

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное
бюджетное образовательное учреждение
высшего образования
«КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ
ЭНЕРГЕТИЧЕСКИЙ УНИВЕРСИТЕТ»**

С.Ю. Ситников, Ю.К. Ситников, Э.А. Мухутдинов

**ИНФОРМАЦИОННЫЕ СИСТЕМЫ И СЕТИ.
ЧАСТЬ 1. ОСНОВЫ КОМПЬЮТЕРНЫХ СЕТЕЙ**

Лабораторный практикум

Казань 2017

УДК 004.7
ББК 32.81
С41

Рецензенты:

доктор физико-математических наук, профессор ФГАОУ ВО
«Казанский (Приволжский) федеральный университет» *О.Н. Шерстюков*;
кандидат технических наук, доцент ФГБОУ ВО
«Казанский государственный аграрный университет» *М.Г. Кузнецов*

Ситников С.Ю., Ситников Ю.К., Мухутдинов Э.А.

С41 Информационные системы и сети. Ч. 1. Основы компьютерных сетей: лабораторный практикум / С.Ю. Ситников, Ю.К. Ситников, Э.А. Мухутдинов. – Казань: Изд-во Казан. гос. энерг. ун-та, 2017. – 68 с. : ил.

Представлен цикл лабораторных работ, который знакомит студентов с компьютерными сетевыми технологиями, лежащими в основе функционирования локальных и глобальных сетей. Каждая отдельная лабораторная работа снабжена подробным теоретическим материалом, позволяющим выполнить практические задания.

Предназначен для студентов очной формы обучения по всем образовательным программам направления подготовки бакалавров 09.03.01 «Информатика и вычислительная техника», изучающих дисциплину «Сети и телекоммуникации». Может служить пособием также для студентов других направлений подготовки, изучающих компьютерные и сетевые технологии.

УДК 004.7
ББК 32.81

© Ситников С.Ю., 2017

© Казанский государственный энергетический университет, 2017

ВВЕДЕНИЕ

Лабораторный практикум «Информационные системы и сети. Часть 1. Основы компьютерных сетей» разработан в соответствии с Федеральным государственным образовательным стандартом для студентов очной формы обучения высших учебных заведений, обучающихся по направлению подготовки бакалавров 09.03.01 «Информатика и вычислительная техника» и изучающих дисциплину «Сети и коммуникации».

Лабораторный практикум нацелен на формирование и закрепление следующих профессиональных компетенций обучаемых:

- способности устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;
- способности разрабатывать технические задания на оснащение отделов, лабораторий, офисов компьютерным сетевым оборудованием;
- способности разрабатывать компоненты программно-аппаратных комплексов, используя современные инструментальные средства.

Данный цикл лабораторных работ помогает ознакомиться с основами сетевых технологий, составляющих фундамент локальных и глобальных сетей. Каждая работа снабжена обширным теоретическим материалом, который необходимо внимательно изучить. Контрольные вопросы позволяют закрепить материал и воспользоваться им при выполнении последующих лабораторных работ. Для выполнения работ требуется один из программных эмуляторов (GNS3/dynamips) или симуляторов (eNSP, Boson NetSim, Cisco Packet Tracer) компьютерных сетей.

Лабораторная работа № 1

ЗНАКОМСТВО С ПРОГРАММНОЙ СРЕДОЙ МОДЕЛИРОВАНИЯ КОМПЬЮТЕРНЫХ СЕТЕЙ

Цель работы: ознакомиться с программным симулятором компьютерной сети и получить базовые навыки работы в среде *Packet Tracer*; научиться проектировать простейшие компьютерные сети; ознакомиться с утилитой **ping**.

Теоретические сведения

Среда моделирования компьютерной сети *Packet Tracer* дает возможность проектировать сетевые топологии из широкого спектра маршрутизаторов и коммутаторов, рабочих станций и сетевых соединений технологий *Ethernet*, *Serial*, *ISDN*, *Frame Relay*. Пакет разработан компанией *Cisco* и поэтому ориентирован в первую очередь на моделирование топологий с использованием продуктов компании. Начиная с версии 6.3 *Packet Tracer* доступен для свободного скачивания на сайте компании.

Packet Tracer позволяет имитировать работу различных сетевых устройств: маршрутизаторов, коммутаторов, точек беспроводного доступа, персональных компьютеров, сетевых принтеров, *IP*-телефонов и т.д. Работа с интерактивным симулятором дает весьма правдоподобное ощущение настройки реальной сети, состоящей из десятков или даже сотен устройств. Настройки, в свою очередь, зависят от характера устройств: одни можно настроить с помощью команд операционной системы, другие – с использованием графического web-интерфейса, третьи – через командную строку операционной системы или графические меню.

Благодаря режиму визуализации пользователь может отследить перемещение данных по сети, появление и изменение параметров *IP*-пакетов при прохождении через сетевые устройства, их скорость и маршруты следования. Анализ событий, происходящих в сети, позволяет понять механизм ее работы и обнаружить неисправности.

Packet Tracer может быть использован не только как симулятор, но и как сетевое приложение для симулирования виртуальной сети через реальную сеть. Пользователи разных компьютеров, независимо от их местоположения, могут работать над одной сетевой топологией, производя ее настройку или устраняя проблемы. Эта функция многопользовательского режима *Packet Tracer* широко применяется для организации командной работы, а также для проведения игр и соревнований между удаленными участниками.

С помощью *Packet Tracer* можно симулировать построение не только логической, но и физической модели сети и, таким образом, получать навыки проектирования. Схему сети можно наложить на чертеж реально существующего здания или даже города и спроектировать всю его кабельную проводку, разместить устройства в тех или иных зданиях и помещениях с учетом физических ограничений, таких как длина и тип прокладываемого кабеля или радиус зоны покрытия беспроводной сети.

Симуляция, визуализация, многопользовательский режим и возможность проектирования делают *Packet Tracer* полезным инструментом для обучения сетевым технологиям.

Интерфейс программы

Рабочая область окна программы состоит из следующих элементов (рис. 1).

1. *Menu Bar* содержит меню *File, Edit, Options, View, Tools, Extensions, Help*. Меню позволяет выполнять сохранение, загрузку сетевых топологий, настройку симуляции, а также много других интересных функций.

2. *Main Tool Bar* содержит графические изображения ярлыков для быстрого доступа к некоторым, наиболее востребованным командам меню *File, Edit, View* и *Tools*, кнопки *Network Information* и *Help*.

3. *Common Tools Bar* обеспечивает доступ к инструментам программы: *Select, Move Layout, Place Note, Delete, Inspect, Resize Shape, Add Simple PDU* и *Add Complex PDU*.

4. *Logical/Physical Workspace and Navigation Bar* предназначен для переключения рабочей области: физической или логической, также позволяет перемещаться между различными уровнями вложенности.

5. *Workspace* – область, в которой происходит создание сети, проводятся наблюдения за симуляцией и просматривается информация и статистика.

6. *Realtime/Simulation Bar* – панель, с помощью закладок которой можно переключаться между режимами *Realtime* и *Simulation*. Она также содержит кнопки, относящиеся к *Power Cycle Devices*, кнопки *Play Control* и переключатель *Event List* в режиме *Simulation*.

7. *Network Component Box* – область, в которой выбираются устройства и связи для размещения на рабочем пространстве. Содержит область *Device-Type Selection* и область *Device-Specific Selection*.

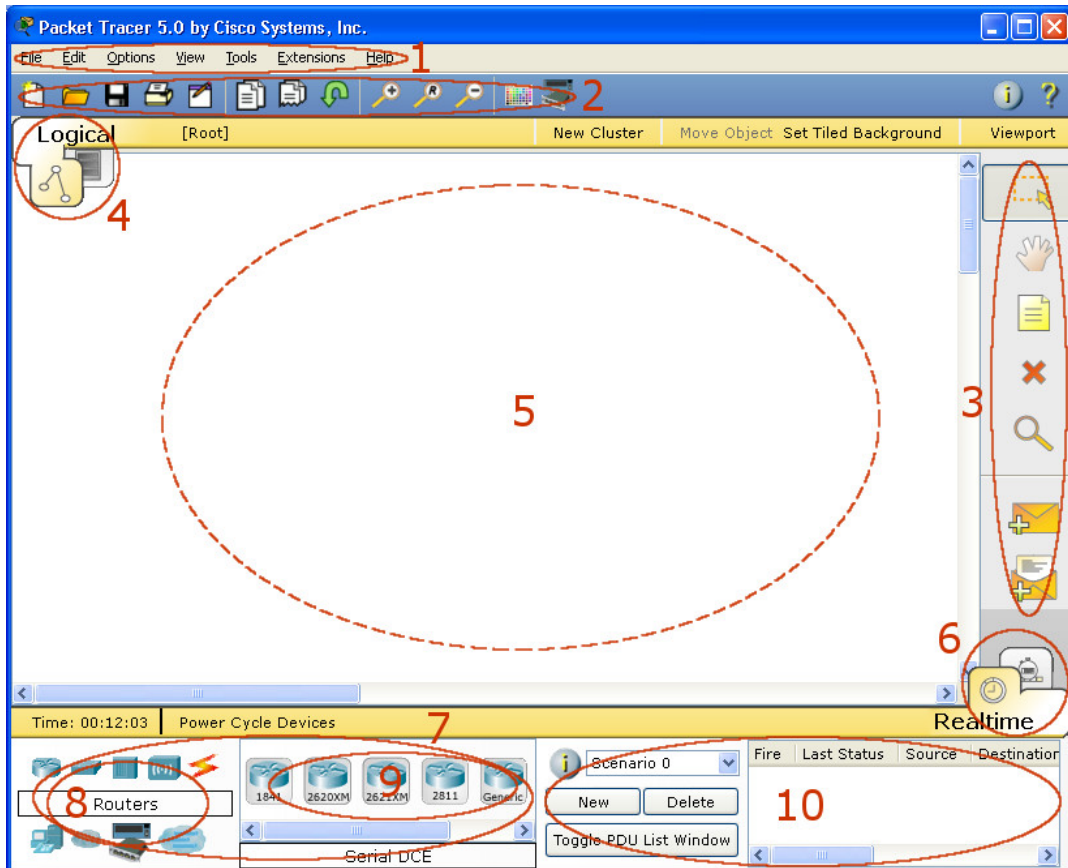


Рис. 1. Общий вид программы *Packet Tracer*

8. *Device-Type Selection Box* содержит доступные типы устройств и связей. При наведении курсором мыши на каждое из устройств в прямоугольнике, находящемся в центре между ними, отображается тип устройства. Типы, наиболее часто используемые в лабораторных работах, представлены на рис. 2.



Рис. 2. Типы устройств в *Packet Tracer*

9. *Device-Specific Selection Box* используется для выбора конкретных устройств и соединений, необходимых для постройки в рабочем пространстве сети. Она изменяется в зависимости от выбранного типа устройства в *Device-Type Selection Box*.

10. Окно *User Created Packet Window* управляет пакетами, которые созданы в сети во время симуляции сценария.

Для создания топологии сети необходимо выбрать вид устройства из панели *Network Component*, а затем из панели *Device-Type Selection* выбрать конкретный тип устройства и переместить устройство из области *Device-Type Selection* на рабочую область.

Для создания нескольких экземпляров устройств нужно, удерживая клавишу **Ctrl**, нажать на значок устройства в области *Device-Specific Selection* и отпустить клавишу **Ctrl**. После этого необходимо несколько раз щелкнуть мышью по рабочей области для добавления копий устройства.

В *Packet Tracer* представлены следующие типы устройств:

- роутеры (маршрутизаторы);
- мосты и свитчи (коммутаторы);
- хабы (концентраторы) и репитеры (повторители);
- конечные устройства – ПК, серверы, принтеры, *IP*-телефоны;
- беспроводные устройства: точки доступа и беспроводной маршрутизатор;
- облако, *DSL*-модем и кабельный модем.

Добавьте необходимые элементы в рабочую область программы, как показано на рис. 3.

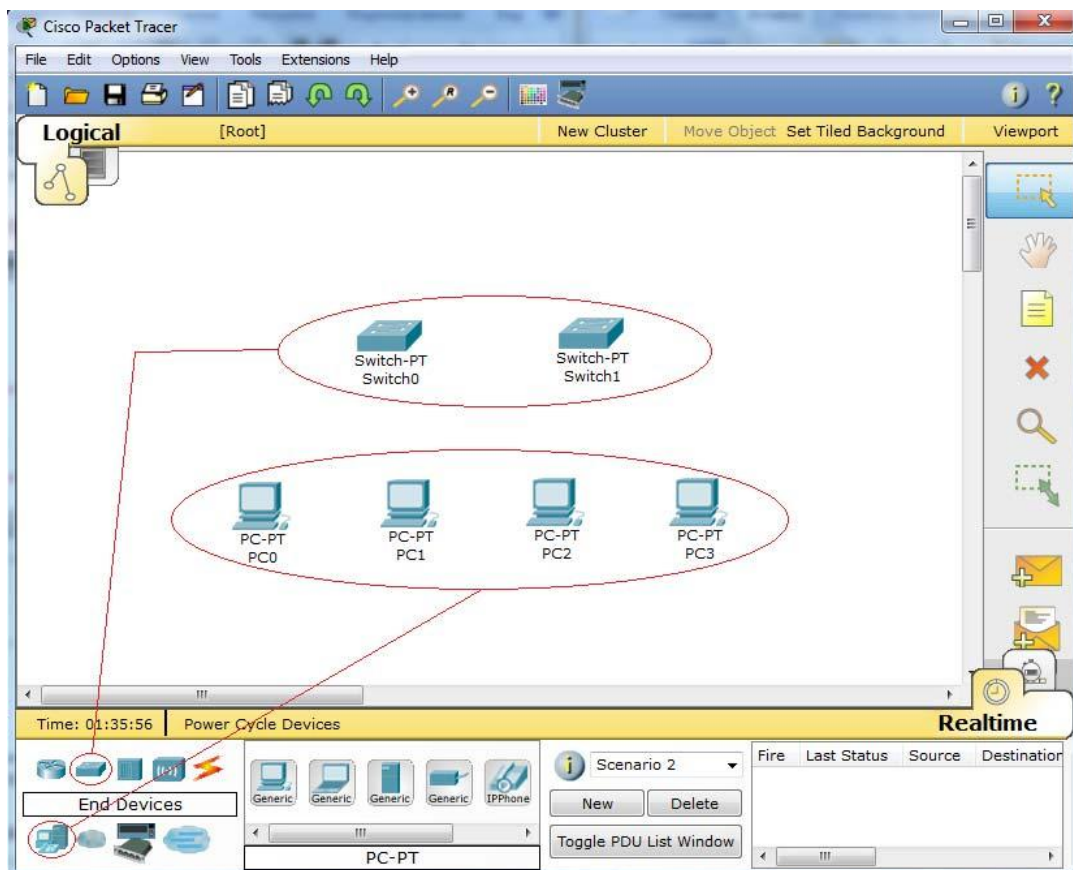


Рис. 3. Добавление элементов сети

При добавлении элемента вы имеете возможность дать ему имя и установить необходимые параметры. Для этого щелкните на устройстве для вызова диалогового окна и перейдите к вкладке *Config* (рис. 4).

