 bytes, 0 underruns
0 output errors, 0 collisions, 8 interface resets, 0 restarts
45 carrier transitions

Другие примеры вывода команды «show»:

Router#show flash:

System flash directory:
File Length Name/status
1 5571584 c2600-i-mz.122-28.bin
[5571584 bytes used, 26942464 available, 32514048 total]
32768K bytes of processor board System flash (Read/Write)

Router#show version

Cisco Internetwork Operating System Software
IOS (tm) C2600 Software (C2600-I-M), Version 12.2(28), RELEASE SOFTWARE (fc5)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2005 by cisco Systems, Inc.
Compiled Wed 27-Apr-04 19:01 by miwang
Image text-base: 0x8000808C, data-base: 0x80A1FECC
ROM: System Bootstrap, Version 12.1(3r)T2, RELEASE SOFTWARE (fc1)
Copyright (c) 2000 by cisco Systems, Inc.
ROM: C2600 Software (C2600-I-M), Version 12.2(28), RELEASE SOFTWARE (fc5)
System returned to ROM by reload
System image file is "flash:c2600-i-mz.122-28.bin"
cisco 2620 (MPC860) processor (revision 0x200) with 60416K/5120K bytes of memory.
Processor board ID JAD05190MTZ (4292891495)
M860 processor: part number 0, mask 49
Bridging software.
X.25 software, Version 3.0.0.
1 FastEthernet/IEEE 802.3 interface(s)
4 Low-speed serial(sync/async) network interface(s)
32K bytes of non-volatile configuration memory.
16384K bytes of processor board System flash (Read/Write)
Configuration register is 0x2102

4.1.9 Конфигурирование интерфейсов маршрутизаторов

4.1.9.1 Настройка последовательного интерфейса

Последовательный интерфейс маршрутизатора может быть настроен посредством консоли или через виртуальный терминал. Для управления син-

хронизацией соединения последовательному интерфейсу требуется синхронизирующий сигнал. В большей части оборудования подача синхронизирующих сигналов обеспечивается самой аппаратурой передачи данных (DCE), такой как устройство обслуживания канала (CSU) и пользовательское устройство, взаимодействующее с цифровым устройством (DSU). Стандартно такими устройствами являются маршрутизаторы Cisco и терминальное оборудование (DTE), однако они могут быть настроены как DCE-устройства.

В последовательном соединении двух устройств одно из них должно быть объявлено DCE-устройством (т. е. передающим) и должно обеспечивать передачу синхронизирующих сигналов. Включение таймера синхронизации и его скорость задаются командой clockrate. Существуют такие возможные скорости передачи в битах в секунду: 1200, 2400, 9600, 19 200, 38 400, 56 000, 64 000, 72 000, 125 000, 148 000, 500 000, 800 000, 1 000 000, 1 300 000, 2 000 000 и 4 000 000. Однако некоторые устройства в зависимости от их структуры могут поддерживать не все скорости передачи данных.

Для настройки последовательного интерфейса необходимо, во-первых, убедиться в наличии такого интерфейса в маршрутизаторе (при необходимости установить соответствующий модуль в свободный слот), во-вторых, выполнить действия, указанные ниже в примере.

Этап 1. Войти в режим глобальной конфигурации.

Этап 2. Войти в режим настройки требуемого интерфейса.

Этап 3. Сконфигурировать IP-адрес для интерфейса и маску подсети.

Этап 4. Указать полосу пропускания канала (необязательный этап).

Этап 5. Установить частоту синхронизирующих импульсов передающего (DCE) устройства (для принимающего устройства DTE этот этап следует пропустить).

Этап 6. Включить интерфейс.

Пример. Настройка последовательного интерфейса

```
Router# configure  terminal
Router(config)ft  interface serial 1/0
Router(config-ifJ#  ip  address  10.0.0.1  255.255.255.0
Router(config-if)# bandwidth 56
Router(config-if)#  clockrate  56000
Router(config-if)#  no shutdown
```

Стандартно все интерфейсы отключены. Для включения интерфейса необходимо ввести команду no shutdown. Иногда интерфейсы необходимо отключить для проведения технического обслуживания аппаратных средств, изменения конфигурации интерфейса, устранения проблем в работе или проведения других регламентных действий. В этом случае для отключения интерфейса может использоваться команда shutdown.

Следующая команда отключает интерфейс:

Router(config-if)# shutdown

Указанная ниже команда включает отключенный интерфейс:

Router(config-if)# no shutdown

Для выхода из текущего режима настройки интерфейса используется команда

Router(config-if)# exit

4.1.9.2 Настройка Ethernet-интерфейса

Ethernet-интерфейс маршрутизатора может быть настроен посредством консоли или через виртуальную терминальную линию. Каждый Ethernet-интерфейс должен иметь собственный IP-адрес и маску подсети.

Для настройки интерфейса Ethernet необходимо выполнить действия, указанные ниже в примере.

Этап 1. Войти в режим глобальной конфигурации.

Этап 2. Войти в режим настройки требуемого интерфейса.

Этап 3. Сконфигурировать IP-адрес для интерфейса и маску подсети.

Этап 4. Включить интерфейс.

Пример. Настройка Ethernet-интерфейса

```
Router# configure terminal
Router(config)# interface FastEthernet0/1
Router(config-if)#  ip address 192.168.1.150 255.255.255.128
Router(config-if)# no shutdown
```

Стандартно все интерфейсы отключены. Для их включения используется команда **no shutdown**. Иногда интерфейсы необходимо отключить для проведения технического обслуживания аппаратных средств, изменения конфигурации интерфейса, устранения проблем в работе или проведения других действий. В этом случае для отключения интерфейса может использоваться команда **shutdown**.

4.1.10 Тестирование соединений

В этом разделе рассмотрены команды, которые могут использоваться для проверки соединения между сетевыми устройствами:

– ping;
– traceroute;
– show ip route;
– show interfaces;
– show interfaces/clear counter;
– debug.

4.1.10.1 Команда «ping»

Протокол обмена управляющими сообщениями ICMP (Internet Control Message Protocol) позволяет маршрутизатору сообщить конечному узлу об ошибках, с которыми машрутизатор столкнулся при передаче какого-либо сооб-

щения. Эхо-протокол в рамках ICMP позволяет провести простейшую проверку сетевого соединения и проверить корректность маршрутизации сетевых пакетов.

Команда «ping» отправляет пакеты получателю и затем ждет ответных пакетов от этого узла. Результаты работы такого эхо-протокола могут помочь оценить надежность соединения, задержки передачи пакетов, а также работоспособность узла. Команда «ping» является основным механизмом тестирования соединения и может быть вызвана из пользовательского или привилегированного EXEC-режима.

Для проверки соединения при помощи команды «ping» следует выполнить действия, описанные ниже.

Этап 1. Ввести команду «ping [IP-address]» или «[name]» получателя.

Этап 2. Нажать клавишу Enter.

В таблице 4.2 приведены возможные значения, возвращаемые командой «ping».

Таблица 4.2 – Таблица соответствий для команды «ping»

Код	Значение	Возможная причина
!	Каждый восклицательный знак означает получение **ICMP** эхо-ответа	Пакет команды **ping** переслан успешно
.	Каждая точка означает, что истекло время ожидания ответа сетевым сервером	Может служить признаком одной из проблем: 1) команда ping блокируется списком управления доступом в маршрутизаторе; 2) маршрутизатор не нашел маршрута для доставки ICMP-сообщения; 3) в линии имеются физические неполадки соединения
U	Получено нераспознанное **ICMP**-сообщение	Маршрутизатор не может найти маршрута к адресу получателя
C	Отправитель сбрасывает полученные **ICMP**-пакеты и указывает на необходимость подавления отправителя трафика	Устройство на маршруте передачи, возможно, получатель, получило слишком много пакетов данных; проверьте статистику очередей пакетов
&	Истекло время существования **ICMP**-пакета	Возможно, пакет зациклился

4.1.10.2 Команда «traceroute»

Команда «traceroute» (также используется ее сокращенный вариант tracert) является инструментом, который позволяет отследить отправителя и маршрут прохождения потока данных по сети. Команда «traceroute» похожа на команду «ping», однако позволяет отследить не только состояние конечных точек маршрута, но и состояние каждого транзитного перехода пакетов в

сети. Эта команда может быть выполнена как из пользовательского, так и из привилегированного EXEC-режима.

Команда «traceroute» используется следующим образом.

Этап 1. Введите команду «traceroute [IP-address]» или «[name]» (имя) получателя.

Этап 2. Нажмите клавишу Enter.

В таблице 4.3 представлены расшифровки кодов, возвращаемых командой **traceroute**, а на рисунке 4.4 – схема проверки соединений с помощью этой команды.

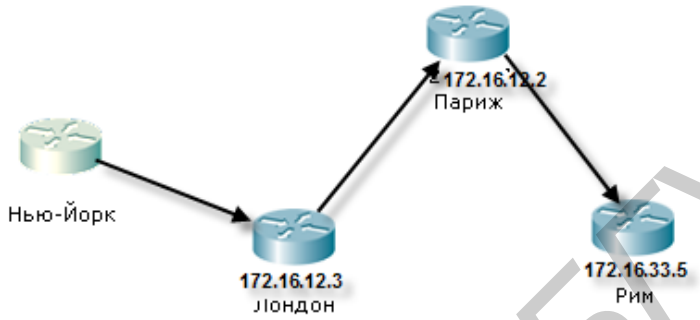


Рисунок 4.4 – Проверка соединения с помощью команды «traceroute»

Таблица 4.3 – Коды, возвращаемые командой «traceroute»

Код	Значение	Возможная причина
nn msec	Время передачи пакета (в миллисекундах) между узлами	Трассировка прошла успешно
*	Истекло время ожидания запроса	Тестируемое устройство не получило запрос или не ответило на ICMP-сообщение «packet life exceeded» («превышено время жизни пакета»)
A	Пересылка пакетов административно запрещена	Устройство на маршруте, например такое как маршрутизатор или брандмауэр (firewall), блокирует пакеты команды traceroute
Q	Отправитель сбрасывает полученные ICMP-пакеты и требует подавление источника пакетов	Устройство на маршруте, возможно, получило слишком много пакетов, проверьте статистику очередей
H	Получено нераспознанное ICMP-сообщение	Возможно, произошло зацикливание

Пример использования команды «traceroute»:

York# trace Rome
Type escape to abort.
Tracing the route to Rome (172.16.33.5)
1 LONDON (172.16.12.3) 1000 msec 8 msec 4 msec
2 PARIS (172.16.16.2) 8 msec 8 msec 8 msec
3 ROME (172.16.35.5) 8 msec 8 msec 4 msec

York#

Команда «traceroute» использует сообщения об ошибках, генерируемые маршрутизаторами, когда истекает время жизни пакета (TTL) или превышается значение максимального числа переходов. Команда **traceroute** отправляет несколько ping-пакетов с увеличивающимся значением TTL и отображает время их прохождения. Поскольку каждый последовательно отправляемый пакет имеет меньшее время жизни, то каждый последующий уничтожается на более близком участке сети. Одним из применений команды traceroute является поиск неисправного участка сети.

4.1.10.3 Команда show ip rout»

В маршрутизаторе имеются мощные инструменты для анализа работы сети. Администратор может просмотреть таблицы маршрутизации, в которых содержится информация о путях передачи данных по сети, а также выполнить другие тесты сетевого уровня стека протоколов TCP/IP. Для просмотра таблицы маршрутизации используется команда show ip route, как показано в нижеследующем примере. В нем также показано, что сеть маршрутизатора Rome (131.108.33.0) доступна через интерфейс Etnernet1 (131.108.16.2) и сеть маршрутизатора Paris (см. рисунок 4.4).

Пример. Выводимая командой **show ip route** информация

Paris# show ip route
Codes: I – IGRP derived, R - RIP derived, O – OSPF derived
C – connected, S – static, E – EGP derived, B – BGP derived
i – IS-IS derived, D – EGRP derived
* – candidate default route, IA – OSPF inter area route
El – OSPF external type 1 route, E2 – OSPF external type 2
route LI – IS-IS level-1 route, L2 – IS-IS level-2 route
EX – EIGRP external route

Gateway of last resort is not set
I 144.253.0.0 [100/1300] via 133.3.32.2 0:00:22 Ethernetl
131.108.0,0 is subnetted (mask is 255.255.255.0), 3 subnets
I 131.108.33.0 (100/180771) via 131.108.16.2, 0:01:29, Ethernetl
C 131.108.12.0 is directly connected, Ethernetl
C 101.108.16.0 is directly connected, Ethernet0
I 219.100.103.0 ЦС0/1200) via 133.3.32.2, 0:00:22, Ethernetl

4.1.11 Внесение изменений в конфигурацию маршрутизатора

Для внесения изменений в конфигурацию маршрутизатора необходимо войти в соответствующий режим и произвести эти изменения. Например, если какой-либо интерфейс отключен, то для его включения необходимо войти в режим глобальной конфигурации, затем – в режим настройки интерфейса и выполнить команду «no shutdown».