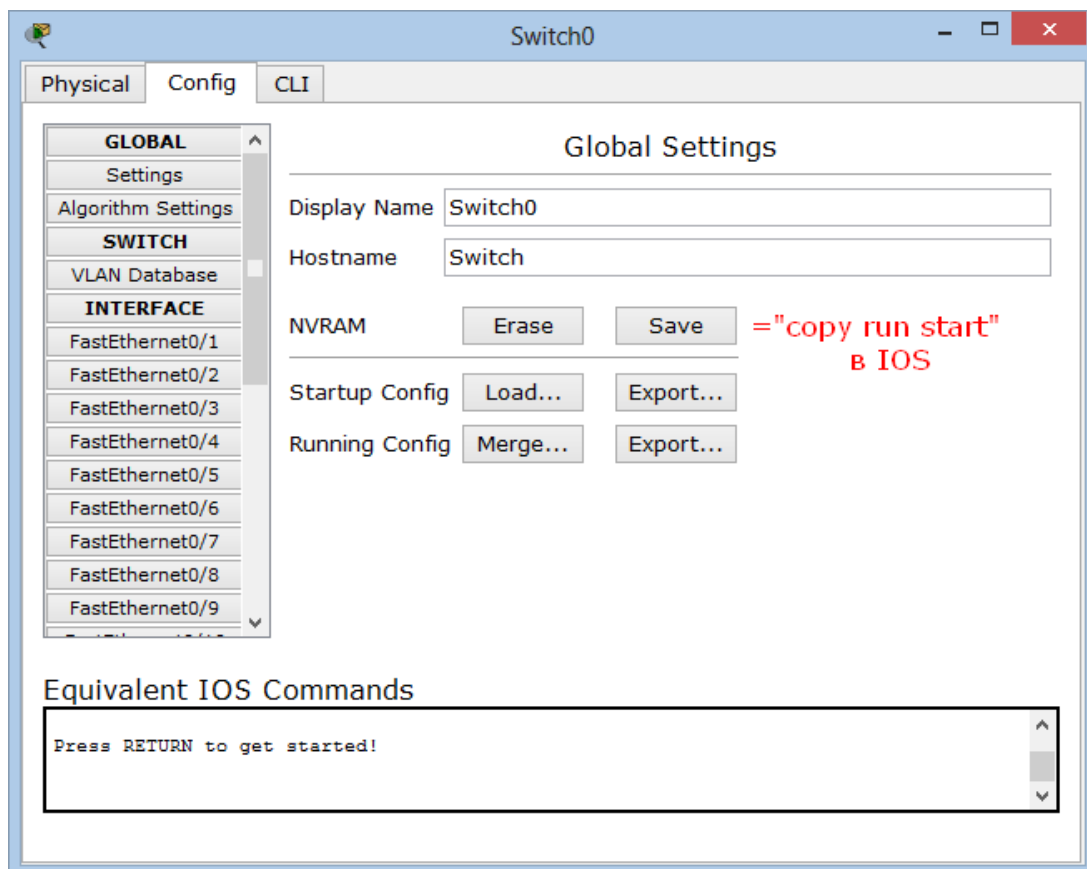


Рис. 4. Диалоговое окно устройства

Диалоговое окно свойств элемента имеет три вкладки:

- *Physical* содержит графический интерфейс устройства и позволяет симулировать работу с ним на физическом уровне;
- *Config* содержит необходимые параметры для настройки устройства.

Также, в зависимости от устройства, свойства могут иметь дополнительную вкладку для управления работой выбранного элемента: *Desktop* (если выбрано конечное устройство) или *CLI* (если выбран маршрутизатор) и т.д. *CLI* предоставляет доступ к ИКС – интерфейсу командной строки (*CLI* – *Command Line Interface*). Когда вы будете изменять параметры устройства с помощью мыши, в *CLI* автоматически появятся команды, соответствующие указанным параметрам. Вы в любой момент можете подробно ознакомиться с этими командами. Кроме того, здесь можно вручную ввести необходимые команды. Иногда это является единственным доступным способом корректно установить нужные значения.

Для удаления устройств с рабочей области программы используется кнопка *Delete* (**Del**).

Добавленные элементы можно связать с помощью соединительных связей. Для этого следует выбрать вкладку *Connections* из панели *Network Component Box* (рис. 5). Появятся возможные типы соединений между устройствами.

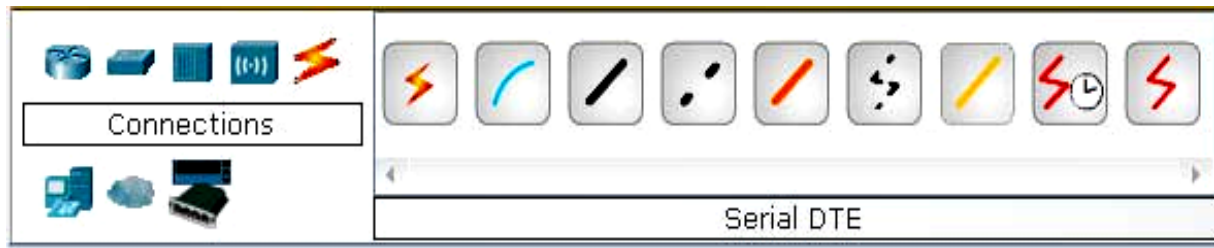


Рис. 5. Поддерживаемые типы кабелей

При указании мыши на конкретный тип кабеля курсор изменится на изображение разъема. Нажмите на первое устройство и выберите интерфейс, с которым нужно выполнить соединение, а затем нажмите на второе устройство и выполните ту же операцию. Можно соединить устройства автоматически с помощью инструмента *Automatically Choose Connection Type*. Выберите и нажмите на каждое из устройств, которые нужно соединить. Между ними появятся кабельные соединения, а индикаторы на каждом конце покажут статусы соединений для интерфейсов. Красный индикатор – порт отключен, зеленый – включен (рис. 6).

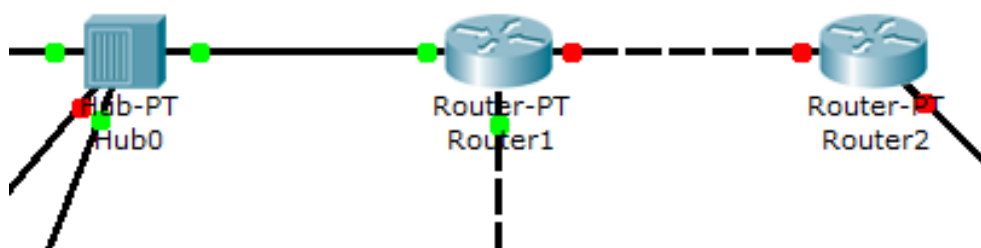


Рис. 6. Индикаторы состояния портов

Часто может оказаться удобным, если непосредственно на схеме отображается задействованный для соединения порт. Чтобы отобразить обозначение порта, выберите пункт главного меню *Options* → *Preferences* и отметьте опцию *Always Show Port Labels* (рис. 7).

Обратите внимание, что порты не обозначаются в случаях, когда это не имеет принципиального значения (рис. 8).

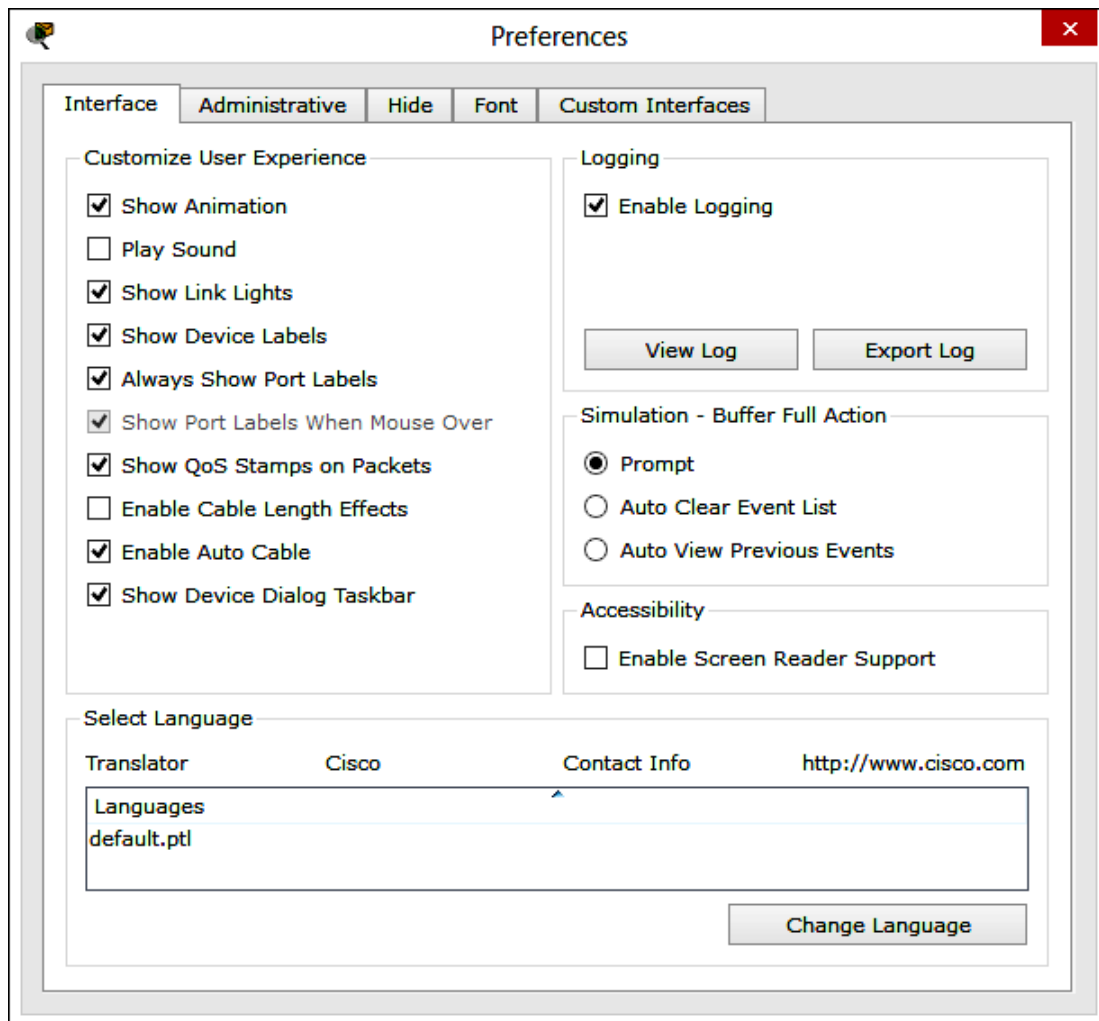


Рис. 7. Настройка параметров интерфейса среды моделирования

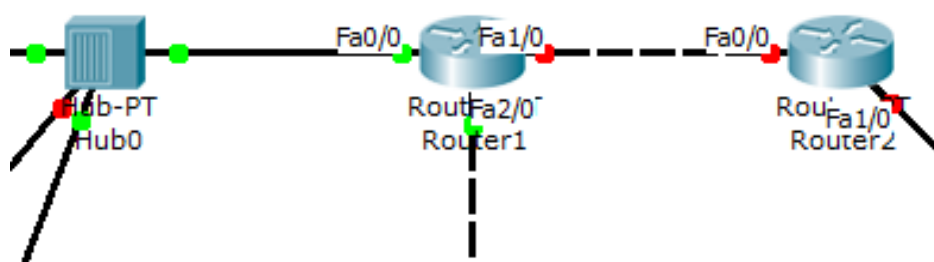










Рис. 8. Индикаторы состояния и обозначения портов

Среда моделирования поддерживает широкий диапазон сетевых соединений (табл. 1). Каждый тип кабеля может быть соединен с определенными типами интерфейсов.

После создания сети ее нужно сохранить, выбрав пункт меню *File* → *Save* или иконку *Save* на панели *Main Tool Bar*. Файл сохраненной топологии имеет тип **.pkt*.