Для проверки внесенных изменений используется команда «show running-config». Эта команда отображает текущую конфигурацию. Если отображенные значения переменных неверны, то для их изменения можно выполнить одно из следующих действий:

– использовать команды конфигурации с префиксом no;

– перезапустить систему и перезагрузить оригинальный конфигурационный файл из энергонезависимой памяти маршрутизатора;

– удалить файл начальной конфигурации при помощи команды «erase startup-configuration», перезагрузить маршрутизатор и войти в режим установки.

Для сохранения конфигурационных переменных в энергонезависимом ОЗУ в привилегированном режиме выполните команду «Router# copy running-configuration startup-configuration»

В таблице 4.4 приведен список команд, позволяющих управлять содержимым энергонезависимой памяти в операционной системе Cisco IOS версии I l.x и более поздних.

Таблица 4.4 – Список команд конфигурации

Команда	Описание
configure memory	Загружает информацию о конфигурации из энергонезависимого ОЗУ (NVRAM)
erase startup-config	Очищает содержимое энергонезависимого ОЗУ (NVRAM)
copy running-config startup-config	Сохраняет текущую конфигурацию, находящуюся в ОЗУ (действующую конфигурацию) в энергонезависимое ОЗУ (загрузочную конфигурацию)
show startup-config	Отображает сохраненную конфигурацию, которая находится в энергонезависимом ОЗУ

4.1.12 Настройка защиты маршрутизатора паролями

Для защиты маршрутизатора от несанкционированного доступа используются пароли. Пароли могут быть установлены на доступ к виртуальной линии терминала и к линии консоли. Также паролем может быть защищен привилегированный EXEC-режим.

Для ограничения доступа паролем к привилегированному режиму в режиме настройки глобальной конфигурации введите команду «enable password». Однако этот пароль будет находиться в незашифрованном виде в конфигурационных файлах маршрутизатора. Для ввода пароля, который будет зашифрован, в привилегированном режиме введите команду «enable secret». Если пароль будет задан этой командой, то он будет использоваться вместо пароля, задаваемого командой «enable password». В этом случае в файлах конфигурации пароль будет содержаться в зашифрованном виде.

Для задания пароля на вход в консоль терминала используется команда «line console 0». Эту команду полезно использовать в сети, в которой к маршрутизатору имеет доступ большое количество людей. Задание пароля на доступ к консоли терминала позволит предотвратить несанкционированный доступ к маршрутизатору.

Парольной защиты требует также и telnet-доступ. В различных аппаратных платформах используется различное количество линий. Диапазон от 0 до 4 задает пять линий. Это означает, что одновременно могут быть установлены до пяти сеансов связи telnet. Для всех линий может быть один пароль или же для каждой линии его можно назначить индивидуально. Эта функция часто используется в больших сетях, обслуживаемых большим количеством сетевых администраторов. При возникновении в сети неразрешимых проблем и при всех занятых линиях доступа для восстановления может быть зарезервирована одна линия.

Для установки пароля к сеансу telnet-связи используется команда «line vty 0 4». В примере показаны различные пути настройки и защиты пароля.

Пароль, заданный командой «enable secret», не может быть прочитан; другой пользователь, получивший доступ к файлам конфигурации, может лишь перезаписать его, но никак не прочитать, поскольку для хранения пароля используется необратимое, одностороннее шифрование, что исключает восстановление пароля. Для запрета отображения пароля в виде открытого текста может быть использована команда «service password-encryption». Ее следует вводить в режиме глобальной конфигурации. Эта команда действует на все пароли, кроме того, который был указан с помощью команды «enable secret», поскольку он и так уже зашифрован. Команда «service password-encryption» позволяет зашифровать все пароли: привилегированного и непривилегированного пользователей, пароли доступа к консоли, через терминальные соединения, пароли линий и др.

Пример установки пароля.

Пароль консоли:

```
Router(config)# line console 0
Router(config-line)# login
Router(config-line)# password Cisco
```

Пароль виртуального терминала:

```
Router(config)# line vty 0 4
Router(config-line)# login
Router(config-line)# password Cisco
```

Пароль для доступа к привилегированному режиму

```
Router(config)# enable password san-fran
```

Шифрование пароля

```
Router(config)# enable secret [пароль]
```

Шифрование всех паролей

```
Router(config)# service password-encryption
```

Отмена шифрования всех паролей

Router(config)# no service password-encryption

4.1.13 Сохранение конфигурационных параметров

Для сохранения конфигурационных параметров в энергонезависимом ОЗУ в привилегированном режиме выполните команду «Router# copy running-configuration startup-configuration».

4.2 Задание к лабораторной работе

1 Ознакомьтесь с основными режимами конфигурирования маршрутизатора.

2 Подключитесь к маршрутизатору для его конфигурирования по схеме рисунка 4.5.

Рисунок 4.5 – Схема подключения к маршрутизатору
для его конфигурирования

3 Ознакомьтесь с результатом выполнения команд «show version», «show flash».

4 Назначьте имя маршрутизатору Router0.

5 Собрать модель сети из 3-х маршрутизаторов. Настроить интерфейсы на маршрутизаторах в соответствии с нижеприведенной информацией и протестировать соединения.

Router0:

Включите маршрутизатор.

Сконфигурируйте интерфейс FastEthernet 0/0

IP = 192.168.1.1

NetMask = 255.255.255.0

Router1

Включите маршрутизатор.

Сконфигурируйте интерфейс FastEthernet 0/0

IP = 192.168.1.2

NetMask = 255.255.255.0

Сконфигурируйте интерфейс Serial 1/0

IP = 192.168.2.1

NetMask = 255.255.255.0

Протестируйте интерфейс на тип подключения, если подключение DCE, то сконфигурируйте Clock Rate = 56000

Router2

Включите маршрутизатор.

Сконфигурируйте интерфейс Serial 1/0

IP = 192.168.2.2

NetMask = 255.255.255.0

Протестируйте интерфейс на тип подключения, если подключение DCE, то сконфигурируйте Clock Rate = 56000

6 Протестируйте все соединения. Покажите результаты выполнения преподавателю.

7 Соберите модель сети, состоящей из двух маршрутизаторов и двух компьютеров. Настройте в ней удаленное управление всеми элементами в соответствии со следующими данными.

Настройка марштуризатора Router0:

Настройте интерфейс FE0/0 с IP 192.168.1.1 и маской 255.255.255.0

Настройте интерфейс FE 0/1 с IP 192.168.0.1 и маской 255.255.255.0

Настройте линии виртуальных терминалов 0-4:

Установите пароль cisco

Установите motd-banner

Установите пароль cisco на доступ к привелегированному режиму работы с маршрутизатором, примените к ним настройки шифрования.

Сохраните конфигурацию маршрутизатора.

Настройка марштуризатора Router1

Настройте интерфейс FE0/0 с IP 192.168.1.2 и маской 255.255.255.0

Настройте интерфейс FE 0/1 с IP 192.168.2.1 и маской 255.255.255.0

Настройте линии виртальных терминалов 0-4:

Установите пароль cisco

Установите motd-banner

Установите пароль cisco на доступ к привелегированному режиму работы с маршрутизатором, примените к ним настройки шифрования.

Сохраните конфигурацию маршрутизатора.

Настройте персональный компьютер PC0

Установите IP 192.168.x.x и маску 255.255.255.0

Настройте персональный компьютер PC1

Установите IP 192.168.x.x и маску 255.255.255.0

7 Попробуйте соединиться по telnet с компьютера PC0 на маршрутизатор Router0, с компьютера PC1 на маршрутизатор Router 1, с маршрутизатора Router 1 на Router0 и наоборот. При положительном результате покажите результат преподователю.

Указание: при выполнении пунктов 6 и 7 задания руководствуйтесь пунктом 1.1.4 лабораторной работы №1.

4.3 Содержание отчета

1 Цель работы.
2 Схемы топологий моделей сетей.
3 Конфигурационные файлы маршрутизаторов.
4 Таблицы параметров настройки.
5 Выводы.

4.4 Контрольные вопросы

1 К оборудованию какого уровня относятся маршрутизаторы?
2 Где в маршрутизаторе хранится информация о параметрах и инструкциях управления потоками данных?
3 Каково назначение маршрутизаторов?
4 Перечислите основные функции маршрутизаторов.
5 Назовите основные компоненты маршрутизатора.
6 Назовите типы интерфейсов маршрутизаторов.
7 Что необходимо сделать для получения доступа к маршрутизатору?
8 Что означают термины DTE, DCE, DSU, CSU?
9 Каков порядок настройки последовательного интерфейса маршрутизатора?
10 Как настроить Ethernet-интерфейс маршрутизатора?
11 Какие команды используются для тестирования работоспособности соединений?
12 Что нужно сделать для внесения изменений в конфигурационные файлы маршрутизатора?
13 Как установить пароль для защиты маршрутизатора от несанкционированного доступа?
14 Перечислите команды группы «show» и их функции.

Лабораторная работа №5

МАРШРУТИЗАЦИЯ. ПОНЯТИЕ АДМИНИСТРАТИВНОГО РАССТОЯНИЯ МАРШРУТА. СТАТИЧЕСКАЯ МАРШРУТИЗАЦИЯ

Цель работы: приобрести общие знания о статической маршрутизации на сетях, а также получить практические навыки настройки статических маршрутов.

5.1 Теоретическая часть

5.1.1 Введение в статическую маршрутизацию

Маршрутизация представляет собой выбор направлений передачи данных от одной сети другой. Эти направления, также называемые маршрутами, могут предоставляться динамически другими маршрутизаторами. Однако они могут также назначаться маршрутизатору статически.

5.1.2 Основы маршрутизации

Маршрутизация – это процесс, который используется маршрутизатором для пересылки пакета в сеть получателя. Маршрутизатор принимает решения, основываясь на IP-адресе получателя пакета. Для того чтобы переслать пакет в требуемом направлении, все устройства на пути его следования используют IP-адрес получателя. Этот адрес позволяет пакету достичь требуемого пункта назначения. Для принятия правильного решения маршрутизаторы должны знать направления к удаленным сетям. При использовании динамической маршрутизации это направление к удаленным сетям маршрутизатор узнает от других маршрутизаторов сети. При использовании статической маршрутизации (static routing) информация об удаленных сетях задается вручную сетевым администратором.

Поскольку статические маршруты конфигурируются вручную, любые изменения сетевой топологии требуют участия сетевого администратора для добавления и удаления статических маршрутов в соответствии с этими изменениями. В крупных сетях такая ручная поддержка таблиц маршрутизации может потребовать огромных затрат времени сетевого администратора. В небольших сетях, в которых изменения незначительны, поддержка статических маршрутов особых затрат не требует. Статическая маршрутизация не обладает возможностями масштабирования, имеющимися у динамической маршрутизации, из-за дополнительных требований к настройке и необходимости вмешательства администратора. Однако и в крупных сетях часто конфигурируются статические маршруты для специальных целей в комбинации с протоколом динамической маршрутизации. Несмотря на то что динамические протоколы маршрутизации могут автоматически определять маршруты, для этого они все же должны быть сначала активизированы и сконфигурированы сетевым администратором.

Поскольку информация статических маршрутов вводится в конфигурацию маршрутизатора вручную сетевым администратором, то в каждом случае, когда изменение топологии объединенной сети требует обновления статических маршрутов, это обновление должно выполняться вручную.

Статическая маршрутизация имеет несколько полезных приложений. При динамической маршрутизации имеется тенденция к распространению всей информации об объединенной сети. Однако по соображениям безопасности иногда требуется скрыть некоторые части сети. Статическая маршрутизация позволяет пользователю указать, какая информация может распространяться относительно таких скрытых сетей с ограниченным доступом.

Если доступ к сети может быть получен только по одному маршруту, то одного статического маршрута может оказаться вполне достаточно.

5.1.3 Принцип действия статических маршрутов

Функционирование статических маршрутов может быть описано тремя положениями.

1 Сетевой администратор задает статический маршрут.

2 Маршрутизатор заносит этот маршрут в свою таблицу маршрутизации.

3 Пакеты пересылаются с использованием указанного статического маршрута.

Поскольку статический маршрут конфигурируется вручную, для его установки на маршрутизаторе сетевой администратор должен ввести соответствующую команду ip route. Эта команда имеет следующий синтаксис:

Router(config)#ip route prefix mask (ip-address\interface-type interface number)[distance]

На рисунке 5.1 сетевому администратору маршрутизатора *Hoboken* требуется сконфигурировать статический маршрут к сетям 172.16.1.0/24 и 172.16.5.0/24, подсоединенным к другим маршрутизаторам.