Таблица 1

Типы соединений

Тип кабеля	Описание
 <i>Automatically Choose Connection</i>	Автоматическое соединение
 <i>Console</i>	Консольное соединение может быть выполнено между ПК и маршрутизаторами или коммутаторами. При этом должны быть выполнены требования работы консольного сеанса с ПК: скорость соединения с обеих сторон одинакова; количество бит данных равно 7 (или 8) для обеих сторон; контроль четности должен быть одинаковым, количество стоповых бит равно 1 или 2; поток данных может быть любым
 <i>Copper Straight-through</i>	Этот тип кабеля является стандартной средой передачи <i>Ethernet</i> для соединения устройств, которые функционируют на разных уровнях <i>OSI</i> . Он должен быть соединен с типами портов: медный 10 Мбит/с (<i>Ethernet</i>), медный 100 Мбит/с (<i>Fast Ethernet</i>) и медный 1000 Мбит/с (<i>Gigabit Ethernet</i>)
 <i>Copper Cross-over</i>	Этот тип кабеля является средой передачи <i>Ethernet</i> для соединения устройств, которые функционируют на одинаковых уровнях <i>OSI</i> . Он может быть соединен со следующими типами портов: медный 10 Мбит/с (<i>Ethernet</i>), медный 100 Мбит/с (<i>Fast Ethernet</i>) и медный 1000 Мбит/с (<i>Gigabit Ethernet</i>)
 <i>Fiber</i>	Оптоволоконная среда используется для соединения между оптическими портами (100 Мбит/с или 1000 Мбит/с)
 <i>Phone</i>	Соединение через телефонную линию может быть осуществлено только между устройствами, имеющими модемные порты. Стандартное представление модемного соединения – это конечное устройство (ПК), дозванивающееся в сетевое облако
 <i>Coaxial</i>	Коаксиальная среда используется для соединения между коаксиальными портами
 <i>Serial DCE, Serial DTE</i>	Соединения через последовательные порты используются для связей WAN. Для настройки соединений необходимо установить синхронизацию на стороне <i>DCE</i> -устройства. Синхронизация <i>DTE</i> выполняется по выбору. Сторону <i>DCE</i> можно определить по иконке часов рядом с портом. При выборе типа соединения <i>Serial DCE</i> первое устройство, к которому применяется соединение, становится <i>DCE</i> -устройством, а второе автоматически станет стороной <i>DTE</i> . Возможно и обратное расположение сторон, если выбран тип соединения <i>Serial DTE</i>

Packet Tracer дает возможность моделировать работу с ИКС операционной системы *IOS*, которая установлена на коммутаторах и маршрутизаторах компании *Cisco*. Следует отметить, что многие команды имеют свои аналоги в других сетевых операционных системах, поэтому их можно использовать в большинстве случаев точно так же, как и в среде системы *IOS*.

Подключившись к устройству, с ним можно работать так же, как за консолью реального устройства. Симулятор обеспечивает поддержку практически всех команд, доступных на реальных устройствах.

Подключение к ИКС коммутаторов или маршрутизаторов можно произвести, нажав на необходимое устройство и перейдя в окне свойств к вкладке *CLI*.

Для симуляции работы командной строки на конечном устройстве (компьютере) необходимо в свойствах выбрать вкладку *Desktop*, а затем нажать на ярлык *Command Prompt*.

Работа с файлами в симуляторе

Среда моделирования дает возможность хранить конфигурацию некоторых устройств, таких как роутеры или свитчи, в текстовых файлах. Для этого необходимо перейти к свойствам требуемого устройства и во вкладке *Config* нажать на кнопку *Export...* для экспорта конфигурации *Startup Config* или *Running Config*. Текст файла с конфигурацией устройства *running-config.txt* (имя по умолчанию) будет аналогичным по содержанию тексту, полученному при использовании команды **show running-config** в *IOS* устройства.

Необходимо отметить, что конфигурация каждого устройства сохраняется в отдельном текстовом файле. Пользователь имеет возможность изменять конфигурацию в сохраненном файле вручную с помощью произвольного текстового редактора. Для предоставления устройству сохраненных или отредактированных настроек нужно во вкладке *Config* нажать кнопку *Load...* для загрузки необходимой конфигурации *Startup Config* или кнопку *Merge...* для загрузки конфигурации *Running Config*.

Задание на лабораторную работу

На конкретном примере ознакомьтесь с программным симулятором компьютерной сети *Packet Tracer*, спроектируйте простую компьютерную сеть на основе коммутаторов и проверьте ее работу с помощью утилиты **ping**.

Работа выполняется по вариантам на отдельном компьютере.

Методические указания по выполнению лабораторной работы

1. Изучите теоретическую часть лабораторной работы.
2. Запустите на компьютере программный симулятор *Packet Tracer*.
3. Добавьте на рабочую область программы два коммутатора *Switch-PT*.
По умолчанию они имеют имена *Switch0* и *Switch1*.
4. Добавьте на рабочее поле четыре компьютера с именами по умолчанию *PC0*, *PC1*, *PC2*, *PC3*.
5. Соедините устройства в сеть *Ethernet*, как показано на рис. 9.

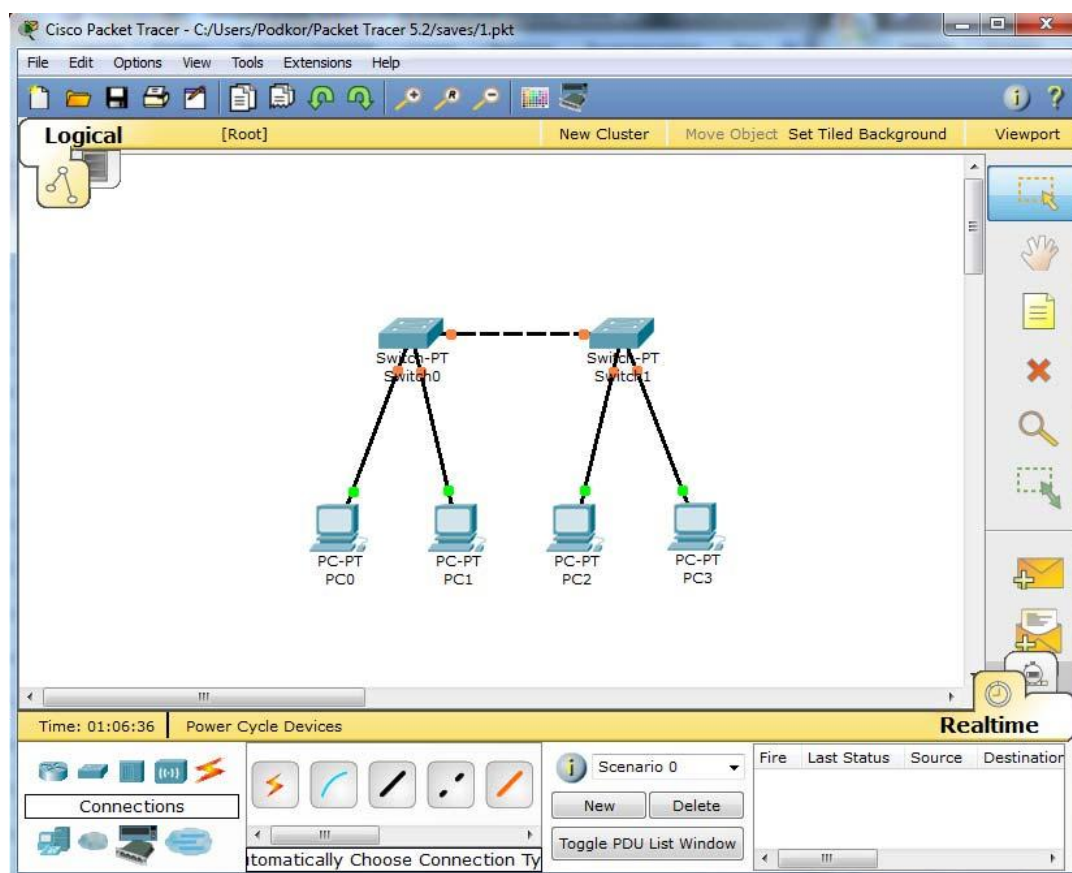


Рис. 9. Сеть с двумя свитчами и четырьмя компьютерами

6. Сохраните созданную топологию, нажав кнопку *Save* (в меню *File* → *Save*).
7. Щелчком левой кнопки мыши по значку устройства *PC0* откройте диалоговое окно устройства. Во вкладке *Desktop* выберите опцию *Command Prompt*. В результате откроется окно с интерфейсом командной строки.
8. Определите диапазон *IP*-адресов для своего варианта задания по шаблону $192.168.N.*$, где N – номер варианта (по списку в журнале). Далее в тексте приводится пример выполнения задания для варианта $N = 1$.

9. Если в окне командной строки *Command Prompt* поставить знак вопроса и нажать **Enter**, будет выведен список доступных команд. Для нашей задачи конфигурирования компьютера воспользуйтесь командой **ipconfig** из командной строки. Например, для задания *IP*-адреса, равного **192.168.1.2**, и маски подсети, равной **255.255.255.0**, следует выполнить команду:

ipconfig 192.168.1.2 255.255.255.0

Тот же результат, а именно задание *IP*-адреса и маски подсети, можно получить с помощью графического интерфейса устройства на вкладке *Config*. В данном графическом окне имеются также дополнительные вкладки окна. Нам следует выбрать вкладку **INTERFACE** (рис. 10).

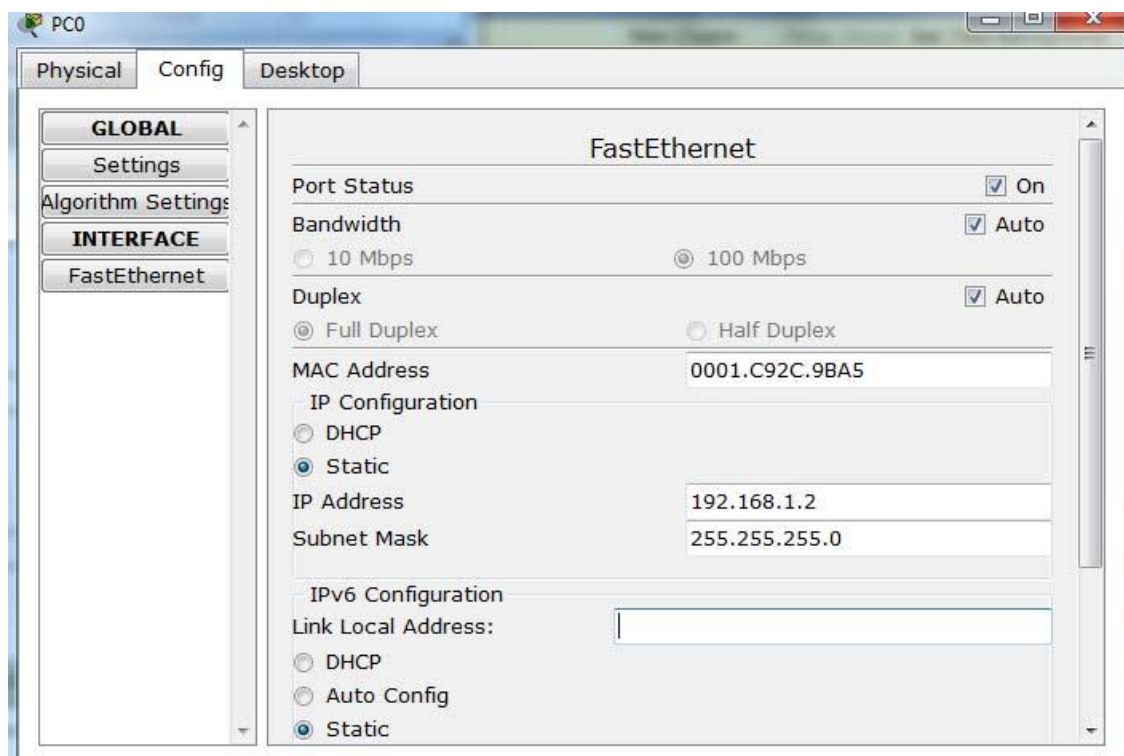


Рис. 10. Графический интерфейс устройства

Примечание. При выборе вкладки **GLOBAL/Settings** будут представлены параметры шлюза сети. В нашем случае поле *Gateway* адреса шлюза задавать не нужно, поскольку создаваемая сеть не требует маршрутизации.

Таким же образом настройте каждый компьютер сети согласно табл. 2.

10. Проверьте выполненные настройки каждого компьютера, перейдя к окну *Command Prompt* и задавая команду **ipconfig** (без параметров).

Таким образом, мы выполнили настройку компьютеров сети.

Таблица 2

Параметры настройки компьютеров в сети

Устройство	IP ADDRESS	SUBNET MASK
PC0	192.168.1.2	255.255.255.0
PC1	192.168.1.3	255.255.255.0
PC2	192.168.1.4	255.255.255.0
PC3	192.168.1.5	255.255.255.0

11. Вернитесь в главное окно симулятора *Packet Tracer*. На нижней панели окна в правом углу перейдите из режима настройки *Realtime* в режим моделирования *Simulation*, нажав соответствующий значок (либо комбинацию клавиш <Shift> + <S>).

В результате откроется окно *Simulation Panel* (рис. 11). В данном режиме подробно и графически показывается, как работает команда **ping** (а именно, будут отображаться все события, связанные с выполнением *ping*-процесса).

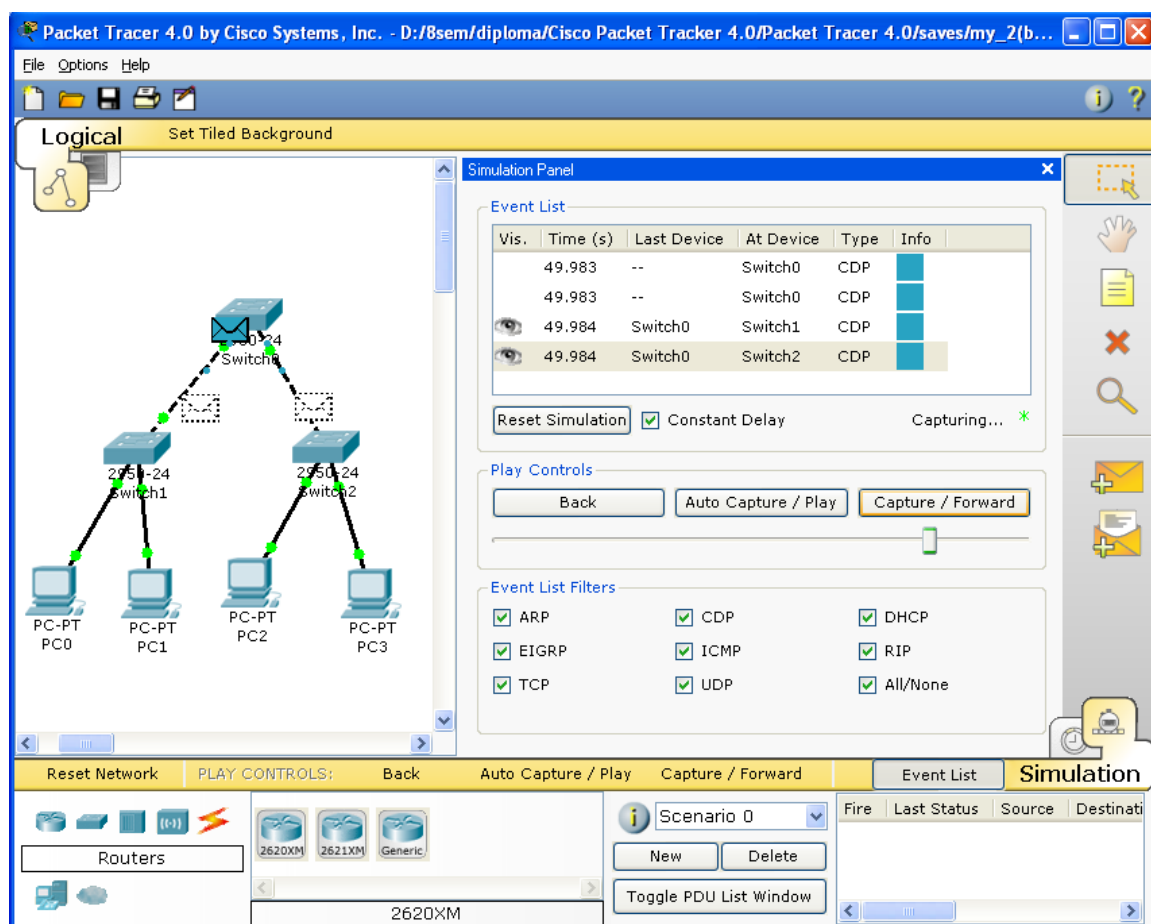


Рис. 11. Панель моделирования

12. Теперь необходимо запустить **ping**-процесс. После его запуска для удобства можно сдвинуть границы окна *Simulation Panel*, чтобы на схеме спроектированной сети наблюдать за процессами отправки и приемыки пакетов.