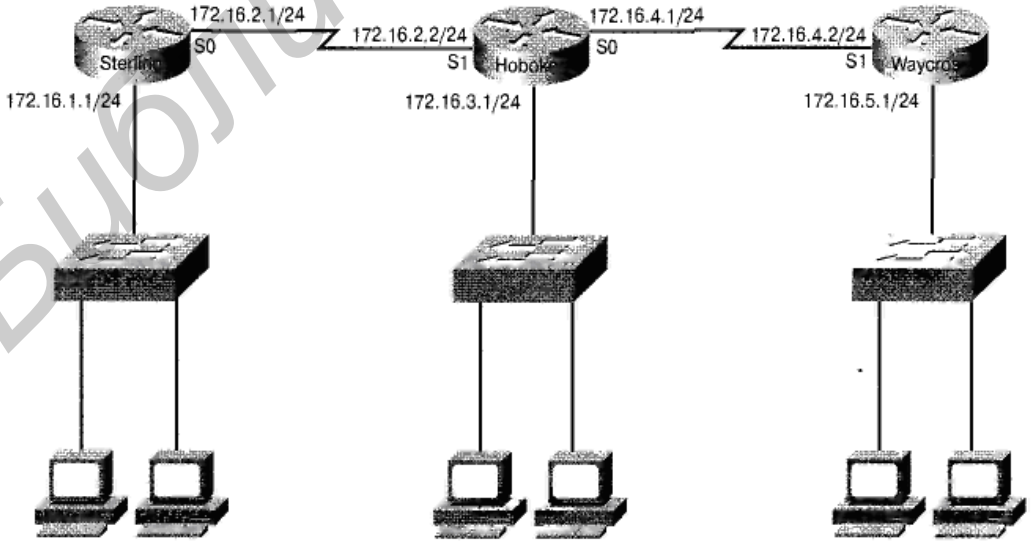


Рисунок 5.1 – Статические маршруты

Для решения этой задачи сетевой администратор может ввести одну или две команды. В примере 5.1 (см. ниже) для этого указывается выходной интерфейс (Serial 0). В примере 5.2 указывается IP-адрес смежного (соседнего) маршрутизатора (172.16.2.2). Любая из этих команд задает статический маршрут в таблице маршрутизации маршрутизатора *Hoboken.*

Пример 5.1. Статический маршрут с использованием интерфейса:

Sterling(config)#ip route 172.16.3.0 255.255.255.0 s0

Пример 5.2. Пример статического маршрута с использованием IP-адреса маршрутизатора:

Sterling(config)#ip route 172.16.3.0 255.255.255.0 172.16.2.2

Единственным различием между этими двумя командами является **административное расстояние (administrative distance)**, назначаемое маршруту при его занесении в таблицу маршрутизации. Под административным расстоянием понимается необязательный параметр, который характеризует надежность маршрута. Меньшему значению административного расстояния соответствует менее надежный маршрут. Такое утверждение означает, что маршрут с меньшим административным расстоянием будет установлен в таблицу маршрутизации прежде, чем маршрут с большим административным расстоянием. Стандартно при использовании адреса следующего перехода административное расстояние устанавливается равным 1. При задании выходного интерфейса для административного расстояния устанавливается значение 0. В таблице 5.1 приведены административные расстояния для каждого поддерживаемого протокола. Маршрутам с меньшим административным расстоянием отдается предпочтение по сравнению с аналогичными маршрутами с большим административным расстоянием. Если требуется установить административное расстояние, отличающееся от стандартного, то следует ввести значение в интервале от 0 до 255 после адреса следующего перехода или указания выходного интерфейса, как показано ниже.

ip route 172.16.3.0 255.255.255.0 192.168.2.1 255

Если маршрутизатор по каким-либо причинам не может использовать выходной интерфейс, заданный в маршруте, то этот маршрут не будет использоваться устройством. Такая ситуация означает, что если указанный интерфейс неработоспособен, то маршрут не будет занесен в таблицу маршрутизации.

Иногда статические маршруты используются в качестве резервных. На маршрутизаторе может быть сконфигурирован статический маршрут, который будет использован только в том случае, если не удастся отправить данные по динамически созданному маршруту. Для использования статического маршрута в этом качестве его административное расстояние должно быть установлено большим, чем у маршрута, предоставляемого протоколом динамической маршрутизации.

Таблица 5.1 – Административные расстояния в операционной системе Cisco IOS

Источник маршрута	Стандартное значение административного расстояния
Подсоединенный интерфейс	0
Статический маршрут	1
Суммарный маршрут протокола EIGRP	5
Протокол BGP	20
Внутренний маршрут протокола EIGRP	90
Протокол IGRP	100
Протокол OSPF	110
Протокол IS-IS	115
Протокол RIP	120
Протокол EGP	140
Внешние маршруты протокола EIGRP	170
Внутренние маршруты BGP	200
Неизвестен	255

5.1.4 Конфигурирование статических маршрутов

Чтобы сконфигурировать статические маршруты, необходимо выполнить следующее.

Этап 1. Определить все требуемые сети-получатели, их маски подсетей и префиксы. В качестве адреса шлюза может выступать либо локальный интерфейс маршрутизатора, либо адрес следующего транзитного перехода, который ведет к требуемому пункту назначения.

Термином **префикс** зачастую обозначают адреса сетей. Наиболее полное определение данного термина подразумевает, что под ним обычно понимается адрес, узловые биты маски которого равны нулю, а сетевые – единице. Префиксный адрес также может подразумевать в себе суммарный адрес. Например, можно суммировать (или, как часто говорят, агрегировать) несколько указанных ниже адресов в один суммарный с префиксом 192.168.0.0/22. Обозначение «/22» указывает на то, что первые 22 бита являются префиксом. Сети, которые войдут в указанный суммарный адрес:

192.168.0.0/24
192.168.1.0/24
192.168.2.0/24
192.168.3.0/24

Этап 2. Войти в режим глобального конфигурирования.

Этап 3. Ввести команду **ip route** с адресом сети-получателя и маской подсети, за которыми следует адрес следующего транзитного узла (о нем говорилось на этапе 1). Указание административного расстояния не является обязательным.

Этап 4. Повторить этап 3 для всех сетей-получателей, к которым требуется задать статический маршрут.

Этап 5. Выйти из режима глобального конфигурирования.

Этап 6. Сохранить активную конфигурацию в памяти NVRAM с помощью команд copy running-config startup-config и write memory.

В сети, которая показана на рисунке 5.1, представлена простая структура с тремя маршрутизаторами. Маршрутизатор **Hoboken** должен быть сконфигурирован таким образом, чтобы он обеспечивал доступ к сетям с адресами 172.16.1.0 и 172.16.5.0. В обеих сетях маска подсети имеет вид 255.255.255.0.

Пакеты, у которых получателем является сеть 172.16.1.0, требуется направлять на маршрутизатор **Sterling**. Пакеты, у которых получателем является сеть 172.16.5.0, требуется направлять на маршрутизатор **Waycross.** Для этого необходимо сконфигурировать статические маршруты с использованием выходных интерфейсов маршрутизатора S0 и S1, как показано в примере 5.3.

Пример 5.3. Задание выходных интерфейсов IP-маршрутов.

Hoboken(config)#ip route 172.16.1.0 255.255.255.0 s1

Hoboken(config)#ip route 172.16.5.0 255.255.255.0 s0

Оба статических маршрута конфигурируются с использованием локального интерфейса в качестве шлюза к сетям-получателям. Поскольку административное расстояние не указано, при занесении маршрутов в таблицу маршрутизации оно стандартно принимается равным нулю. Следует обратить внимание на то, что административное расстояние, равное нулю, также присуще непосредственно подсоединенной сети.

Те же статические маршруты могут быть сконфигурированы с использованием в качестве шлюза адреса следующего перехода. Первый маршрут к сети 172.16.1.0 проходит через шлюз 172.16.2.1. У сети 172.16.5.0 шлюз имеет адрес 172.16.4.2. В примере 5.4 показано, как сконфигурировать статические маршруты с использованием адреса интерфейса следующего транзитного перехода; в него включены комментарии (которым предшествует символ «!»), которые не будут отображены в файле конфигурации. Поскольку административное расстояние явным образом не задано, стандартно оно устанавливается равным единице.

Пример 5.4. Статические маршруты с использованием адреса следующего транзитного перехода и комментариями:

Hoboken(config)#ip route 172.16.1.0 255.255.255.0 172.16.2.1 ! Данный маршрут ведет к локальной сети Sterling

Hoboken(config)#ip route 172.16.1.0 255.255.255.0 172.16.4.2 ! Данный маршрут ведет к локальной сети Waycross

5.1.5 Конфигурирование пересылки пакетов по стандартному маршруту

Стандартные маршруты используются маршрутизаторами в тех случаях, когда адрес сети-получателя пакета не совпадает ни с одним из маршрутов, содержащихся в таблице маршрутизации. Стандартные маршруты, как правило, конфигурируются для передачи потоков данных через сеть Internet,

поскольку нерационально и нет необходимости поддерживать все маршруты ко всем сетям Internet. Стандартный маршрут фактически является специальным статическим маршрутом, использующим следующий формат:

ip route 0.0.0.0 0.0.0.0 [next-hop-address/outgoing interface]

Для конфигурирования маршрутов по умолчанию необходимо выполнить описанные ниже действия.

Этап 1. Войти в режим глобальной конфигурации.

Этап 2. Ввести в командной строке команду ip route с адресом 0.0.0.0 для сети-получателя и значением 0.0.0.0 для маски подсети. Шлюзом стандартного маршрута может быть либо локальный интерфейс маршрутизатора, через который осуществляется связь с внешними сетями, либо адрес маршрутизатора следующего перехода. В большинстве случаев предпочтительнее задавать IP-адрес маршрутизатора следующего перехода.

Этап 3. Выйти из режима глобального конфигурирования.

Этап 4. Сохранить текущую конфигурацию в памяти NVRAM с помощью команды «copy running-config startup-config».

В примере 5.5 приведена команда, которую требуется выполнить для конфигурирования стандартных маршрутов для маршрутизатора.

Пример 5.5. Стандартный маршрут для маршрутизатора Waycross

Waycross(config)#ip route 0.0.0.0 0.0.0.0 s1

5.1.6 Проверка статических маршрутов

После того как статические маршруты сконфигурированы, важно проверить, что они находятся в таблице маршрутизации и пересылка пакетов по ним осуществляется требуемым образом. Для просмотра активной конфигурации в памяти NVRAM и проверки правильности ввода статических маршрутов используется команда «show running-config». Для проверки наличия маршрута в таблице маршрутизации используется команда «show ip route».Для тестирования конфигурации статических маршрутов следует выполнить описанные ниже действия.

Этап 1. В привилегированном режиме ввести команду «show running-config» для просмотра активной конфигурации.

Этап 2. Проверить правильность строк статических маршрутов. Если маршрут введен неправильно, следует вернуться в режим глобального конфигурирования, удалить неверный маршрут и ввести правильный.

Этап 3. Ввести команду «show ip route».

Этап 4. Проверить, что сконфигурированный маршрут находится в таблице маршрутизации.

5.2 Задание к лабораторной работе

Вам необходимо настроить статическую маршрутизацию на двух маршрутизаторах Router_Brest и Router_Minsk сети, изображенной на рисунке 5.2, так, чтобы из вашей сети был доступ к ISP (Internet Service Provider –

интернет сервис провайдер). В вашем распоряжении только один персональный компьютер («Ваш ПК) и пароли к доступу по telnet к маршрутизаторам (password: cisco).

Указание: при выполнении работы руководствуйтесь пунктом 1.1.4 лабораторной работы №1.

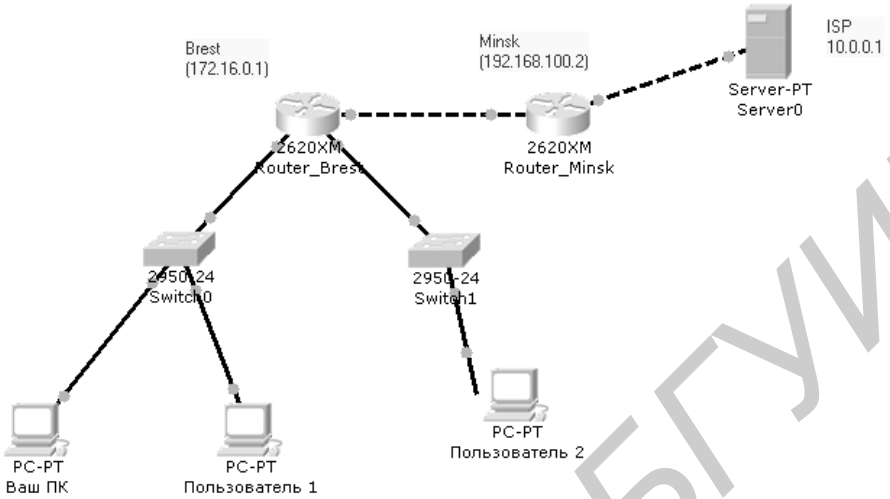


Рисунок 5.2 – Задание к лабораторной работе

5.3 Содержание отчета

1 Цель работы.
2 Схема сети.
3 Конфигурационные файлы маршрутизаторов.
4 Выводы.