Кнопка *Auto Capture / Play* подразумевает моделирование всего **ping**-процесса в едином процессе, тогда как *Capture / Forward* позволяет отображать его в пошаговом режиме.

Чтобы узнать информацию, которую несет в себе пакет, его структуру, достаточно нажать правой кнопкой мыши на цветной квадратик, показанный в столбце *Info*.

13. В режиме моделирования можно не только отслеживать используемые протоколы, но и видеть, на каком из семи уровней модели *OSI* данный протокол задействован (рис. 12).

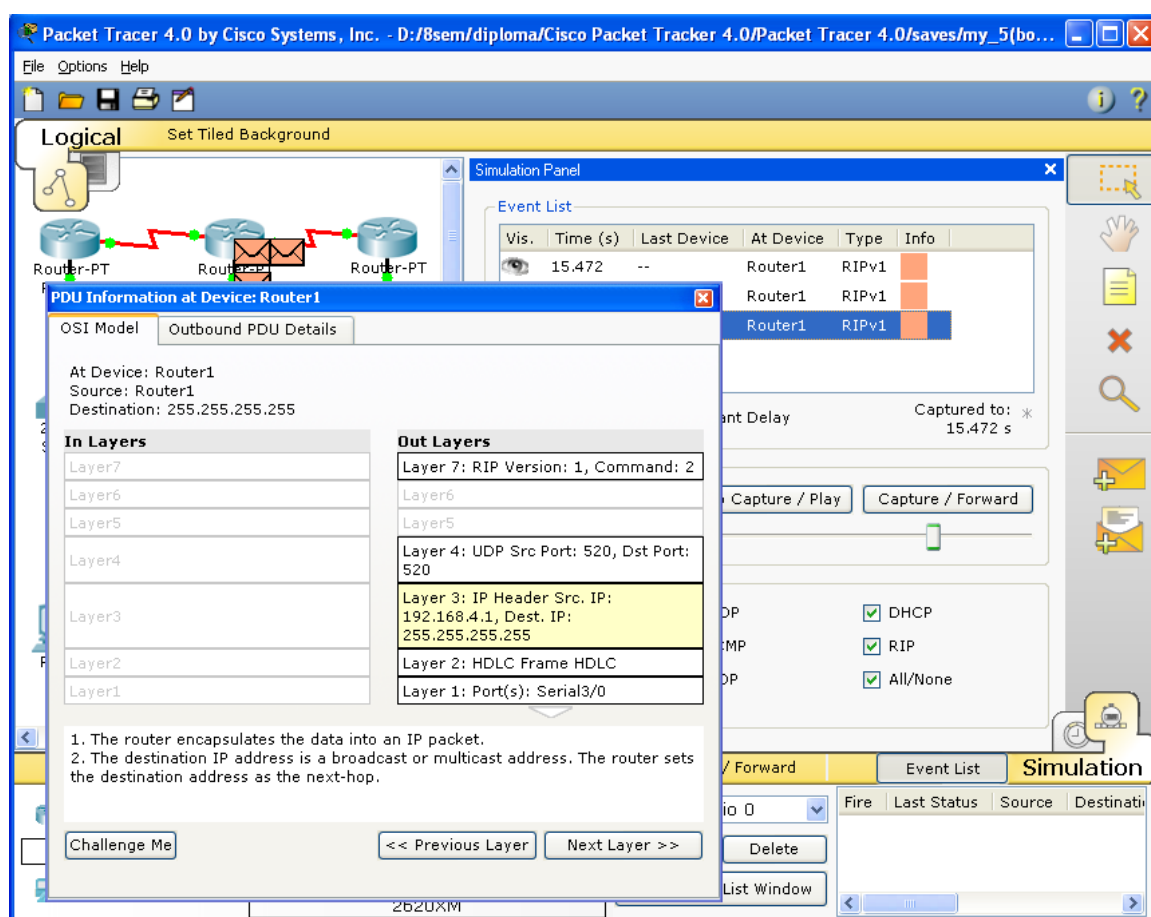


Рис. 12. Анализ семиуровневой модели *OSI*

Моделирование прекращается либо при завершении **ping**-процесса, либо при закрытии окна редактирования соответствующей рабочей станции.

14. Если все сделано правильно, вы сможете *пропинговать* узлы сети (проверить прохождение пакетов от выбранного компьютера на любой другой компьютер).

Для этого вернитесь в главное окно симулятора *Packet Tracer*. Выберите, например, компьютер *PC3*. Перейдите в окно настройки и выберите опцию *Command Prompt*. После открытия окна командной строки *Command Prompt* пропикуйте компьютер *PC0*. В результате выполненных действий вы должны увидеть отчет о пройденном пинге, подобный представленному на рис. 13.

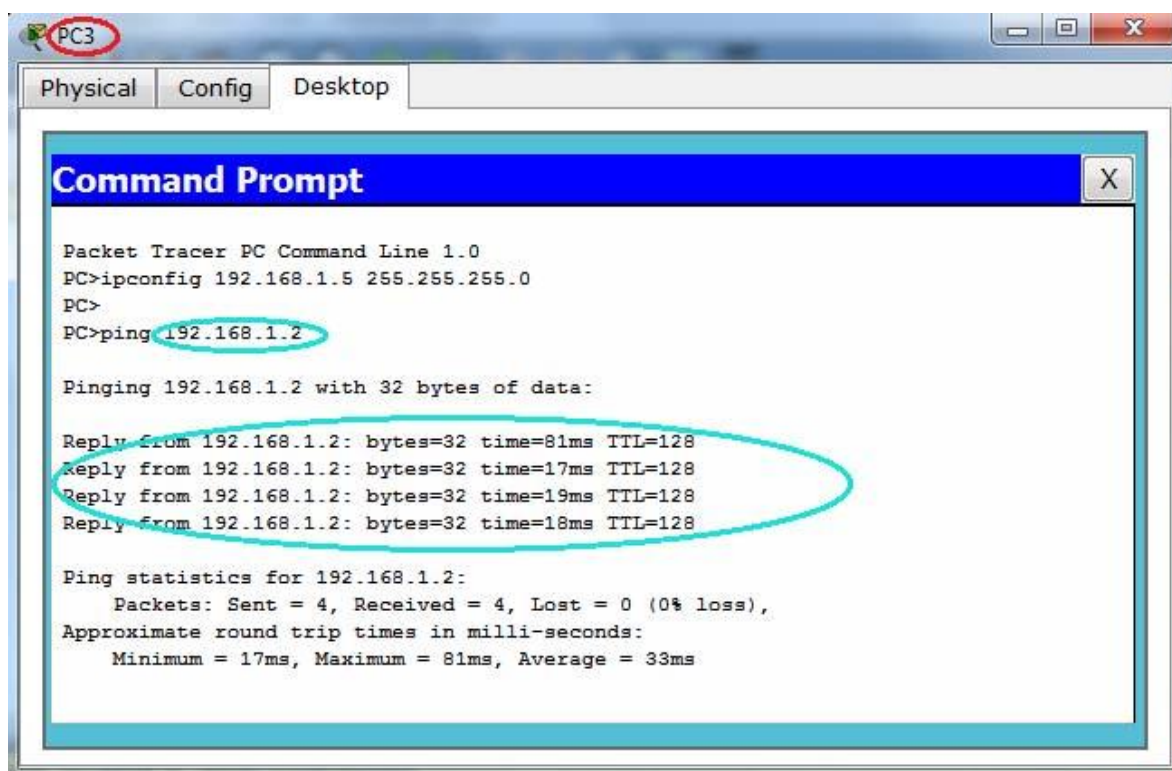


Рис. 13. Отчет о пинге

15. Составьте отчет о выполненной работе. Отчет должен содержать:

- 1) титульный лист с указанием названия лабораторной работы, фамилии студента, номера группы, варианта задания;
- 2) краткое изложение теоретических сведений по теме (2-3 страницы);
- 3) скриншоты экрана с результатами последовательно выполненных заданий (по своему варианту) и поясняющими комментариями к ним;
- 4) ответы на контрольные вопросы;
- 5) общие выводы по работе (заключение).

Контрольные вопросы

1. Приведите значение максимального количества устройств в сети, которое поддерживает программа *Packet Tracer*.
2. Перечислите типы сетевых устройств и соединений, которые можно использовать в *Packet Tracer*.

3. Опишите последовательность действий для перехода к интерфейсу командной строки устройства.
4. Приведите последовательность действий для конфигурирования сетевого устройства из другого компьютера.
5. Объясните, как добавить новое устройство в топологию сети и настроить его параметры.
6. Расскажите о том, как сохранить спроектированную конфигурацию устройства в текстовый файл.

Лабораторная работа № 2

АНАЛИЗ ФУНКЦИОНИРОВАНИЯ КОМПЬЮТЕРНОЙ СЕТИ

Цель работы: изучить основные функциональные возможности и режимы работы среды моделирования компьютерной сети; провести анализ работы простой сети при помощи команды **ping**.

Теоретические сведения

Некоторые сведения о симуляторе компьютерной сети

Симулятор *Packet Tracer* является интегрированной средой моделирования компьютерной сети. Он помогает создавать сетевые модели, осуществлять визуализацию и анимацию передачи информации в сети. Однако, как и любая среда моделирования, *Packet Tracer* опирается на упрощенные модели сетевых устройств и протоколов. Сетевые протоколы, реализованные в *Packet Tracer*, а также другие параметры программы приведены в табл. 3.

Таблица 3

Возможности программы *Packet Tracer*

Наименование	Описание
Протоколы	<p>LAN: Ethernet (including CSMA/CD), 802.11 a/b/g/n wireless, PPPOE</p> <p>Switching: VLANs, 802.1q, trunking, VTP, DTP, STP*, RSTP, multilayer switching, Etherchannel, LACP, PAgP</p> <p>TCP/IP: HTTP, HTTPS, DHCP, DHCPv6, Telnet, SSH, TFTP, DNS, TCP*, UDP, IPv4, IPv6, ICMP, ICMPv6, ARP, IPv6 ND, FTP, SMTP, POP3, VOIP(H.323)</p> <p>Routing: static, default, RIPv1, RIPv2, EIGRP, single-area OSPF, multi-area OSPF, BGP, inter-VLAN routing, redistribution</p> <p>Other: ACLs (standard, extended, and named), CDP, NAT (static, dynamic, inside/outside, and overload), NATv6</p> <p>WAN: HDLC, SLARP, PPP, Frame Relay</p> <p>Security: IPsec, GRE, ISAKMP, NTP, AAA, RADIUS, TACACS, SNMP, SSH, SYSLOG, CBAC, Zone-based policy firewall, IPS</p> <p>QoS: Layer 2 QoS, Layer 3 Diffserv QoS, FIFO Hardware queues, Priority Queuing, Custom Queuing, Weighted Fair Queuing, MQC, NBAR</p>

Наименование	Описание
Логическое пространство	Создание сетевой топологии <i>Доступные устройства:</i> маршрутизаторы, коммутаторы, хранилища (Server, Desktop and Laptop), хабы, мосты, беспроводные точки доступа, беспроводные маршрутизаторы и DSL/cable модемы Соединение устройств осуществляется с использованием медных, оптоволоконных, коаксиальных кабелей
Физическое пространство	Поддерживает следующие виды: иерархия устройств, коммутационные шкафы, здания, города. Также поддерживается отображение допустимой длины кабелей в сети Ethernet, масштабирование созданных пользователем графиков
Режим реального времени	Обмен данными происходит в режиме реального времени <i>Настраиваемая конфигурация:</i> DHCP, DNS, HTTP, TFTP, Syslog, AAA, and NTP servers
Режим симуляции	Анимация передачи пакетов, лист событий, широкий выбор протоколов модели OSI. Пользователь имеет возможность создания сценария передачи пакетов

Реальные компьютерные сети остаются эталоном для понимания поведения сети и развития навыков для их построения.

Логическое пространство

Для того чтобы расположить устройство, необходимо выбрать его из меню и перетащить на главную панель (рис. 14).

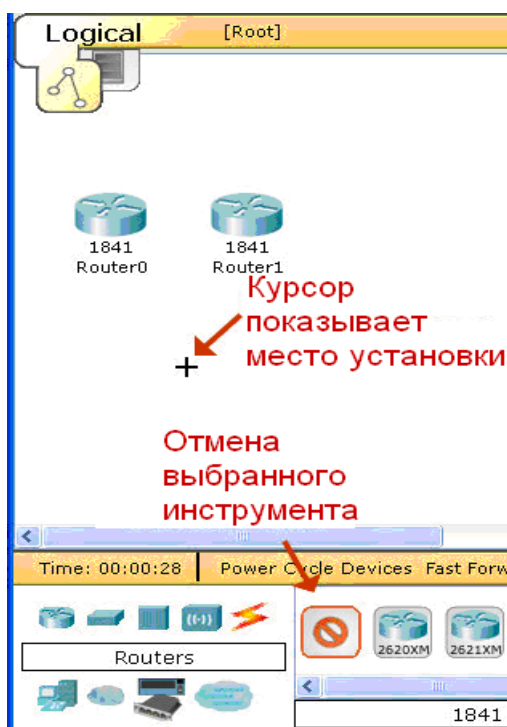


Рис. 14. Выбор устройства

Большинство из устройств в среде моделирования имеют модули расширения, необходимые для подключения дополнительных портов. Добавление модулей осуществляется в панели настройки устройства (рис. 15). При подключении нового модуля устройство должно быть отключено от электросети.

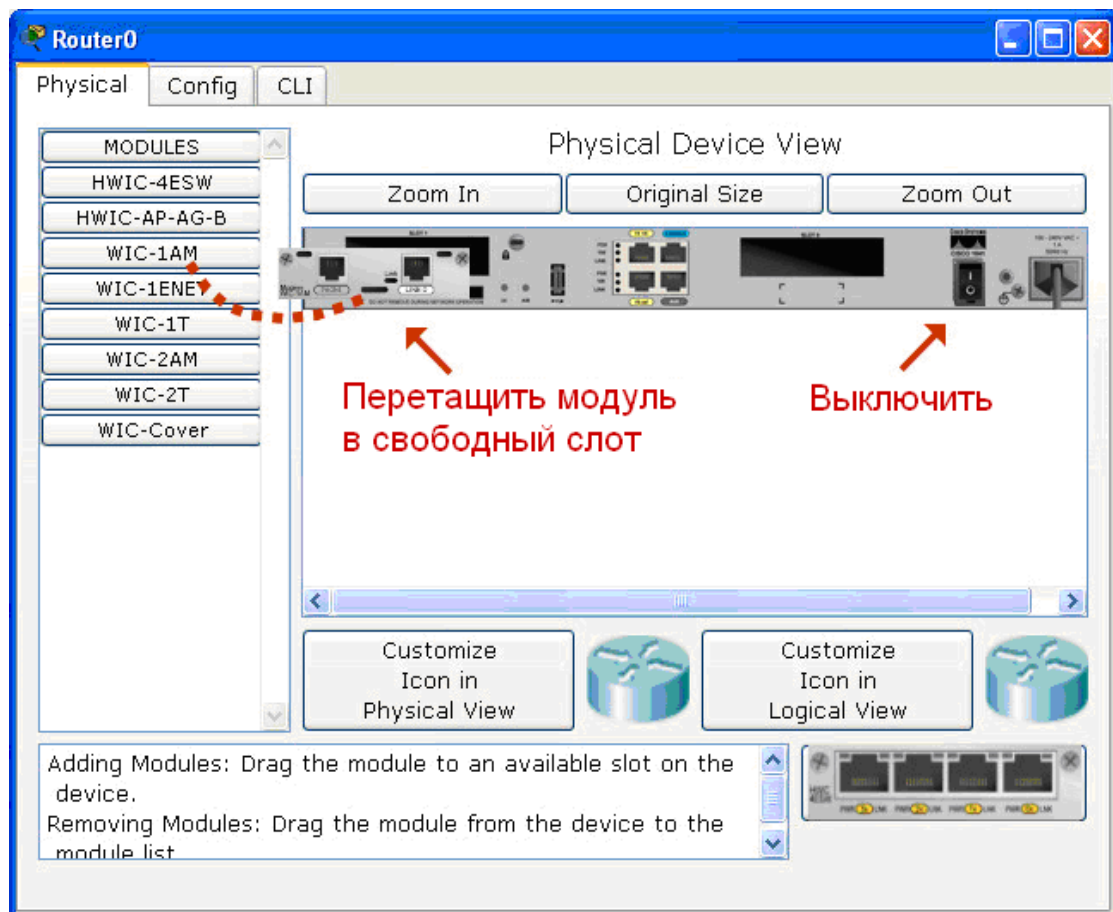


Рис. 15. Добавление модулей расширения

Packet Tracer предоставляет возможность создания шаблонов устройств (рис. 16). Для создания шаблона необходимо выбрать устройство, добавить требующиеся модули расширения, затем перейти в окно *Custom Devices Dialog*. Далее следует вставить описание выбранного устройства, нажав на *Select*. Добавить новое созданное пользователем устройство можно также через *Custom Made Devices*.

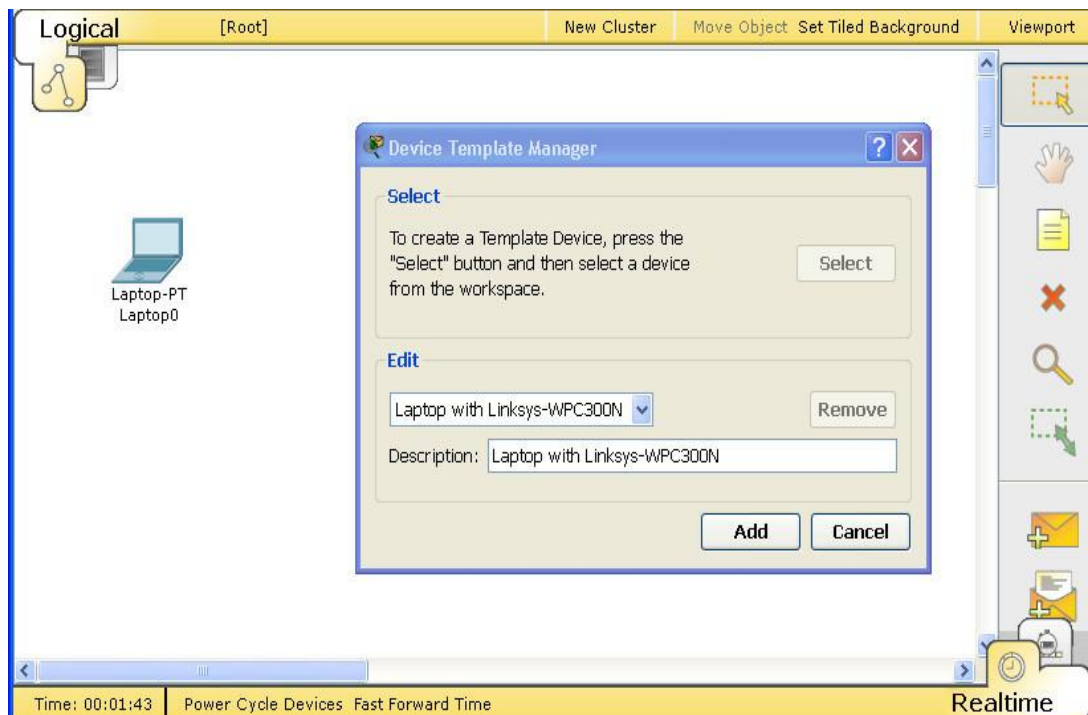


Рис. 16. Создание шаблона устройства

Для соединения устройств между собой необходимо выбрать подходящие кабели, расположенные на панели *Connections*, затем щелкнуть правой кнопкой мыши по одному из устройств и выбрать порт подключения (рис. 17). Аналогичные действия следует выполнить для второго устройства.

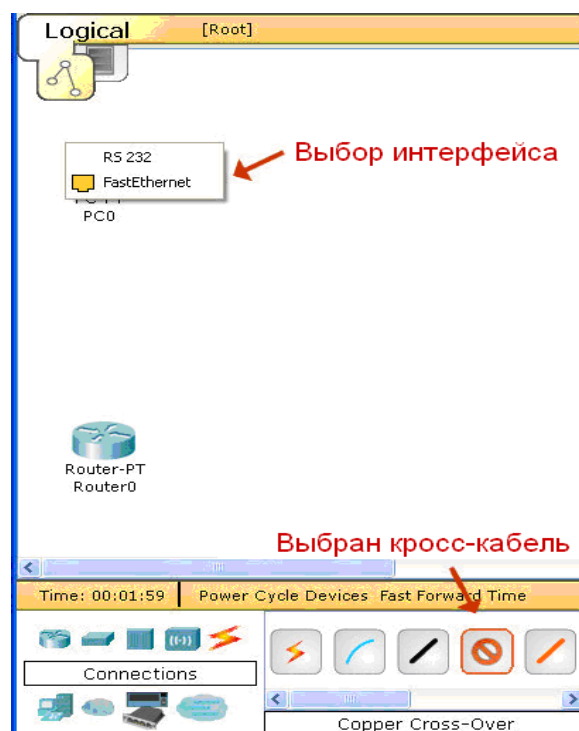


Рис. 17. Создание соединений

Режим реального времени

В режиме реального времени (*Realtime*) сеть всегда работает независимо от действий пользователя (рис. 18). Конфигурирование сети осуществляется в режиме реального времени *Realtime*. При просмотре статистики сети данные отображаются также в режиме реального времени.

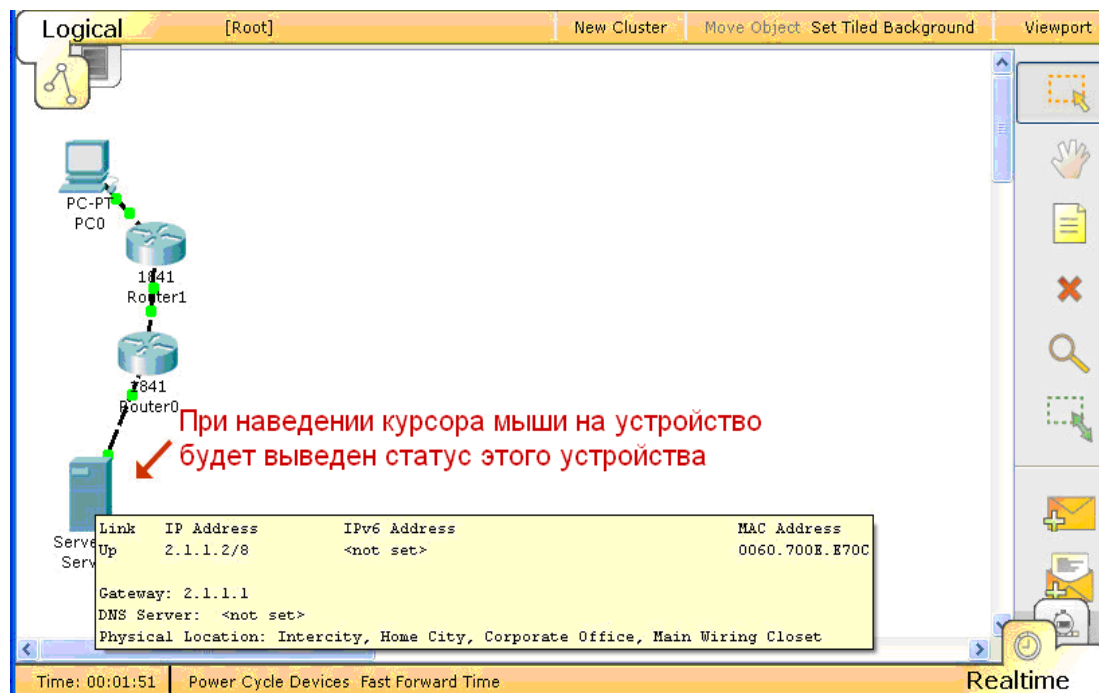


Рис. 18. Функционирование сети в режиме реального времени

Режим симуляции

В режиме симуляции (*Simulation*) можно изучать работу сети в более медленном темпе, исследуя пути, по которым пересылаются пакеты. При переключении в режим моделирования (симуляции) появится специальная панель (рис. 19). Можно графически просматривать распространение пакетов по сети, если нажать на кнопку *Add Simple PDU*. Имеется возможность контроля скорости моделирования с использованием кнопки *Speed Slider*. Также можно просматривать предыдущие события, нажав на кнопку *Back*.

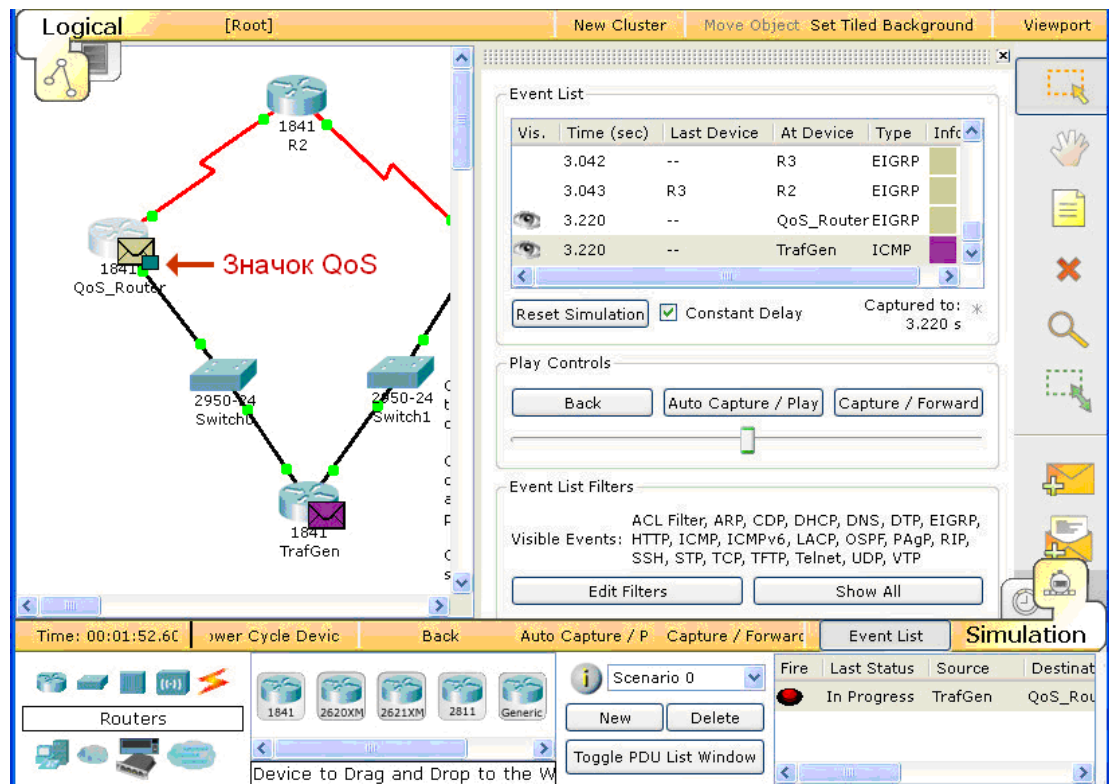


Рис. 19. Режим моделирования

Во время моделирования можно кликнуть vsim. на пересылаемом пакете и получить о нем подробную информацию (рис. 20).