5.4 Контрольные вопросы

1 Что такое маршрутизация?
2 В чем отличие статических и динамических маршрутов?
3 Что такое административное расстояние?
4 Назовите этапы конфигурирования статических маршрутов.
5 Для чего нужен стандартный маршрут?
6 Как проверить правильность конфигурирования статических маршрутов?

Лабораторная работа №6

ПОНЯТИЕ ДИНАМИЧЕСКОЙ МАРШРУТИЗАЦИИ. ПРОТОКОЛЫ МАРШРУТИЗАЦИИ. ПРОТОКОЛ RIP

Цель работы: приобрести общие знания о динамической маршрутизации в сетях, принципах работы дистанционно-векторных протоколов, а также получить практические навыки в настройке протокола маршрутизации RIP.

6.1 Теоретическая часть

6.1.1 Введение в динамическую маршрутизацию

Динамическая маршрутизация (dynamic routing) необходима для того, чтобы сети могли обновлять свои таблицы маршрутизации и быстро адаптироваться к изменениям в топологии и состоянии соединений. Протоколы динамической маршрутизации могут также для повышения эффективности работы сети направлять потоки данных одного и того же сеанса по нескольким маршрутам. Этот механизм представляет собой *распределение нагрузки (load sharing)* между несколькими каналами и устройствами.

Динамические маршруты устанавливаются следующим образом. После того как сетевой администратор вводит команды конфигурирования динамической маршрутизации, информация о маршрутах обновляется автоматически в процессе маршрутизации при каждом получении из сети новой информации о маршрутах. Маршрутизаторы обмениваются сообщениями об изменениях в топологии сети в процессе динамической маршрутизации.

6.1.2 Операции динамической маршрутизации

Успешное функционирование динамической маршрутизации зависит от выполнения маршрутизатором двух его основных функций:

– поддержки таблицы маршрутизации в актуальном состоянии;

– распространения информации в виде анонсов и обновлений маршрутов среди остальных маршрутизаторов.

При распространении информации о сети механизм динамической маршрутизации использует один из протоколов маршрутизации. Такой протокол определяет набор правил, используемых маршрутизатором при осуществлении связи с соседними маршрутизаторами. Например, протокол маршрутизации определяет:

– каким образом рассылаются обновления маршрутов;

– какая информация содержится в обновлениях;

– как часто рассылаются обновления;

– каким образом выполняется поиск получателей обновлений.

6.1.3 Определение длины сетевых маршрутов

При обновлении алгоритмом маршрутизации таблицы маршрутизации первичной задачей устройства является выбор наилучшего маршрута для включения его в таблицу. Каждый алгоритм маршрутизации использует свой собственный способ выбора наилучшего маршрута. Для этого он генерирует определенное значение, называемое **метрикой (metric)**, для каждого маршрута в сети. Обычно чем меньше значение метрики, тем лучше маршрут.

Могут использоваться простые метрики, которые вычисляются на основе одной характеристики, такой, например, как количество переходов на маршруте, или более сложные метрики, использующие несколько параметров маршрутов. Ниже перечислены наиболее часто используемые в метриках характеристики.

Полоса пропускания (Bandwidth) описывает пропускную способность канала (обычно канал Ethernet со скоростью 10 Мбит/с предпочтительнее выделенной линии со скоростью 64 Кбит/с).

Задержка (Delay) представляет собой время, требуемое пакету для прохождения по каналу от отправителя до получателя.

Нагрузка (Load) – это степень использования сетевых ресурсов на маршрутизаторе или канале.

Надежность (Reliability) обычно характеризует уровень ошибок в сетевом канале.

Количество переходов (Hop count) – это число маршрутизаторов, через которые должен пройти пакет до поступления в пункт назначения.

Стоимость (Cost) представляет собой произвольное значение, обычно вычисляемое на основе ширины полосы пропускания, финансовых затрат или других характеристик, выбираемых сетевым администратором.

6.1.4 Введение в протоколы маршрутизации

Протоколы маршрутизации (иначе – маршрутизирующие протоколы) отличаются от маршрутизируемых протоколов как по своим функциям, так и по задачам, которые перед ними ставятся. Протокол маршрутизации – это средство коммуникации между маршрутизаторами, которое позволяет устройствам совместно использовать информацию о сетях и определять расстояние до разных узлов и сетей. Информация, которую один маршрутизатор получает от другого (посредством протокола маршрутизации), используется для построения и поддержания в актуальном состоянии таблицы маршрутизации.

К наиболее распространенным протоколам маршрутизации локальных сетей можно отнести следующие:

– протокол маршрутной информации (Routing Information Protocol – RIP);

– протокол маршрутизации внутреннего шлюза (Interior Gateway Routing Protocol –IGRP);

– усовершенствованный протокол маршрутизации внутреннего шлюза (Enhanced Interior Gateway Routing Protocol – EIGRP);

– протокол выбора кратчайшего маршрута (Open Shortest Path First – OSPF).

Маршрутизируемый (routed protocol), или **сетевой протокол** – это любой сетевой протокол, предоставляющий в своем адресе сетевого уровня достаточно информации для пересылки пакета от одного узла другому на основе используемой схемы адресации. Маршрутизируемые протоколы определяют форматы полей внутри пакета. Пакеты обычно передаются от одной конечной системы другой.

Маршрутизируемые протоколы (часто их называют протоколами передачи данных) используются для доставки пользовательской информации. Маршрутизируемый протокол содержит достаточное количество информации в адресе сетевого уровня, которую позволит доставить от одного узла другому в рамках используемой схемы адресации.

К наиболее распространенным маршрутизируемым протоколам сетей можно отнести следующие:

– Internet-протокол (IP – Internet Protocol);

– межсетевой пакетный обмен (Internetwork Packet Exchange – IPX).

6.2.5 Автономные системы

Автономная система (Autonomous System – AS) – это набор сетей, которые находятся под единым административным управлением и в которых используются единая стратегия и правила маршрутизации. Автономная система для внешних сетей представляется как некий единый объект. Ее могут поддерживать и несколько операторов-владельцев, и один, они будут нести ответственность за правильную маршрутизацию.

Американский реестр Internet-номеров (American Registry of Internet Numbers-ARIN), провайдер службы или сетевой администратор присваивает номер (идентификатор) каждой автономной системе. Идентификатор автономной системы представляет собой 16-битное число. Некоторые протоколы маршрутизации, такие как фирменные протоколы IGRP и EIGRP корпорации Cisco, используют такое понятие, как «номер автономной системы» в своей конфигурации; в действительности же нет никакой необходимости устанавливать туда реальный номер. Этот параметр представляет собой просто идентификатор процесса. Для двух указанных протоколов маршрутизации нет необходимости использовать номер системы, который получен от реестра ARIN, или частный номер автономной системы.

Автономные системы (AS) делят объединенную сеть на несколько меньших и легче управляемых сетей. Каждая автономная система имеет свой набор правил и политик, а ее номер является глобально уникальным, т. е. отличает ее от всех остальных автономных систем мира.

6.1.6 Назначение протоколов маршрутизации и цели использования автономных систем

Целью использования протокола маршрутизации является построение и поддержка таблицы маршрутизации. В этой таблице содержатся информация об известных маршрутизатору сетях и соответствующие порты, ведущие к этим сетям. Маршрутизаторы используют протоколы маршрутизации для управления информацией, полученной от других маршрутизаторов, и информацией, получаемой из конфигурации своих собственных интерфейсов.

Протокол маршрутизации идентифицирует все доступные маршруты, помещает лучшие таблицы в таблицу маршрутизации и удаляет из нее маршруты, если они становятся недоступными. Маршрутизатор использует информацию таблицы маршрутизации для пересылки пакетов сетевых (маршрутизируемых) протоколов.

Основой динамической маршрутизации является алгоритм маршрутизации. При любом изменении топологии сети, связанном с ее увеличением, реконфигурацией или выходом из строя устройств, информация о сети должна быть обновлена. Она должна точно и последовательно отражать текущее состояние новой топологии сети.

Когда все маршрутизаторы объединенной сети имеют одинаковую информацию, это означает, что в ней произошла конвергенция. Быстрая конвергенция является желательной, поскольку она сокращает период принятия неправильных решений маршрутизации.

Зачастую можно обнаружить, что крупные сети, например сети университетов, крупных компаний, даже школ, имеют свою собственную автономную систему, каждая подсеть или сегмент сети университета может быть построена с использованием какого-либо протокола маршрутизации, статических маршрутов; тем не менее все отдельные подсети в организации соединены между собой статическими или коммутируемыми каналами и входят в состав единой автономной системы.

6.1.7 Идентификация класса протокола маршрутизации

Большинство алгоритмов маршрутизации может быть отнесено к одной из трех категорий:

1) дистанционно-векторный протокол;
2) протокол с учетом состояния канала;
3) гибридный протокол.

Дистанционно-векторный протокол (distance vector routing protocol) определяет направление, или вектор, и расстояние до нужного узла объединенной сети. **Протокол с учетом состояния канала (link-state routing protocol)**, также называемый алгоритмом выбора кратчайшего пути (shortest path first – SPF), воссоздает топологию всей сети. *Сбалансированный гибридный протокол (balanced hybrid routing protocol)* соединяет в себе опре-

деленные черты обоих алгоритмов: дистанционно-векторного и алгоритма с учетом стояния канала.

6.1.8 Особенности дистанционно-векторных протоколов

При использовании дистанционно-векторных алгоритмов между маршрутизаторами они периодически пересылают копии таблиц маршрутизации друг другу. В этих регулярных обновлениях маршрутизаторы сообщают друг другу об изменении топологии сети. Дистанционно-векторные алгоритмы маршрутизации также называются алгоритмами Беллмана-Форда (Bellman-Ford).

На рисунке 6.1 каждый маршрутизатор получает таблицу маршрутизации от соседних маршрутизаторов. В частности, маршрутизатор Б получает информацию от маршрутизатора А. Маршрутизатор Б добавляет значение вектора расстояния, количество переходов, что увеличивает результирующий вектор расстояния. После этого маршрутизатор Б передает свою новую таблицу маршрутизации своему соседу, маршрутизатору В. Такой пошаговый процесс происходит на всех соседних маршрутизаторах.

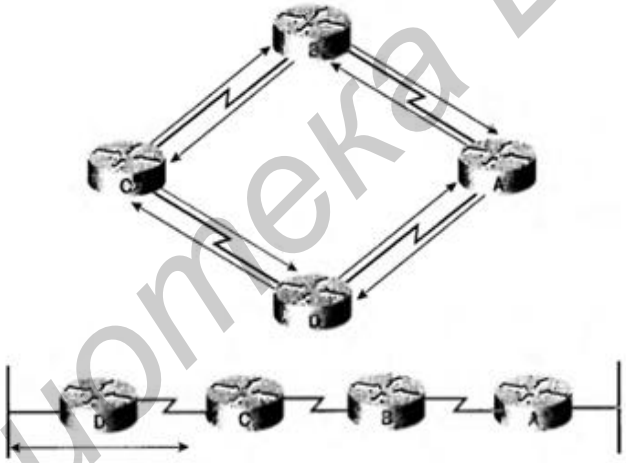


Рисунок 6.1 – Концепция дистанционно-векторной маршрутизации

В дистанционно-векторном алгоритме накапливаются расстояния в сети, что позволяет поддерживать базу данных, содержащую информацию о топологии сети. Однако дистанционно-векторные алгоритмы не предоставляют маршрутизатору точную топологию всей сети, поскольку каждому маршрутизатору известны только соседние с ним маршрутизаторы.

Каждый маршрутизатор, использующий дистанционно-векторную маршрутизацию, начинает свою работу с определения соседних маршрутизаторов.

Формирование вектора расстояния. Для каждого интерфейса, ведущего к непосредственно подсоединенной сети, вектор расстояния устанавливается равным нулю. По мере того как процесс расчета вектора расстояния продолжается, маршрутизаторы находят наилучший маршрут к сетям-

получателям на основе информации, которую они получают от своих соседей. Например, маршрутизатор А узнает о других сетях на основе информации», которую он получает от маршрутизатора Б. В каждой из позиций таблицы маршрутизации есть суммарный вектор расстояния, который показывает, на каком расстоянии находится соответствующая удаленная сеть.

Обновление таблицы маршрутизации происходит при изменении топологии сети. По мере формирования векторов расстояния изменения топологии заносятся в таблицы маршрутизации последующих маршрутизаторов. Дистанционно-векторные алгоритмы требуют, чтобы каждый маршрутизатор пересылал всю таблицу маршрутизации каждому из своих соседей. В этой таблице содержатся общая оценка маршрута, определяемая метрикой, и логический адрес маршрутизатора на пути к каждой сети, имеющейся в таблице.

6.1.9 Обновления маршрутов

Каждый маршрутизатор получает таблицу маршрутизации от соседних, непосредственно подсоединенных к нему маршрутизаторов. Например, как показано на рисунке 6.2, маршрутизатор Б получает информацию от маршрутизатора А. Маршрутизатор Б добавляет свое значение к вектору расстояния (например количество переходов) и передает новую таблицу маршрутизации соседнему маршрутизатору.

Рисунок 6.2 – Обработка изменений топологии дистанционно-векторным протоколом маршрутизации

Подобный пошаговый процесс происходит между всеми соседними маршрутизаторами. Вектор расстояния можно сравнить с дорожными знаками на шоссе. Эти знаки указывают направление к пункту назначения и расстояние до него. Далее по этому же шоссе могут встретиться знаки, указывающие то же направление, однако указываемое ими расстояние будет меньшим. Уменьшение этого расстояния при последующем движении свидетельствует о движении в правильном направлении.

6.1.10 Основы маршрутизации по состоянию канала

Вторым базовым алгоритмом маршрутизации является алгоритм выбора маршрута по состоянию канала. Такие алгоритмы известны как алгоритмы Дейкстры (Dijkstra) или как алгоритмы выбора кратчайшего пути (Shortest Path First-SPF). Они поддерживают сложную базу топологической информации.

В то время как дистанционно-векторные алгоритмы не содержат определенной информации об удаленных сетях и удаленных маршрутизаторах, алгоритмы с использованием состояния канала поддерживают полную информацию об удаленных маршрутизаторах и их соединениях друг с другом. При маршрутизации по состоянию канала используются следующие компоненты:

– анонсы состояния канала (Link-State Advertisemen-LSA*)*. Эти обьявления представляют собой небольшие пакеты, рассылаемые между маршрутизаторами и содержащие информацию о маршрутах;

– топологическая база данных (Topological Database). Эта база включает в себя информацию, полученную в сообщениях LSA;

– алгоритм выбора кратчайшего пути (Shortest Path First – SPF*)*. Соответствующий алгоритм осуществляет вычисления над базой данных, результатом чего является построение связующего дерева протокола SPF;

– таблица маршрутизации (Routing table). Эта таблица содержит известные маршруты и соответствующие им интерфейсы.

Такая концепция маршрутизации на основе состояния канала была реализована в протоколе маршрутизации, называемом протоколом выбора первого кратчайшего пути (Open Shortest Path First-OSPF). Основные положения и операции протокола состояния канала связи OSPF описаны в документе RFC 1583.

6.1.11 Процесс обнаружения сетей для маршрутизации по состоянию канала

Маршрутизаторы обмениваются сообщениями LSA, начиная с непосредственно подсоединенных сетей. Каждый маршрутизатор параллельно с остальными создает топологическую базу данных, состоящую из информации, полученной из этих сообщений LSA.

Алгоритм SPF вычисляет доступность сетей. Маршрутизатор строит логическую топологию в виде дерева, корнем которого является он сам, а ветвями – все возможные маршруты ко всем сетям, входящим в объединенную сеть протокола стояния канала. После этого маршруты сортируются с помощью алгоритма выбора кратчайшего пути (Shortest Path First-SPF). Маршрутизатор заносит наилучшие маршруты и связанные с ними интерфейсы в таблицу маршрутизации. Маршрутизатор также поддерживает другие базы данных – топологических элементов и подробностей состояния каналов.

6.1.12 Обмен информацией о маршрутах в протоколах с учетом состояния каналов

Для создания общей картины всей сети в протоколах с учетом состояния канала используются специализированные механизмы обнаружения сетей. Такая подробная информация совместно используется всеми маршрутизаторами объединенной сети. Информацию о топологии можно сравнить с наличием нескольких идентичных карт города. Для обнаружения сетей в протоколе

маршрутизации по состоянию канала используются перечисленные ниже процессы.

Маршрутизаторы обмениваются друг с другом LSA-сообщениями. Каждый маршрутизатор начинает построение своей таблицы маршрутизации с непосредственно подсоединенных к нему сетей, от которых он получает информацию непосредственно «из первых рук».

Каждый маршрутизатор параллельно с остальными создает топологическую базу данных, состоящую из информации, полученной из всех LSA-сообщений объединенной сети.

Если маршрутизатор узнает об изменении состояния канала, он рассылает эту информацию всем остальным маршрутизаторам объединенной сети с тем, чтобы они могли ее использовать для маршрутизации. Для того чтобы закончилась конвергенция, каждый маршрутизатор поддерживает информацию о соседних маршрутизаторах, их именах, состоянии интерфейсов и стоимости каналов к соседним устройствам. Маршрутизатор создает пакет LSA; в котором содержится перечисленная информация наряду с информацией о новых соседях, изменениях в стоимостях каналов и о каналах, которые перестали функционировать. Затем этот пакет LSA направляется всем остальным маршрутизаторам.

6.1.13 Три проблемы в протоколах состояния канала

При использовании протоколов состояния канала возникают три основные проблемы:

– перегрузка процессора служебной информацией;

– повышенные требования к памяти;

– потребление процессом маршрутизации значительной части полосы пропускания.

Маршрутизаторы, на которых работают протоколы с учетом состояния канала, требуют большего объема памяти и выполняют больший объем обработки данных, чем при использовании дистанционно-векторного протокола маршрутизации. Они должны иметь достаточно памяти для хранения большого объема информации в различных базах данных, поддержки логического дерева и таблицы маршрутизации. Первоначальные потоки маршрутных данных о состоянии каналов занимают большую часть полосы пропускания, поскольку в первоначальной фазе обнаружения сетей все маршрутизаторы, использующие протоколы с маршрутизацией по состоянию канала, рассылают друг другу пакеты LSA. Эта рассылка в значительной степени заполняет сеть и временно уменьшает полосу пропускания, доступную для передачи данных пользователей. После этого временного переполнения протоколы состояния канала обычно требуют лишь минимальной полосы пропускания для рассылки нечастых или вызванных особыми изменениями в сети пакетов LSA, отражающих эти изменения.

6.1.14 Дополнительная информация: функции гибридных протоколов маршрутизации

Третий тип протоколов маршрутизации, называемых протоколами сбалансированной гибридной маршрутизации, соединяет в себе черты как дистанционно-векторных протоколов, так и протоколов с учетом состояния каналов связи. Протоколы сбалансированной гибридной маршрутизации для определения наилучших маршрутов используют векторы расстояния с более точными метриками. Однако они отличаются от дистанционно-векторных протоколов тем, что обновления баз данных маршрутизации происходят не периодически, а только при изменении топологии сети. Как и протоколы состояния канала связи, сбалансированные гибридные протоколы обладают быстрой сходимостью. Однако они отличаются от дистанционно-векторных протоколов и от протоколов с учетом состояния канала связи тем, что они в меньшей степени используют полосу пропускания, память и создают меньшую нагрузку на процессор для обработки служебной информации. Примером гибридного протокола может служить усовершенствованный протокол внутреннего шлюза (Enhanced Interior Gateway Routing Protocol-EIGRP).

6.1.15 Конфигурирование службы маршрутизации

Для включения на маршрутизаторе протокола IP-маршрутизации должны быть установлены как глобальные, так и локальные параметры интерфейса. Глобальные установки включают в себя выбор протокола маршрутизации, такого как IGRP, EIGRP или OSPR. Главной задачей, решаемой в режиме конфигурирования маршрутизации, является указание IP-адресов сетей. Для связи с другими маршрутизаторами динамическая маршрутизация использует широковещательные адреса многоадресатной рассылки. Для поиска наилучших маршрутов к каждой сети или подсети маршрутизаторы используют какую-либо метрику маршрутизации.

Процесс конфигурирования маршрутизации начинается с выполнения команды **router**. Эта команда имеет следующий синтаксис:

Router(config)#**router** protocol (process-id | autonomous-system),

где

– под параметром **protocol** понимается один из протоколов маршрутизации R1P, IGRP или EIGRP;

– параметр **prrocess-id** или **autonomous-system** содержит идентификатор процесса маршрутизации или номер автономной системы, используемой в протоколах IGRP и EIGRP,

Команда «network» является необходимой, поскольку она позволяет протоколу маршрутизации идентифицировать интерфейсы, которые принимают участие в правке и получении сообщений обновления маршрутов. Команда «network» имеет следующий синтаксис:

Router(config-router)#network **network number**,

где параметр «network number» представляет собой номер (IP-адрес) непосредственно подсоединенной сети.

Для протоколов RIP и IGRP номер сети должен базироваться на классах сетевых адресов, а не на адресах подсетей или индивидуальных адресах узлов.

В качестве возможных адресов сетей могут выступать только номера (т. е. адреса) сетей классов A, B и C.

На Internet-уровне стека протоколов TCP/IP маршрутизатор может использовать протокол IP-маршрутизации для осуществления маршрутизации путем реализации конкретного алгоритма. Примеры протоколов IP-маршрутизации:

– протокол маршрутной информации (Routing Information Protocol-RIP) – дистанционно-векторный протокол внутренней маршрутизации;

– протокол маршрутизации внутреннего шлюза (Interior Gateway Routing Protocol-IGRP) – дистанционно-векторный протокол маршрутизации, разработанный корпорацией Cisco;

– протокол выбора первого кратчайшего маршрута (Open Shortest Path First – OSPF) – протокол внутренней маршрутизации по состоянию канала;

– усовершенствованный протокол маршрутизации внутреннего шлюза (Enhanced Interior Gateway Routing Protocol-IGRP) – гибридный протокол маршрутизации, разработанный корпорацией Cisco;

– протокол граничного шлюза(Border Gateway Protocol-BGP) – протокол внешней маршрутизации.

6.1.16 Протокол RIP

Протокол маршрутной информации (Routing Information Protocol – RIP) был первоначально определен в документе RFC 1058 в 1988 г. Наиболее существенны его следующие характеристики:

– RIP является дистанционно-векторным протоколом маршрутизации;

– в качестве метрики при выборе маршрута используется количество переходов;

– если количество переходов становится больше 15, пакет отбрасывается;

– стандартно **обновления маршрутизации (routing updates)** рассылаются широковещательным способом каждые 30 с.

Протокол RIP с течением времени претерпел значительную эволюцию: от основанного на классах протокола маршрутизации RIP первой версии (RIP-1) к бесклассовому протоколу RIP второй версии (RIP-2). Усовершенствования протокола RIP-2 включают в себя:

– способность переносить дополнительную информацию о маршрутизации пакетов;

– механизм аутентификации для обеспечения безопасного обновления таблиц маршрутизации;

– способность поддерживать маски подсетей.

Протокол RIP предотвращает появление петель в маршрутизации, по которым пакеты могли бы циркулировать неопределенно долго, устанавливая максимально допустимое количество переходов на маршруте от отправителя к получателю. Стандартное максимальное значение количества переходов равно 15. При получении маршрутизатором обновления маршрутов, содержащего новую или измененную запись, он увеличивает значение метрики на единицу. Если при этом значение метрики превышает 15, то считается бесконечно большим, и сеть получателя считается недостижимой. Протокол RIP обладает рядом функций, которые являются общими для него и других протоколов маршрутизации.

6.1.17 Конфигурирование протокола RIP

Команда **router rip** включает RIP в качестве протокола маршрутизации. После этого выполняется команда network для указания протоколу сетей, которые непосредственно подсоединены к маршрутизатору и должны быть им анонсированы. Процесс маршрутизации после выполнения указанных двух действий логически связывает эти интерфейсы с сетевыми адресами и начинает использовать протокол RIP на интерфейсах маршрутизатора.

Как и большинство других протоколов, RIP рассылает регулярные сообщения об обновлении маршрутов, а реализация корпорации Cisco данного протокола использует мгновенные анонсы (их называют как event-triggered, так и event-driven) в тех случаях, когда изменяется топология сети. Изменения в топологии сети запускают рассылку обновлений также и в протоколе IGRP, вне зависимости от значения таймера удержания информации. Без такого механизма мгновенной рассылки обновлений как протокол RIP, так и IGRP не будут работать с максимальной эффективностью, поскольку мгновенные обновления значительно ускоряют конвергенцию таблиц маршрутизации и, следовательно, снижают риск образования петель маршрутизации.