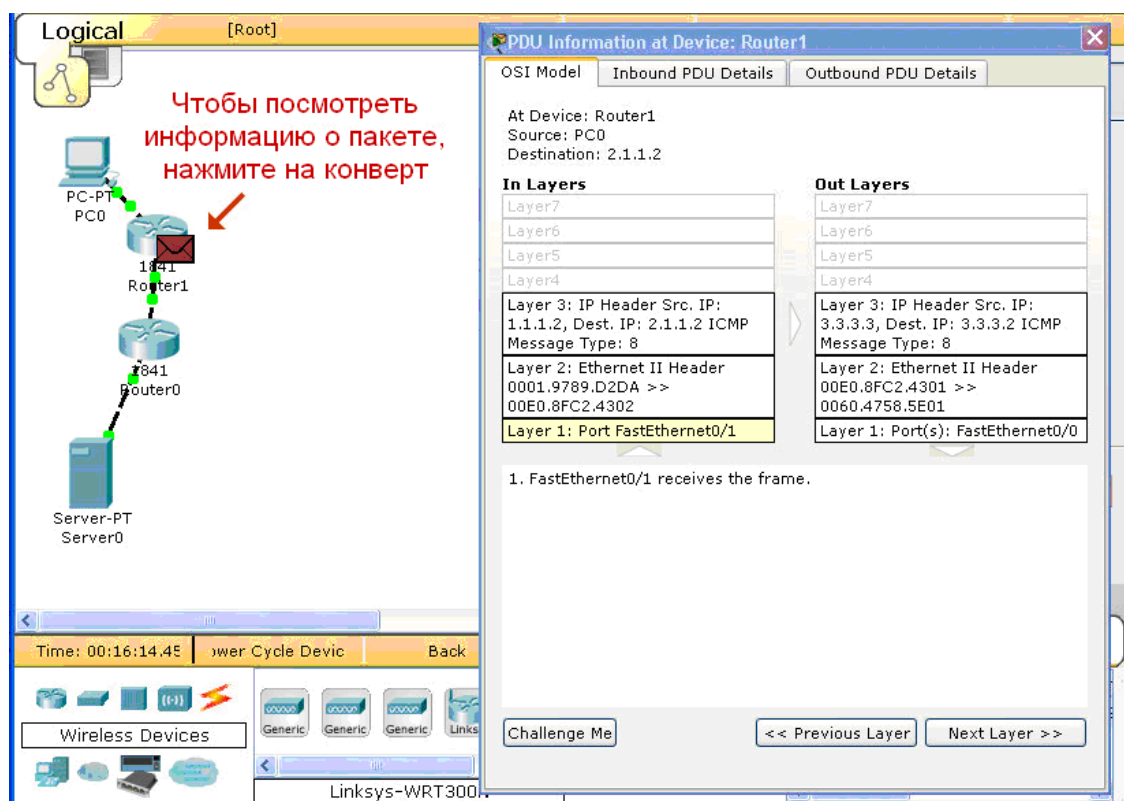


Рис. 20. Информация о пакете

Физическое пространство

Физическое пространство – это графическое отображение, которое служит для наложения абстрактной топологии сети на реальную карту объектов (помещений, зданий, городских кварталов). То есть целью физической рабочей области является обзор физических аспектов логической топологии сети. Это дает ощущение масштаба и размещения (как именно сеть может выглядеть в реальной среде).

Физически рабочая область разделена на четыре слоя, чтобы отразить масштаб: междугородный, город (рис. 21), дом или строение (рис. 22), коммутационный узел (рис. 23 и 24). Междугородный слой является по масштабу крупнейшим, он может содержать много городов. Каждый город может содержать множество зданий. Наконец, каждое здание может содержать множество коммутационных узлов.

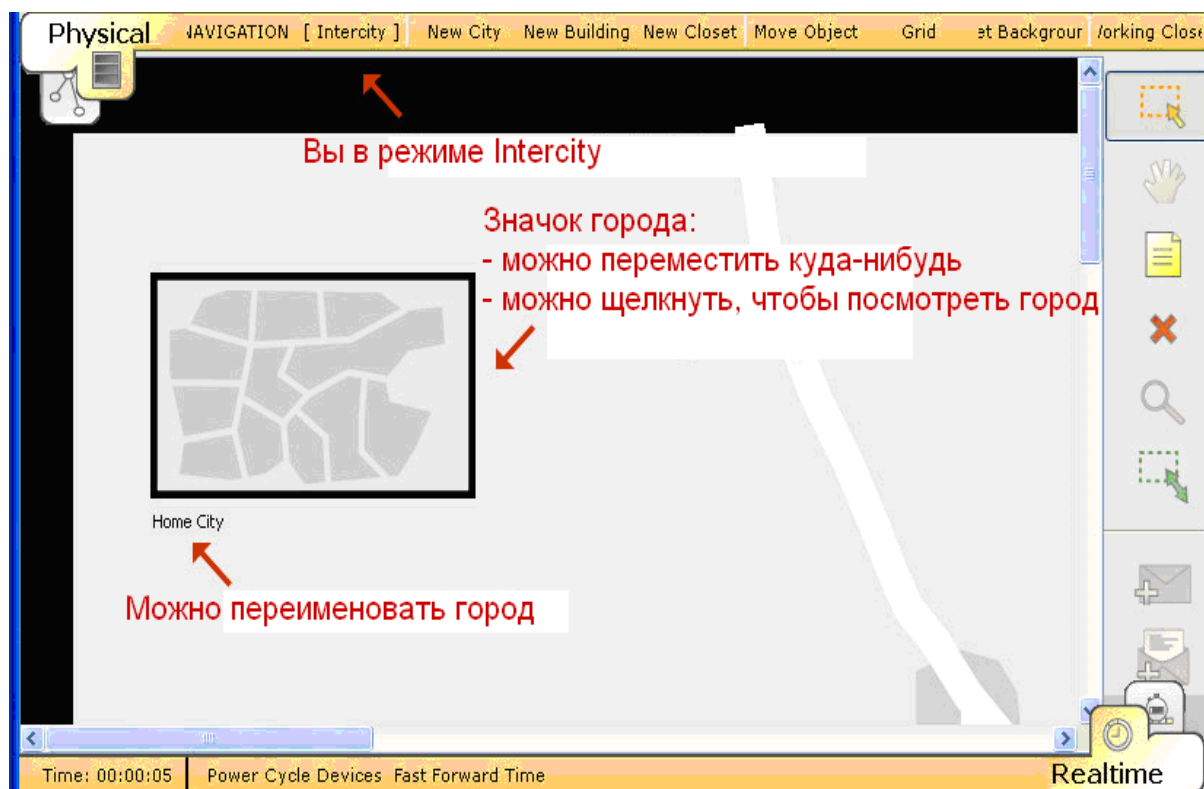


Рис. 21. Междугородный масштаб

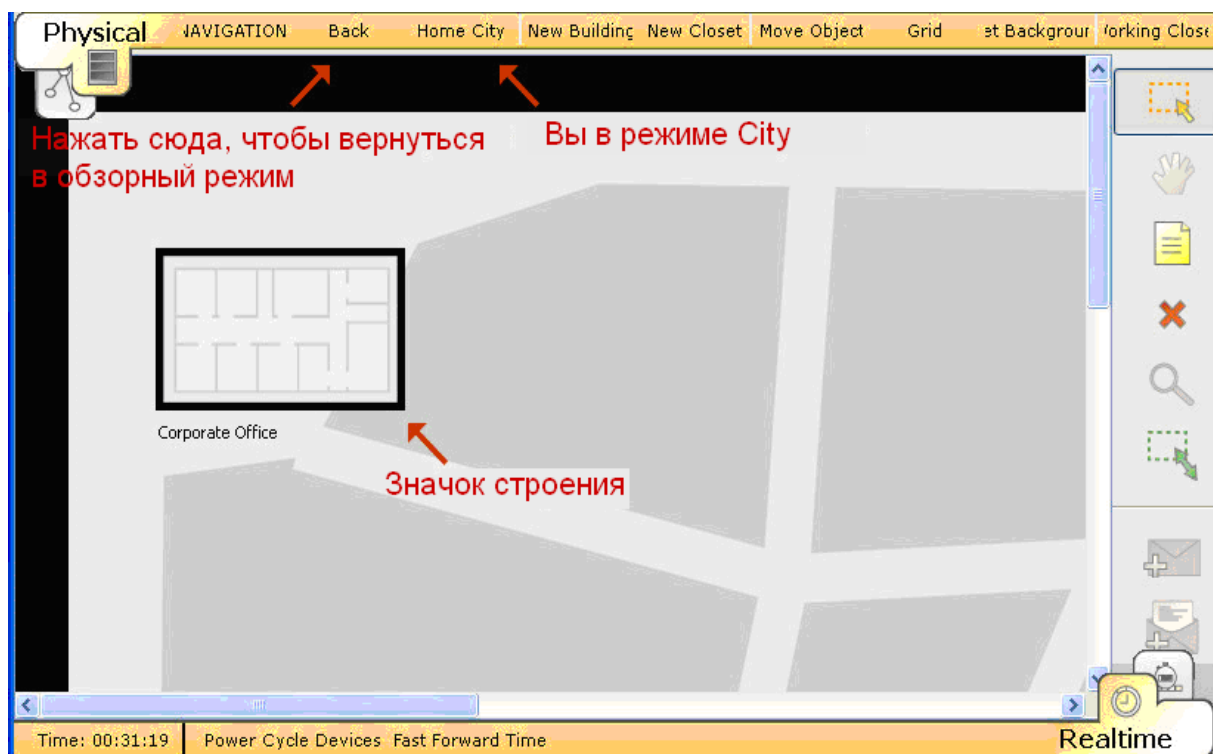


Рис. 22. Масштаб здания

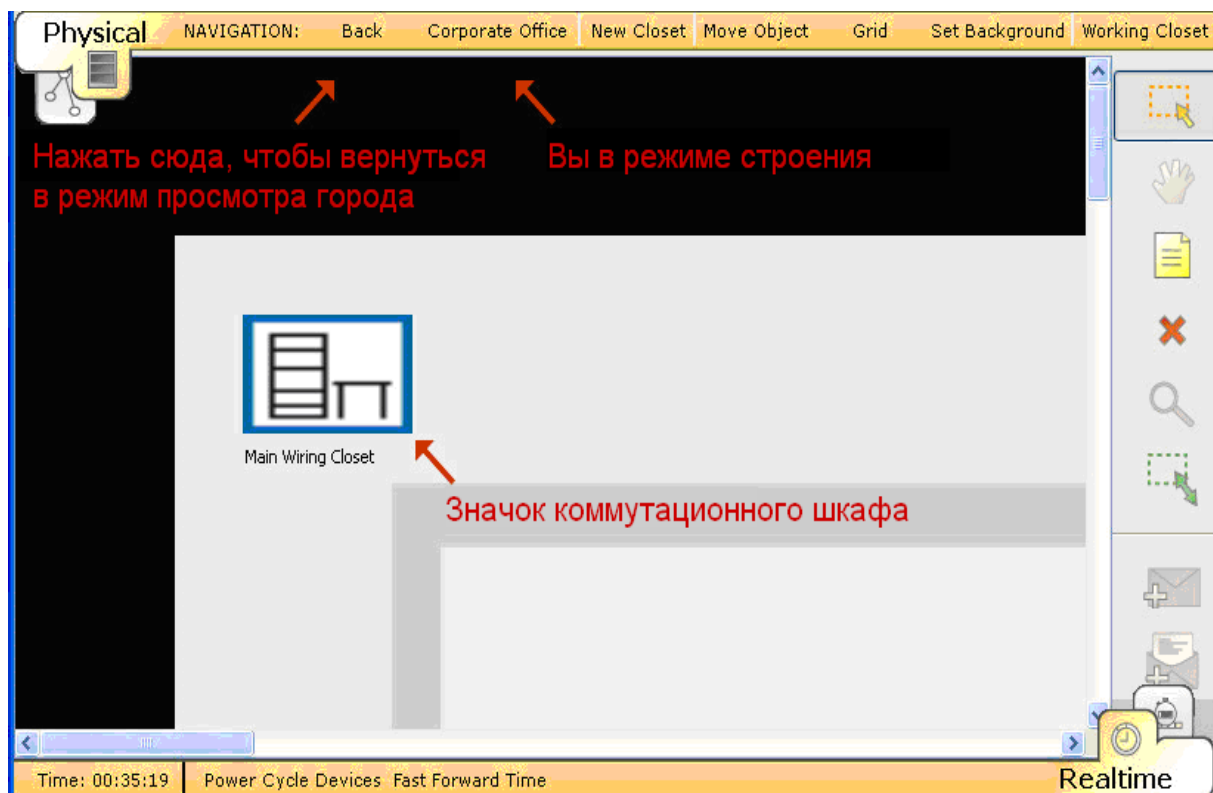


Рис. 23. Коммутационный шкаф



Рис. 24. Устройства, размещенные в коммутационном шкафу

Задание на лабораторную работу

На конкретном примере спроектируйте простую компьютерную сеть на основе концентраторов и проверьте ее работу с помощью утилиты **ping**. Затем проведите моделирование сети по отдельным вариантам.

Методические указания по выполнению лабораторной работы

1. Изучите теоретическую часть лабораторной работы.
2. Запустите на компьютере программный симулятор *Packet Tracer*.
3. Постройте конфигурацию, представленную на рис. 25. Назначьте каждому устройству (кроме хабов) IP-адреса (согласно рис. 25).