При получении маршрутизатором сообщения об обновлении, содержащего изменения, он обновляет свою таблицу маршрутизации для отображения в ней нового маршрута. Значение метрики при этом увеличивается на единицу, а интерфейс отправителя обновления указывается в качестве следующего транзитного перехода на маршруте. Маршрутизаторы RIP вписывают только наилучший маршрут к пункту назначения, однако могут поддерживать и несколько маршрутов, если они имеют одинаковое значение метрики.

После обновления таблицы маршрутизации вследствие изменения топологии сети маршрутизатор сразу начинает рассылать сообщения об обновлении маршрутов, для того чтобы проинформировать другие маршрутизаторы о произошедших изменениях. Обновления рассылаются независимо от обычных регулярных сообщений RlP-маршрутизаторов. Если обновление пересылается через интерфейс другой суперсети с несовпадающим суммарным адресом, то протокол RIP анонсирует только сети, основанные на классах, или сети главного класса. Иными словами, информация о подсетях не сум-

мируется к одному агрегированному адресу и пересылается в виде отдельных записей, если анонс пересылается через интерфейс, адрес которого принадлежит той же суперсети. Классовые протоколы маршрутизации, например RIP версии 1, в анонсах маршрутизации не пересылают информацию о масках подсетей.

Для включения на маршрутизаторе протокола RIP используются команды режима глобального конфигурирования:

– Router(config)#router RIP // Включает процесс RIP-маршрутизации, после чего устройство переходит в режим конфигурирования;

– Router(config-router)#network // Связывает сеть с процессом RIP-маршрутизации.

Приведенные ниже команды иллюстрируют процесс включения на маршрутизаторе с именем BHM протокола RIP и указания ему непосредственно подсоединенных сетей:

– BHM(config)#router rip // включение протокола маршрутизации RIP;

– BHM(config-router)#network 1.0.0.0 // Указание непосредственно подключенной к устройству сети;

– BHM(config-router)#network 2.0.0.0 // Указание непосредственно подключенной к устройству сети.

Интерфейсы маршрутизатора Cisco, подсоединенные к сетям 1.0.0.0 и 2.0.0.0, рассылают и получают обновления протокола RIP. Эти обновления позволяют данному маршрутизатору изучить сетевую топологию с помощью соседних маршрутизаторов, на которых также включен протокол RIP.

В команде network протокола RIP можно указывать только классовые или суперсети. Если на одном или более интерфейсов маршрутизатора используются подсети такой сети, то для ее подключения можно использовать только одну команду network, в которой указан классовый адрес сети. Если же администратор попытается указать подсеть в данной команде, программное обеспечение Cisco 1OS автоматически преобразует такой адрес в адрес классовой сети, в чем можно убедиться с помощью команды «show running-config».

6.2 Задание к лабораторной работе

Вы администратор корпоративной локальной сети, состоящей из трех различных подсетей (рисунок 6.3):

192.168.0.0 /24

192.168.1.0 /24

192.168.2.0 /24

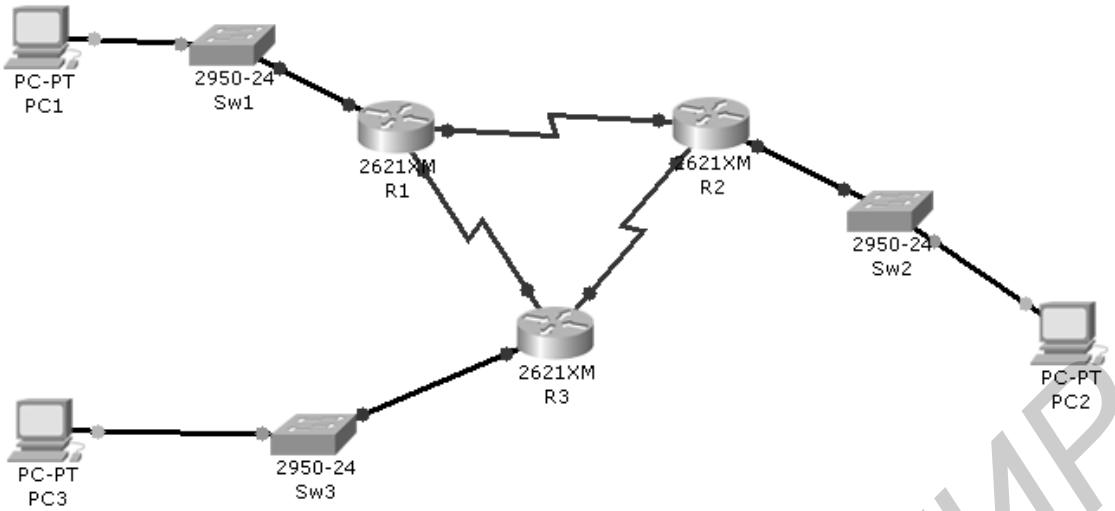


Рисунок 6.3 – Сеть для выполнения задания к лабораторной работе

Маршрутизация осуществляется с помощью соединенных между собой соединением Serial трех маршрутизаторов R1, R2 и R3.

1. Проведите настройку интерфейсов маршрутизаторов. Обратите внимание, что соединение Serial требует назначение со стороны DCE устройств параметра clock rate. Задайте clock rate равным 56 000.

R1:

Включите маршрутизатор

Сконфигурируйте интерфейс serial 1/0, используя ip address 192.168.4.1 /30.

Сконфигурируйте интерфейс serial 1/1, используя ip address 192.168.6.1 /30.

Сконфигурируйте интерфейс Fast Ethernet 0/0, используя ip address 192.168.0.1 /24.

R2:

Включите маршрутизатор

Сконфигурируйте интерфейс serial 1/0, используя ip address 192.168.6.2 /30.

Сконфигурируйте интерфейс serial 1/1, используя ip address 192.168.5.1 /30.

Сконфигурируйте интерфейс Fast Ethernet 0/0, используя ip address 192.168.1.1 /24.

R3:

Включите маршрутизатор

Сконфигурируйте интерфейс serial 1/0, используя ip address 192.168.4.2 /30.

Сконфигурируйте интерфейс serial 1/1, используя ip address 192.168.5.2 /30.

Сконфигурируйте интерфейс Fast Ethernet 0/0, используя ip address 192.168.2.1 /24.

2. Настройте проколол маршрутизации RIP. После обновления таблиц маршрутизации проверьте доступность одной подсети из другой.

Указание: при выполнении работы руководствуйтесь пунктом 1.1.4 лабораторной работы №1.

6.3 Содержание отчета

1 Цель работы.
2 Схема сети.
3 Таблицы конфигурации интерфейсов.
4 Таблицы статических маршрутов.
5 Выводы.

6.4 Контрольные вопросы

1 Что такое метрика?
2 Перечислите основные характеристики, используемые в метриках.
3 Что такое протокол маршрутизации и маршрутизируемый протокол, в чем их отличия?
4 Что такое автономная система и зачем она нужна?
5 Назовите категории (классы) протоколов маршрутизации.
6 Как происходит процесс обнаружения сетей при маршрутизации по состоянию канала?
7 Что такое протокол RIP?

Лабораторная работа №7

ПРОТОКОЛ IGRP

Цель работы: получить практические навыки в настройке протокола маршрутизации IGRP.

7.1 Теоретическая часть

7.1.1 Введение

Как и RIP, протокол маршрутизации внутреннего шлюза *(Interior Gateway Routing Protocol – IGRP)* является дистанционно-векторным протоколом маршрутизации. Однако в отличие от протокола RIP он не основан на стандартах, а является фирменным протоколом корпорации Cisco. Протокол IGRP прост в реализации, но вместе с тем является более развитым протоколом маршрутизации по сравнению с протоколом RIP и позволяет использовать большее количество параметров для определения наилучшего маршрута к пункту назначения.

7.1.2 Функции протокола IGRP

IGRP представляет собой дистанционно-векторный протокол внутреннего шлюза. Дистанционно-векторные протоколы маршрутизации определяют наилучший маршрут путем сравнения соответствующих числовых величин, отражающих длину маршрутов. Измерение такой длины называется построением **вектора расстояния (distance vector)**. Маршрутизаторы, использующие дистанционно-векторные протоколы, должны регулярно рассылать свои таблицы маршрутизации полностью или частично в сообщениях об обновлениях маршрутов всем соседним маршрутизаторам.

По мере того как информация маршрутизации будет распространяться по сети, маршрутизаторы могут, в частности, выполнять следующие функции:

– обнаруживать новые пункты назначения;

– обнаруживать ставшие недействительными маршруты.

Дистанционно-векторный протокол маршрутизации IGRP был разработан корпорацией Cisco. Этот протокол рассылает обновления маршрутизации с 90-секундными интервалами, анонсируя сети, принадлежащие конкретным автономным системам. Перечислим важнейшие характеристики протокола IGRP:

– содержит разнообразные функции, позволяющие работать со сложными и запутанными топологиями сетей;

– предоставляет высокий уровень гибкости, требуемый для работы с сегментами, имеющими различную ширину полосы пропускания и характеристики задержки;

– характерна высокая степень масштабируемости, позволяющая упростить работу в очень крупных сетях.

Стандартно в качестве метрики протокол маршрутизации IGRP использует ширину полосы пропускания и задержку. Кроме того, возможно иное конфигурирование протокола IGRP, при котором используется комбинация переменных параметров для вычисления сложной составной метрики.

В качестве параметров метрики могут выступать:

– ширина полосы пропускания;

– задержка;

– уровень загрузки канала;

– надежность канала.

7.1.3 Метрики протокола IGRP

С помощью команды show ip protocols отображаются параметры, фильтры и другая сетевая информация о протоколах маршрутизации, функционирующих в маршрутизаторе. Такая информация требуется для определения метрик К1-К5 и включает в себя максимальное количество переходов, а также используется для вычисления составной метрики протокола IGRP, которая вычисляется следующим образом:

метрика = [К1*полоса пропускания + К2*полоса пропускания/(256 – нагрузка) + К3*задержка]*[К5/(надежность + К4)].

Параметр метрики К1 представляет ширину полосы пропускания, а параметр К3 – задержку. Стандартно значения параметров метрик К1 и К3 принимаются равными единице, а параметры К2, К4 и К5 устанавливаются равными нулю.

Стандартными значениями весов являются К1 = К3 = *1* и К2 = К4 = = К5 = 0; в таком случае используется упрощенная формула расчета метрики протокола 1GRP, в которой множитель [К5/(кадежность + К4)] опущен. Композитная метрика рассчитывается по формуле

метрика = полоса пропускания + задержка.

Значения параметров метрики К в указанных формулах являются постоянными и могут быть заданы с помощью следующей команды режима конфигурирования маршрутизатора:

metric weights tos kl k2 k3 k4 k5

Для нахождения ширины полосы пропускания необходимо выбрать наименьшее ее значение среди всех выходных интерфейсов и разделить это значение на 10 000 000. (Полоса пропускания выражается в Кбит/с с коэффициентом 10 000 000). Для вычисления задержки необходимо сложить ее значения для всех выходных интерфейсов и разделить это значение на 10 (задержка выражается в десятках долей микросекунды). Следует помнить о том, что наилучшим считается маршрут с наименьшей метрикой.

При выборе маршрута к пункту назначения такая составная метрика дает более точную характеристику маршрута, чем метрика протокола RIP, учитывающая только количество переходов. Маршрут с наименьшей метрикой принимается в качестве наилучшего.

Метрики протокола IGRP включают в себя следующие компоненты:

– полосу пропускания (Bandwidth) – выбирается наибольшее значение ширины полосы пропускания на маршруте;

– задержку (Delay) – кумулятивную задержку на интерфейсах при прохождении пакетов по маршруту;

– надежность (Reliability) – описывает надежность канала, ведущего к пункту назначения; эта величина определяется в процессе обмена текстовыми сообщениями (keep alives);

– загрузку канала (Load), ведущего к пункту назначения; это значение выражается в битах в секунду.

Протокол IGRP использует составную метрику, которая вычисляется как функция полосы пропускания, задержки, загрузки и надежности канала. Стандартно в качестве параметров метрики используются только полоса пропускания и задержка, остальные параметры учитываются только в том случае, если их коэффициенты явно заданы в конфигурации. Значения задержки и полосы пропускания не измеряются в процессе работы устройством, а задаются в конфигурации командами **delay** и **bandwidth** определенного интерфейса. В примере 7.1 команда **show ip route** отображает в скобках значения метрик протокола IGRP. Первое значение представляет собой административное расстояние, а второе – вычисленное значение метрики. Канал с большей шириной полосы пропускания имеет меньшую метрику, аналогично, маршрут с наименьшей задержкой также имеет меньшую метрику.

Пример 7.1:

```
RouterAtt#show  ip  route
Codes:  C – connected, S – static, I – IGRP, R – RIP, M – mobile, B – BGP, D –
          EIGRP, EX – EIGRP external, O – OSPF, IA – OSPF inter area N1
          – OSPF NSSA external type 1, N2 – OSPF NSSA external type 2, El
          – OSPF external type 1, E2 – OSPF external type 2, E – EGP i – IS-
          IS, LI – TS-TS level-1, L2 – TS-TS level-2, ia – TS-TS
          inter area
          * – candidate default, V – per-user static route, o – ODR, P –
            periodic downloaded static route
Gateway of last resort is not set
C       192.168,1.0/24      is directly connected, FastEthernet0/0
C       192,168.2.0/24      is directly connected, Serial0/0
T       192.168.3.0/24      [100/30135] via 192.168.2.2, 00:00:30,Serial0/0
```

7.1.4 Маршруты протокола IGRP

Протокол IGRP анонсирует три типа маршрутов:

внутренний (Inferior route) – представляет собой маршрут между подсетями сети, подсоединенной к интерфейсу маршрутизатора. Если сеть,

подсоединенная к маршрутизатору, не имеет подсетей, то внутренние маршруты не анонсируются;

системный (System route) – представляет собой маршрут между сетями, находящимися в одной автономной системе. Программное обеспечение Cisco IOS создаст системные маршруты на основе интерфейсов непосредственно подсоединенных сетей и информации, полученной от других IGRP-маршрутизаторов или серверов доступа. Системные маршруты не содержат информацию о подсетях;

внешний (Exterior route) – представляет собой маршрут к сетям, находящимся вне рассматриваемой автономной системы, которые устанавливаются при поиске стандартного шлюза («шлюза последней надежды»). Программное обеспечение Cisco IOS выбирает стандартный шлюз из списка внешних маршрутов, предоставляемого протоколом IGRP. Такой стандартный шлюз (маршрутизатор) используется программным обеспечением в том случае, если не найден лучший маршрут и сеть получателя не является непосредственно подсоединенной сетью. Если автономная система имеет более одного соединения с внешней сетью, то разные маршрутизаторы могут выбрать в качестве стандартного шлюза различные внешние маршрутизаторы.

7.1.5 Функции поддержки устойчивости сети протокола IGRP

Протокол IGRP имеет ряд функций, предназначенных для повышения устойчивости работы сети:

– таймеры удержания информации (Holddown);

– механизм расщепления горизонта (Split horizon);

– удаление маршрута в обратном направлении (Poison reverse update).

Таймеры удержания информации используются для предотвращения рассылки обновлений маршрутизации, содержащих маршруты, которые в действительности неработоспособны. Если маршрутизатор выходит из строя, то соседние маршрутизаторы определяют такое его состояние по отсутствию регулярных сообщений об изменении маршрутизации.

Использование механизма расщепления горизонта основано на предположении, что обычно нецелесообразно посылать информацию о маршруте в том же направлении, по которому она была получена. Использование этого механизма помогает предотвратить появление петель в маршрутизации.

Расщепление горизонта предотвращает появление кольцевых маршрутов между смежными маршрутизаторами, однако, для предотвращения петель большей протяженности требуется использование другого механизма – удаления маршруте. Строго говоря, увеличение метрики маршрутизации обычно указывает на появление петель маршрутизации. Удаление маршрутов в обратном направлении происходит посредством рассылки уведомлений для отмены маршрута и перевода его в состояние удержании. В протоколе IGRP такие сообщения рассылаются только в том случае, если метрика маршрута увеличилась в 1,1 раза или более.

Протокол IGRP также поддерживает ряд таймеров и переменных, в которых содержатся временные интервалы, влияющие на работу механизма маршрутизации. Эти таймеры и их параметры описаны ниже.

Таймер обновления (Update timer) задает частоту, с которой рассылаются сообщения об обновлении маршрутизации. Его стандартное значение составляет 90 с.

Таймер действительности маршрута (Invalid timer) задает промежуток «времени ожидания», в течение которого маршрутизатор, не получая сообщения об обновлении по определенному маршруту, не рассылает информацию перед объявлением этого маршрута недействительным. В протоколе IGRP стандартно для этого параметра устанавливается значение в три раза больше, чем период регулярной рассылки анонсов маршрутов.

Таймер удержания информации (Hold timer) задает время, в течение которого информация о ненадежных маршрутах игнорируется. В протоколе IGRP стандартным значением для этого параметра принимается утроенное значение периода рассылки анонсов маршрутов, к которому добавляется 0 с.

Таймер сброса маршрута (Flush timer) задает время до того момента, когда маршрут будет удален. Стандартно значение этого параметра в семь раз больше периода рассылки анонсов маршрутизации.

В примере 7.2 приведена выводимая командой «show ip protocols» информация. Следует обратить внимание на строку, указывающую на функционирование протокола IGRP и значения его метрик.

Пример 7.2. Статистика маршрутизации протокола IGRP:

```
 RouterB#show ip protocols
Routing Protocol is "igrp 101"
Sending updates every 90 seconds, next due in 51 seconds
Invalid after 270 seconds, hold down 280, flushed after 630
Outgoing update filter list for all interfaces is Incoming update filter list for all interfaces is Default networks flagged in outgoing updates Default networks accepted from incoming updates
IGRP metric weight Kl=l, K2=0,
K3=1,                    K4=0,
IGRP maximum hopcount 100
IGRP maximum metric variance 1
Redistributing: igrp 101
Routing for Networks:
      192.168.2.0
      192.163.3.0
 Routing Information Sources:
Gateway Distance Last Update
192.168.2.1 100 00:00:54
Distance: (default is 100)
```

7.1.6 Конфигурирование протокола IGRP

Для конфигурирования процесса маршрутизации протокола IGRP используется команда глобального конфигурирования **router igrp:**

RouterA(config)#router igrp as-number

Для отключении процесса IGRP-маршрутизации используется форма этой команды с ключевым словом **no:**

RouterA(config)#no router igrp *as-number*

Под номером автономной системы понимается номер, идентифицирующий процесс маршрутизации протокола IGRP. Следует помнить, что такой номер не обязательно должен быть реальным номером автономной системы, которую присваивает соответствующая международная организация, например ARIN, или номер частной автономной системы. Такой номер действует только внутри домена маршрутизации протокола IGRP и должен быть одинаков на всех маршрутизаторах, которые должны обмениваться информацией по протоколу IGRP. Этот номер представляет собой просто идентификатор процесса. Он также используется для маркировки информации о маршрутизации.

Для задания списка сетей процессов IGRP-маршрутизации используется команда network режима конфигурирования маршрутизатора:

RouterA(config) #router igrp 101

RouterA(config-router)# network 192.168.1.0

Для удаления сети из списка используется форма этой команды с ключевым словом **no**, аналогично отключается сам процесс маршрутизации протокола IGRP:

RouterA(config)#no router igrp 101

RouterA(config-router)# no network

192.163.1.0

В примере 7.3 показана конфигурация протокола IGRP на маршрутизаторах *RouterA* и *RouterB*, принадлежащих автономной системе (Autonomous System-AS) с номером 101.

Пример 7.3 Конфигурирование протокола IGRP:

RouterA(config)# router igrp 101

RouterA(config-router)#network

RouterA(config-router)# network 192.168.2.0

RouterE(config)# router igrp 101

RouterB(config-router)# network 192.168.2.0

RouterE(config-router)# network

7.1.7 Проверка конфигурации протокола IGRP

Для проверки правильности конфигурации протокола IGRP необходимо использовать команду show ip route и проанализировать маршруты IGRP, отмеченные символом «i».

Дополнительно используются следующие команды проверки конфигурирования протокола IGRP:

– команда «show interface *interface*» позволяет проверить правильность конфигурирования Ethernet-интерфейса;

– команда «show running-config» указывает, включен ли в маршрутизаторе протокол IGRP;

– команда «show running-config interface *interface*» проверяет правильность конфигурации IP-адреса;

– команда «show running-config | begin interface *interface*» проверяет, включен ли протокол IGRP на интерфейсах маршрутизатора, начиная с указанного в команде интерфейса;

– команда «show running-config | begin igrp» проверяет, что в маршрутизаторе включен протокол IGRP;

– команда «show ip protocols» проверяет, что в маршрутизаторе функционирует протокол IGRP.

7.1.8 Поиск и устранение ошибок в конфигурации протокола IGRP

Большинство ошибок в конфигурации протокола IGRP связаны с неверными параметрами команд «network», с неверным указанием подсетей, которые не являются непрерывными, или неправильным указанием номеров автономных систем.

При поиске и устранении ошибок в конфигурации протокола IGRP используются следующие команды:

– команда «show ip protocols» используется для отображения общей информации протоколу IP-маршрутизации;

– команда «show ip route» используется для отображения таблицы IP-маршрутизации маршрутизатора;

– команда «debug ip igrp events» используется для отображения информации общего характера о маршрутизации для данной сети;

– команда «debug ip igrp transactions» отображает сообщения, полученные от соседних маршрутизаторов, на которых запрашивается обновление маршрутов, и широковещательные сообщения, посылаемые маршрутизатором-инициатором соседнему маршрутизатору;

– команда «ping» используется для определения доступности конкретного IP-адреса;

– команда «traceroute» используется для трассировки пути перемещения пакета от компьютера пользователя к узлу сети Internet; при этом выводится число требуемых переходов и время, затрачиваемое на такие переходы.

7.2 Задание к лабораторной работе

Вы администратор корпоративной локальной сети, состоящей из трех различных подсетей (см. рисунок 6.3 и задание к лабораторной работе №6).

1 Выполните настройку интерфейсов маршрутизаторов аналогично лабораторной работе №6.

2 Настройте протокол маршрутизации IGRP для автономной системы 101. После обновления таблиц маршрутизации проверьте доступность одной подсети из другой.

Указание: при выполнении работы руководствуйтесь пунктом 1.1.4 лабораторной работы №1.

7.3 Содержание отчета

1 Цель работы.
2 Схема топологии сети.
3 Таблицы настройки маршрутизаторов.
4 Выводы.

7.4 Контрольные вопросы

1 Назовите функции протокола IGRP.
2 Что входит в метрики протокола IGRP?
3 Охарактеризуйте типы маршрутов протокола IGRP.
4 Перечислите таймеры протокола IGRP.
5 Какие команды используются для настройки и поиска ошибок в рамках протокола IGRP?

Лабораторная работа №8

ПОСТРОЕНИЕ И НАСТРОЙКА ЛОКАЛЬНОЙ СЕТИ

Цель работы: получить начальные навыки построения и конфигурирования локальной сети.

8.1 Теоретическая часть

При проектировании локальных сетей исходными данными обычно являются размер, структура и род деятельности организации, где планируется развертывание сети, количество объединяемых в сеть компьютеров, характер решаемых задач, вопросы обеспечения безопасности, финансовые ресурсы, размеры сети, прогноз расширения сети и т. д.

8.1.1 Виды локальных сетей

8.1.1.1 Одноранговые сети

В одноранговой сети все компьютеры равноправны: нет иерархии среди компьютеров и нет выделенного (dedicated) сервера. Как правило, каждый компьютер функционирует и как клиент, и как сервер; иначе говоря, нет отдельного компьютера, ответственного за администрирование всей сети. Все пользователи самостоятельно решают, какие данные на своем компьютере сделать общедоступными по сети.

Одноранговые сети иногда называют также рабочими группами. Рабочая группа – это небольшой коллектив, поэтому в одноранговых сетях чаще всего не более 30-ти компьютеров. Одноранговые сети относительно просты. Поскольку каждый компьютер является одновременно и клиентом, и сервером, нет необходимости в мощном центральном сервере или в других компонентах, обязательных для более сложных сетей. Одноранговые сети обычно дешевле сетей на основе сервера, но требуют более мощных (и более дорогих) компьютеров, поскольку в одноранговой сети каждый компьютер функционирует и как клиент, и как сервер. В одноранговой сети требования к производительности и к уровню защиты для сетевого программного обеспечения, как правило, ниже, чем в сетях с выделенным сервером. Выделенные серверы функционируют исключительно в качестве серверов, но не клиентов или рабочих станций (workstation). В операционных системах Microsoft Windows NT Workstation, Microsoft Windows 9X, Microsoft Windows 2000/XP встроена поддержка одноранговых сетей. Поэтому при их использовании не требуется дополнительного программного обеспечения, чтобы установить одноранговую сеть.

Одноранговая есть характеризуется рядом стандартных решений:

– компьютеры расположены на рабочих столах пользователей;

– пользователи сами выступают в роли администраторов и обеспечивают защиту информации;

– для объединения компьютеров в сеть применяется простая кабельная система.

Применение одноранговой сети можно считать оправданным, если:

– количество пользователей не превышает 30 человек;

– пользователи расположены компактно;

– вопросы зашиты данных не критичны;

– в обозримом будущем не ожидается значительного расширения фирмы и, следовательно, сети;

– пользователи обладают достаточным уровнем знаний, чтобы работать и как пользователи, и как администраторы своего компьютера.