Здесь приведены четыре рабочие станции (компьютеры), сервер, принтер и два концентратора. Концентратор (*хаб*) повторяет пакет, пришедший на один порт, на все другие порты. Между собой концентраторы соединяются кроссоверным кабелем (отображается пунктирной линией).

4. После создания топологии, попробуйте с одного из узлов пропинговать другой узел. Для этого перейдите в режим командной строки (см. лабораторную работу № 1). После этого в окне введите команду `ping 192.168.0.5` (рис. 26). Если «пинг» не проходит, еще раз проверьте правильность топологии и настройки сети (IP-адреса узлов).

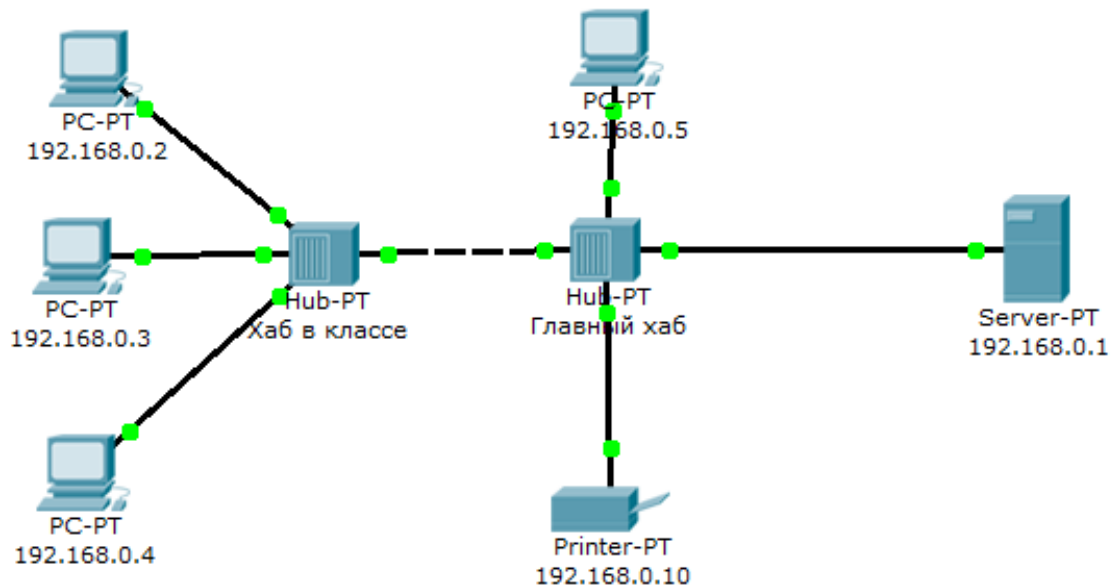


Рис. 25. Простая сеть: два концентратора, четыре компьютера, сервер, принтер

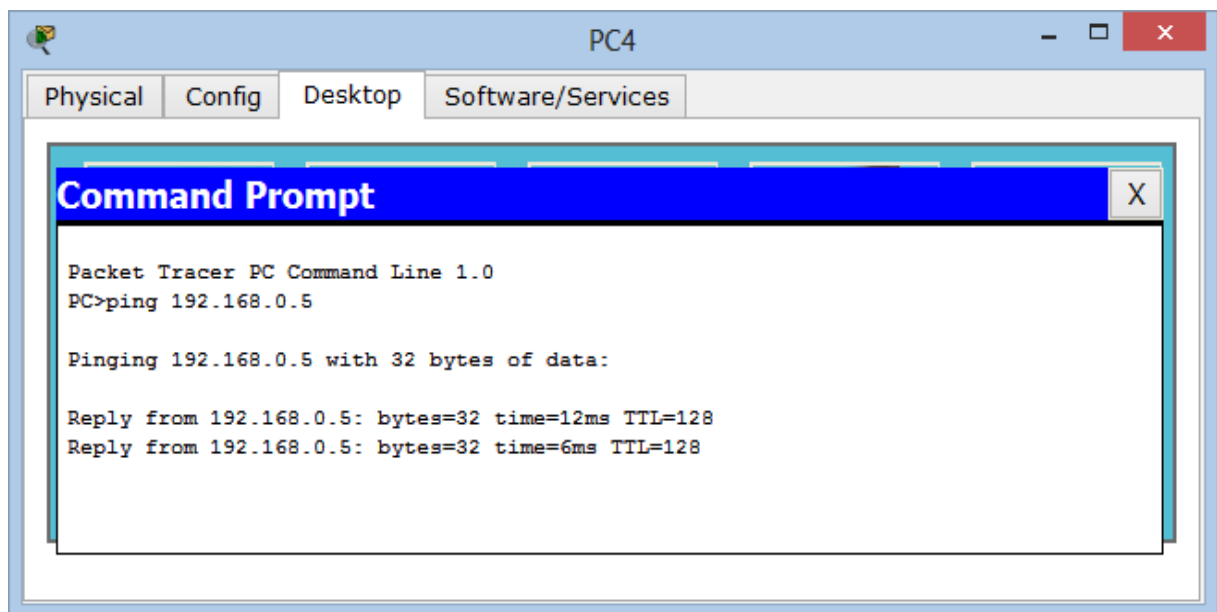


Рис. 26. Пинг: отправка пакета

5. После прохождения «пинга» перейдите в режим симуляции, нажав клавиатурную комбинацию **Shift S** (либо кликнув мышью на иконке *Simulation*).

Здесь присутствует окно списка событий и кнопки управления. В событиях обычно фиксируется много протоколов, но необходимые из них можно отфильтровать. С помощью кнопки *Edit Filters* отфильтруйте только протокол *ICMP*, это позволит исключить загромождение списка случайным трафиком между узлами (рис. 27).

6. Запустите процесс симуляции в пошаговом режиме. В этом режиме для перехода к следующему событию используйте кнопку *Capture / Forward*. Есть также возможность автоматического воспроизведения. Для этого следует нажать кнопку *Auto Capture / Play*. При этом можно регулировать скорость воспроизведения процесса (анимации) специальным движком, расположенным ниже (см. рис. 27).

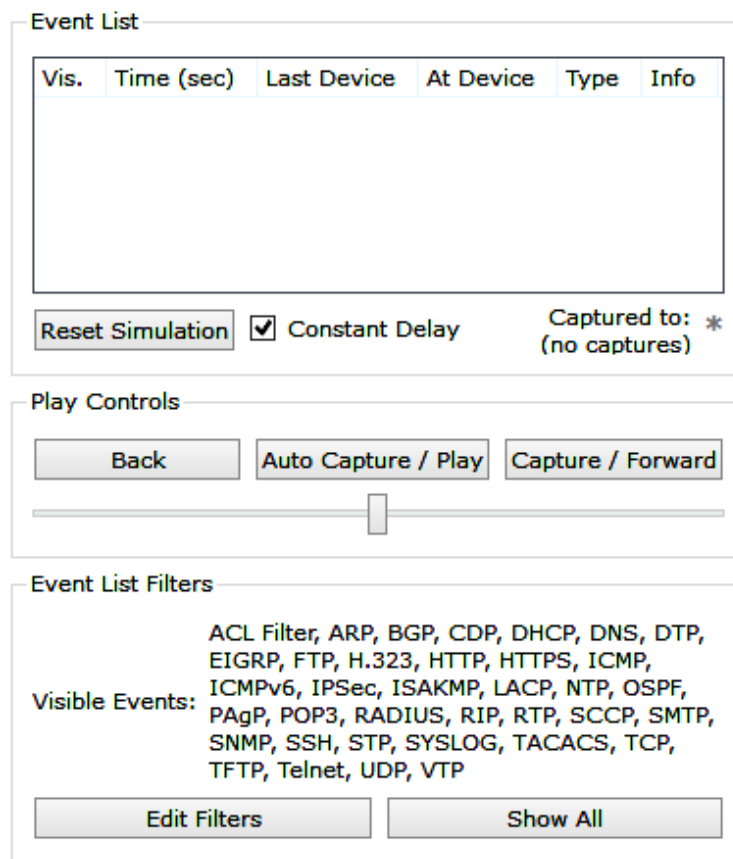


Рис. 27. Окно списка событий

Чтобы нагляднее увидеть, как будут проходить пакеты по сети в режиме симуляции, выберите узлы, расположенные далеко друг от друга, например **192.168.0.4** и **192.168.0.5** (рис. 28).

В самом начале на схеме сети на узле **192.168.0.4** образуется пакет (конвертик), ждущий отправки (иконка паузы). Нажмите кнопку *Capture / Forward*, чтобы стронуть его с места (рис. 28).

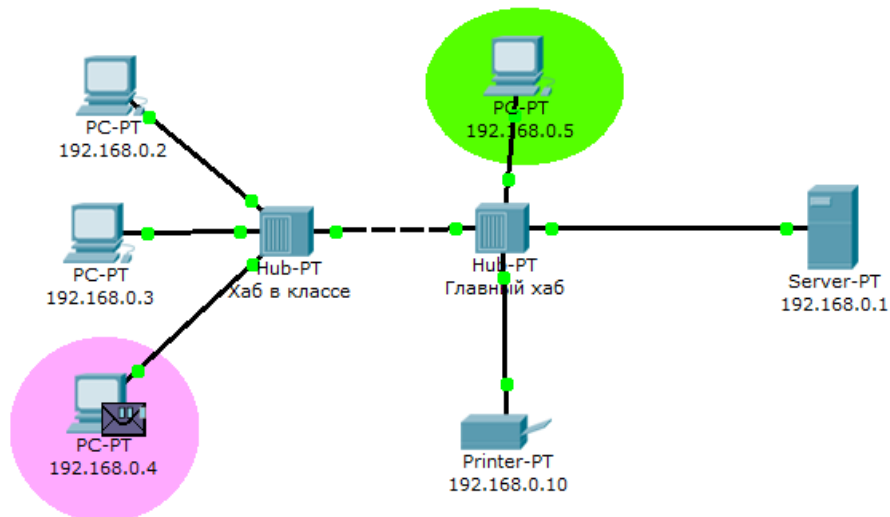


Рис. 28. Пинг: начало процесса

Пакет начнет перемещение, и одновременно с этим в списке событий появится соответствующая запись с указанием типа пакета (*ICMP*) и источника (**192.168.0.4**) (рис. 29).

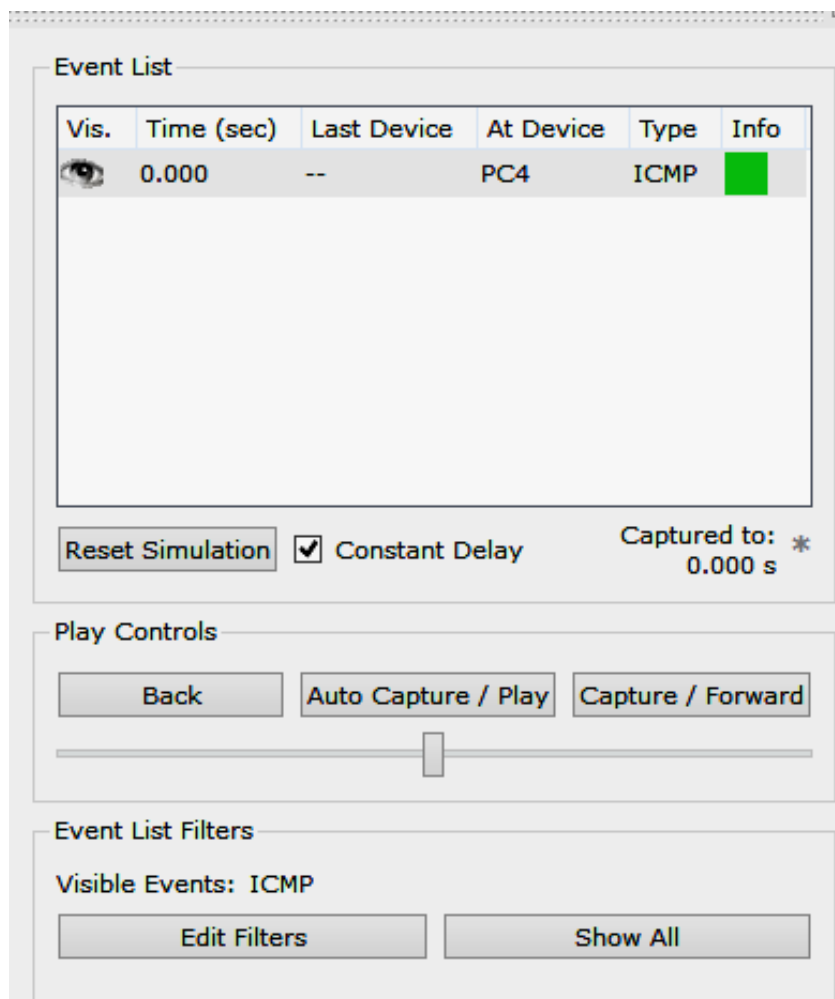


Рис. 29. Пинг: запись в списке событий

7. Для просмотра подробной информации о пакете кликните на событии в списке (рис. 30). Откроется вкладка *OSI Model*, отображающая субординатное перемещение пакета по уровням модели *OSI*. Видно, что пакет возникает на исходящем направлении на третьем (сетевом) уровне. Далее пакет идет на второй уровень (канальный), затем на первый (уровень физической среды) и потом передается на следующий узел. При переходе на другой узел пакет последовательно поднимается по уровням в обратном порядке (с первого до третьего).