Если эти условия выполняются, то скорее всего выбор одноранговой сети будет правильным.

8.1.1.2 Сети на основе сервера

Если к сети подключено более 30-ти пользователей, то одноранговая сеть, где компьютеры выступают в роли и клиентов, и серверов, может оказаться недостаточно производительной. Поэтому большинство сетей использует выделенные серверы. Выделенным называется такой сервер, который функционирует только как сервер (исключая функции клиента или рабочей станции). Они специально оптимизированы для быстрой обработки запросов от сетевых клиентов и для управления защитой файлов и каталогов. Сети на основе сервера стали промышленным стандартом.

С увеличением размеров сети и объема сетевого трафика необходимо увеличивать количество серверов. Распределение задач среди нескольких серверов гарантирует, что каждая задача будет выполняться самым эффективным способом из всех возможных.

Круг задач, которые должны выполнять серверы, достаточно многообразен. Чтобы приспособиться к возрастающим потребностям пользователей, серверы в больших сетях стали специализированными (specialized). Например, в сети Windows 2003 Server существуют различные типы серверов.

К их числу относятся файл-серверы, принт-серверы, серверы приложений, почтовые серверы, факс-серверы, коммуникационные серверы, серверы каталогов, сетевые серверы и т. д.

Файл-серверы и принт-серверы управляют доступом пользователей, соответственно, к файлам и принтерам. Например, чтобы работать с текстовым процессором, вы прежде всего должны запустить его на своем компьютере. Документ текстового процессора, хранящийся на файл-сервере, загружается в память вашего компьютера, и, таким образом, вы можете работать с этим документом на своем компьютере. Другими словами, файл-сервер предназначен для хранения файлов и данных.

На серверах приложений выполняются серверные части клиент-серверных приложений, а также находятся данные, доступные клиентам. Например, чтобы упростить извлечение данных, серверы хранят большие объемы информации в структурированном виде. Эти серверы отличаются от файл- и принт-серверов. В последних файл или данные целиком копируются на запрашивающий компьютер. А в сервере приложений на запрашивающий компьютер пересылаются только результаты запроса. Приложение-клиент на удаленном компьютере получает доступ к данным, хранимым на сервере приложений. Однако вместо всей базы данных на клиентский компьютер с сервера загружаются только результаты запроса.

Почтовые серверы управляют передачей электронных сообщений между пользователями сети.

Факс-серверы управляют потоком входящих и исходящих факсимильных сообщений через один или несколько факс-модемов.

Коммуникационные серверы управляют потоком данных и почтовых сообщений между этой сетью и другими сетями, мэйнфреймами или удаленными пользователями через модем и телефонную линию.

Служба каталогов в Windows 2000/2003 (Active Directory) позволяет организовывать центральное управление всеми объектами сети, объединяя домены компьютеров, интергируясь с DNS (DNS – служба доменных имен, устанавливающая соответствие между доменными именами и IP-адресами), обеспечивая также систему защиты.

Сетевой сервер и сетевая операционная система (ОС) – неотъемлемые части структуры сети. ОС позволяет реализовать потенциал аппаратных ресурсов сервера. Некоторые сетевые ОС были созданы специально для того, чтобы использовать преимущества наиболее передовых серверных технологий.

В расширенной сети использование серверов разных типов приобретает особую актуальность. Необходимо поэтому учитывать все возможные нюансы, которые могут проявиться при разрастании сети, с тем чтобы изменение роли определенного сервера в дальнейшем не отразилось на работе всей сети.

8.1.1.3 Комбинированные сети

Существуют и комбинированные типы сетей, совмещающие лучшие качества одноранговых сетей и сетей на основе сервера. Такая сеть способна наиболее полно удовлетворить противоречивые запросы, т. к. в ней могут функционировать оба типа операционных систем.

Операционные системы для сетей на основе сервера, например Microsoft Windows NT/2000/2003 Server или Novell NetWare, в этом случае отвечают за совместное использование основных приложений и данных. На компьютерах-клиентах могут устанавливаться любые операционные системы семейства Microsoft Windows, которые будут управлять доступом к ресурсам выделенного

сервера и в то же время предоставлять в совместное использование свои жесткие диски, а по мере необходимости разрешать доступ и к своим данным. Комбинированные сети – наиболее распространенный тип сетей, но для их правильного построения и надежной защиты необходимы специальные знания и навыки планирования.

8.1.2 Построение простых сетей

Из двух вариантов сетей – одноранговые (децентрализованные) и на основе сервера – предпочтение лучше отдавать второму варианту, т. к. такие сети лучше масштабируются (одноранговые сети нецелесообразны при числе узлов более 30), легче структурируются, с ними проще решаются вопросы обеспечения безопасности и выхода в глобальные сети. Для большого количества приложений вполне подходят проводные сети Ethernet на неэкранированной витой паре 5-й категории (UTP 5, UTP 5E) по стандарту физического уровня 100-Base-TX. Данный стандарт предполагает звездообразную или древовидную (многоуровневую) топологию сети. Максимальная длина сегмента (отрезка кабеля) в такой сети – 100 м. При диаметре сети (максимальном расстоянии между станциями) до 200 м возможно построение сети на основе концентраторов (хабов), при этом сеть будет представлять собой единый домен коллизий (разделяемую среду, за право использования которой будут конкурировать все узлы). Такая конфигурация приемлема только для небольших сетей. В противном случае необходима структуризация сети (логическая и физическая) с применением коммутаторов или маршрутизаторов.

Построение локальной сети подразумевает следующие этапы.

1 Разработка локальной сети. В перечень основных работ в соответствии с этим этапом построения локальной сети предприятия входят первичное обследование территории, в рамках которой будет располагаться сеть, обсуждение с заказчиком основных задач, возложенных на сеть, подготовка технического задания и предварительный подбор необходимого оборудования.

2 Закупка оборудования, установка и монтаж локальной сети ЛВС. Прокладка кабеля, а также установка и настройка оборудования, программного обеспечения и систем защиты информации.

3 Тестирование сети. Проверка сети на ее безопасность, работоспособность и соответствие стандартам качества.

4 Гарантийное и послегарантийное обслуживание сети.

В качестве примера на рисунке 8.1 приведена модель сети небольшой организации, состоящей из четырех отделов. В каждом отделе имеется по 3 компьютера и один сетевой принтер. Топологии сетей отделов – звезда на основе 24-портовых коммутаторов Cisco 2950. Вся сеть имеет древовидную топологию на основе корневого коммутатора Cisco 2950 (в центре), к которому подключен сервер организации (Server0), на котором находится вебсайт организации, и маршрутизатор 1841 (Router0), который является шлюзом в Internet, где имеется удаленный сервер с именем www.inter.net, на кото-

ром установлена служба доменных имен DNS, а также размещен веб-сайт www.inter.net.

Обратите внимание на типы кабелей (прямые или кроссовые, которыми соединяются между собой узлы и коммутационное оборудование сети.

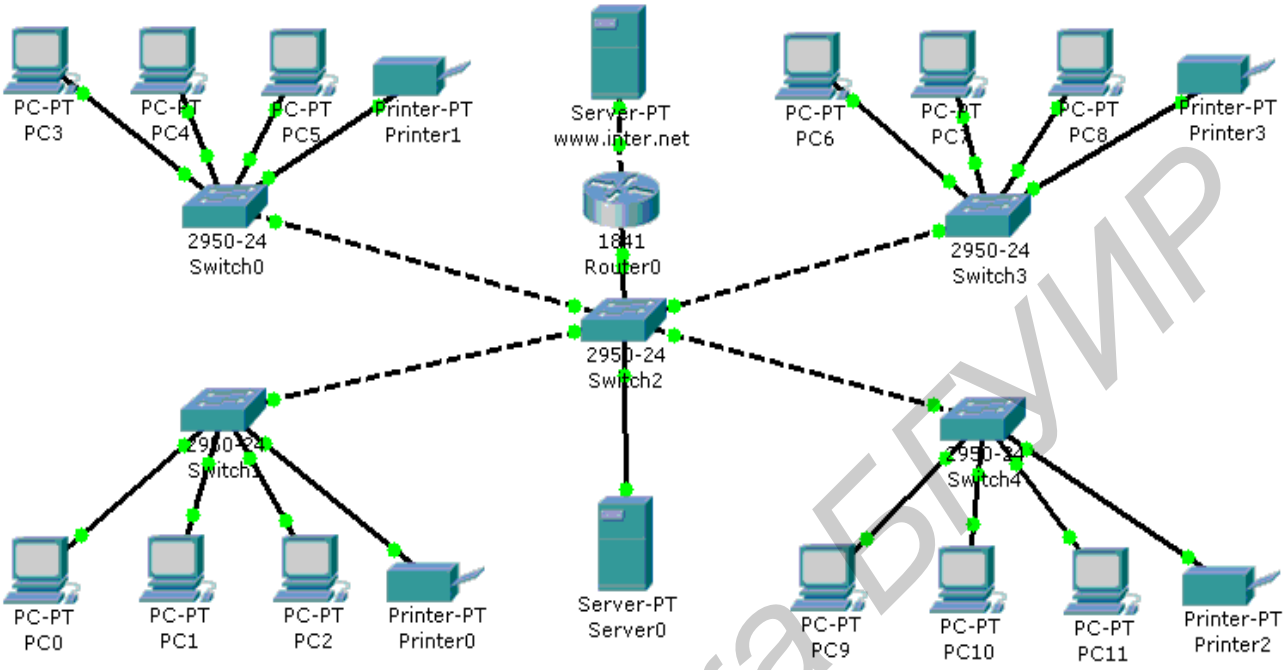


Рисунок 8.1 – Модель простой локальной сети с выходом в Internet

8.2 Задание к лабораторной работе

1 Получить у преподавателя исходные данные для построения сети;

2 Пользуясь навыками, полученными при выполнении лабораторных работ 1–7, собрать модель сети в Packet Tracer, сконфигурировать сетевые интерфейсы узлов и службы серверов для успешного прохождения теста связности сети. Протестировать всю сеть на предмет ее связности с помощью утилиты ping командной строки. Продемонстрировать связность сети преподавателю.

3 Модифицировать содержимое веб-страничек на сервере организации и интернет-сервере, сконфигурировать маршрутизатор (шлюз) и серверы таким образом, чтобы пользователь любого компьютера в сети смог открыть в браузере веб-сайт на интернет-сервере.

4 Подключить еще один компьютер непосредственно к интернет-серверу, сконфигурировать его так, чтобы пользователь этого компьютера смог просмотреть содержимое веб-сайта организации.

5 Продемонстрировать результаты преподавателю.

Указание: при выполнении работы руководствуйтесь пунктом 1.1.4 лабораторной работы №1.

8.3 Содержание отчета

1 Обоснование выбора топологии сети.
2 Схема сети.
3 Таблица IP-адресов и масок подсетей с указанием шлюзов (Gateways).
4 Таблица конфигурационных записей протоколов RIP и DNS.

8.4 Контрольные вопросы

1 Назовите известные вам виды локальных сетей, перечислите их преимущества и недостатки.
2 Какие операционные системы пригодны для построения одноранговых сетей?
3 Что такое выделенный сервер?
4 Перечислите известные вам типы специализированных серверов.
5 Какие этапы включает построение локальной сети?

ЛИТЕРАТУРА

1 Олифер, В. Г. Компьютерные сети. Принципы, технологии, протоколы / В. Г. Олифер, Н. А. Олифер. – 3-е изд. – СПб. : Питер, 2007. – 958 с.
2 Танненбаум, Э. Компьютерные сети / Э. Танненбаум. – 4-е изд. – СПб. : Питер, 2007. – 993 с.
3 CCNA 2: Routers and Routing Basics v.3.0. Student Lab Manual. – Cisco Systems Inc., 2003. – 300 с.

Св. план 2012, поз. 55

Учебное издание

Гурский Александр Леонидович
Певнева Наталья Алексеевна

ТЕЛЕКОММУНИКАЦИОННЫЕ И ИНФОРМАЦИОННЫЕ СИСТЕМЫ И СЕТИ. ЛАБОРАТОРНЫЙ ПРАКТИКУМ

УЧЕБНО-МЕТОДИЧЕСКОЕ ПОСОБИЕ

Редактор Н. В Гриневич
Корректор И. П Острикова
Компьютерная верстка Ю. Ч. Клочкевич

Подписано в печать 20.07.2012. Формат 60x84 1/16. Бумага офсетная. Гарнитура «Таймс».
Отпечатано на ризографе. Усл. печ. л. 5,81. Уч.-изд. л. 6,0. Тираж 100 экз. Заказ 165.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
ЛИ №02330/0494371 от 16.03.2009. ЛП №02330/0494175 от 03.04.2009.
220013, Минск, П. Бровки, 6