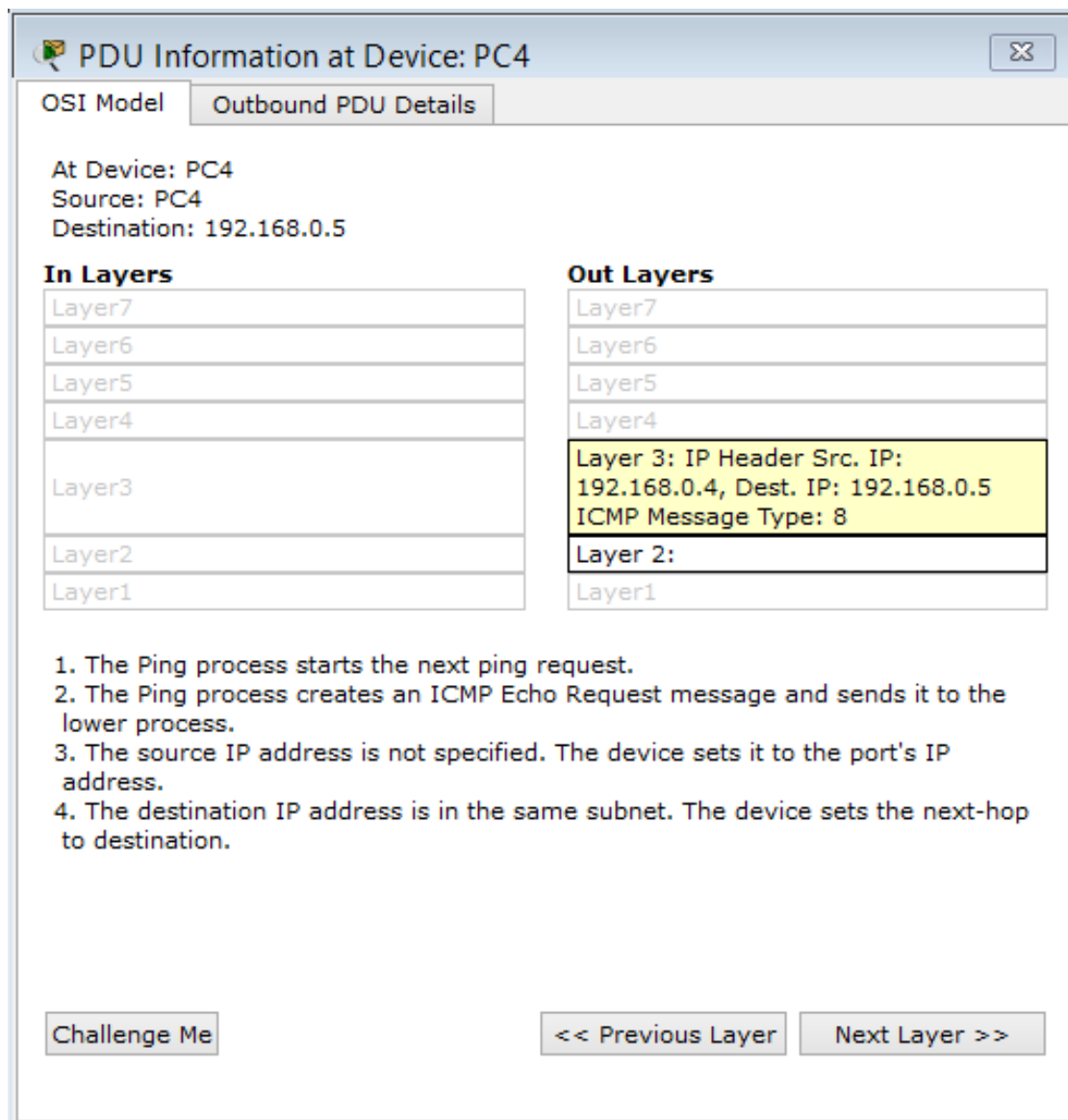


Рис. 30. Пинг: пакет *ICMP* в структуре сетевой модели *OSI*

На другой вкладке – *Outbound PDU Details* – можно подробно просмотреть структуру пакета (рис. 31).

Примечание. В некоторых случаях пакет, отдаваемый устройством, отличается от принятого, тогда добавляется еще одна вкладка. Переключаясь между ними, можно сравнить содержимое этих пакетов.

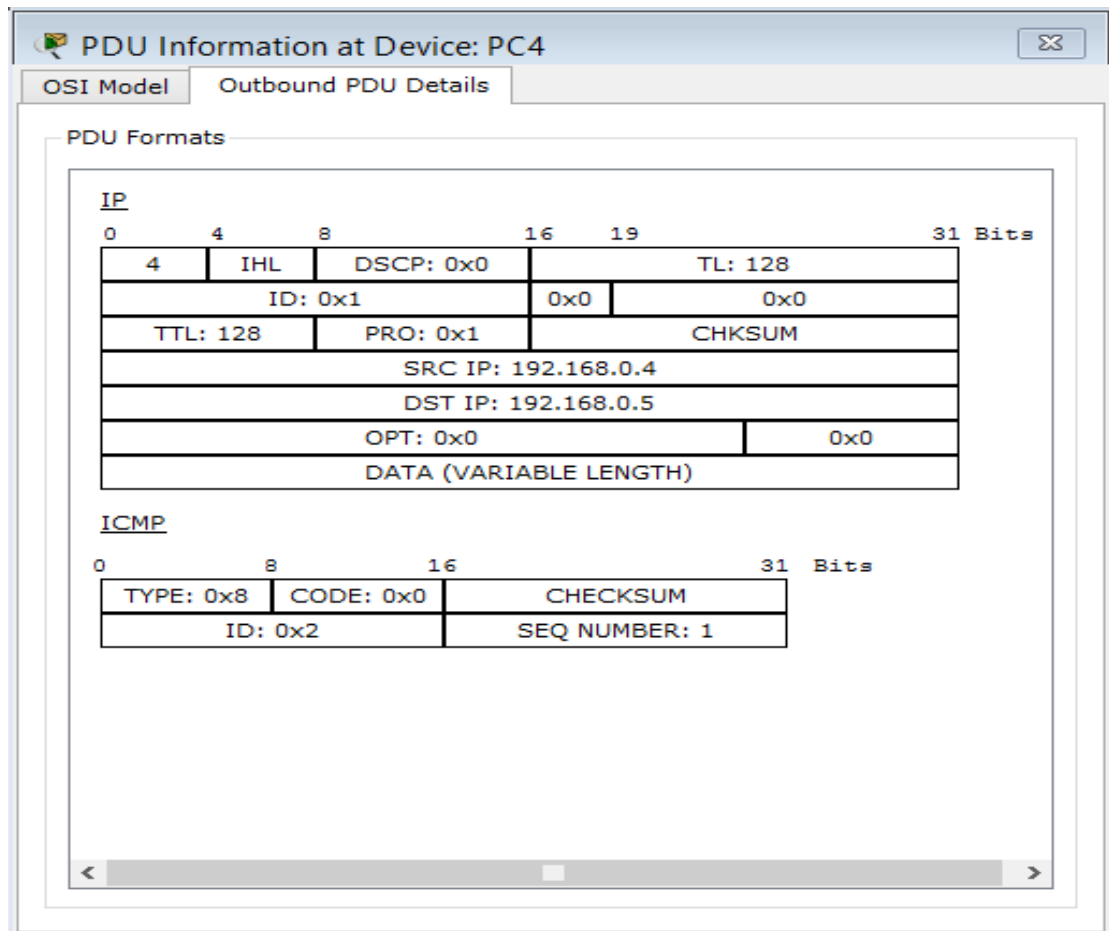


Рис. 31. Пинг: структура пакета

После нажатия кнопки *Capture / Forward* пакет начинает движение по единственному сетевому подключению и передается на концентратор (рис. 32).

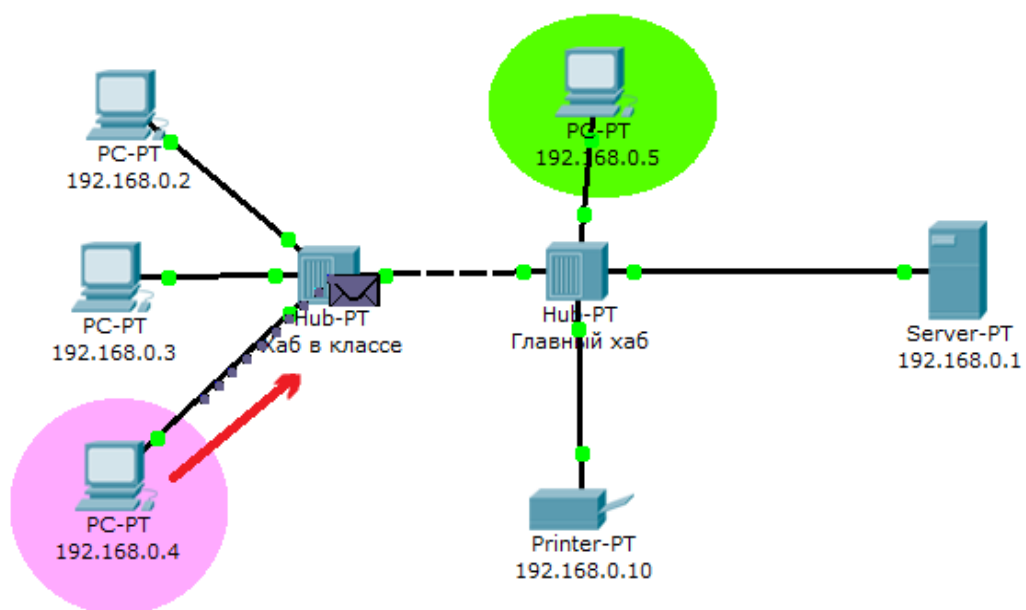


Рис. 32. Пинг: продвижение пакета на концентратор

Концентратор повторяет пакет на все остальные порты в предположении, что на одном из них есть адресат (рис. 33).

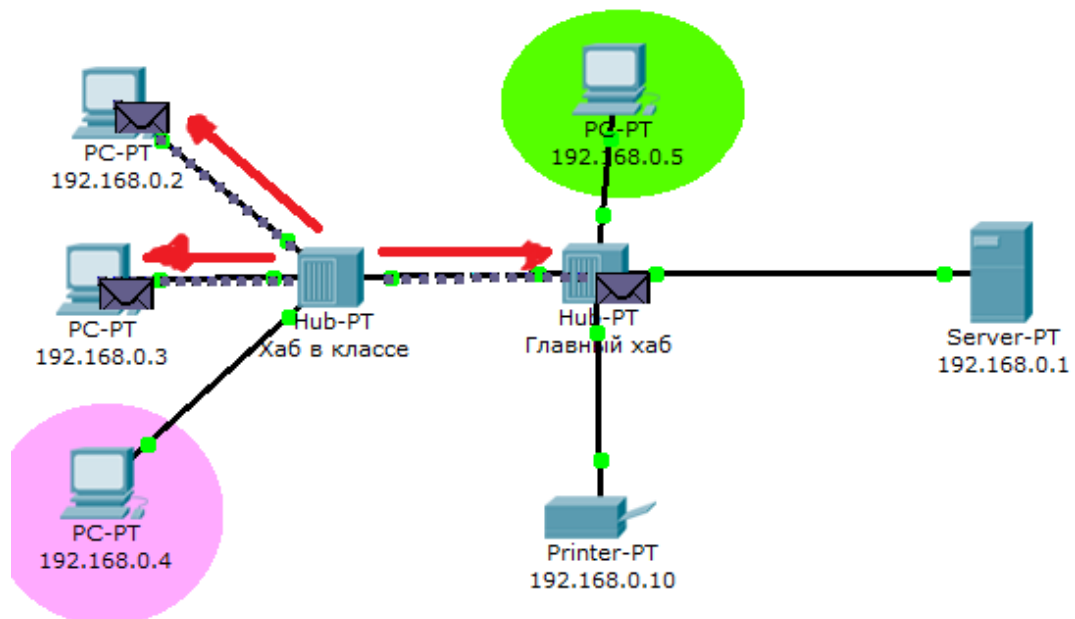


Рис. 33. Пинг: продвижение пакета от концентратора

Узел, получивший пакет, для него не предназначенный, этот пакет должен проигнорировать (рис. 34), исключение возможно в том случае, если на узле установлено специальное программное обеспечение – *сниффер*.

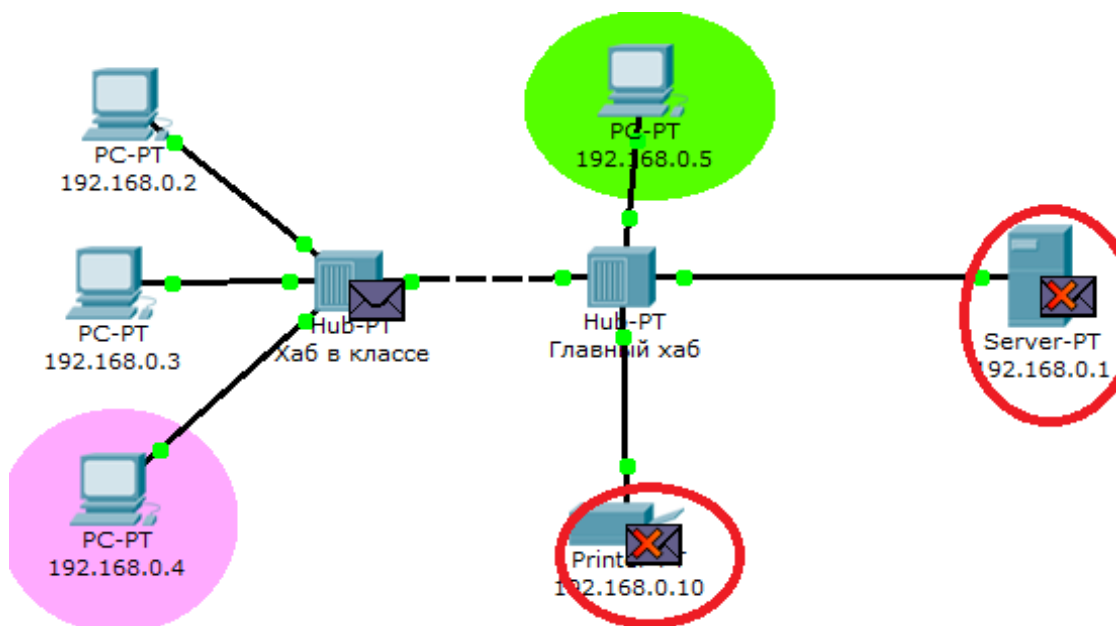


Рис. 34. Пинг: нецелевые пакеты узлы должны сбросить

Узел-получатель при поступлении пакета *ICMP* отправляет ответ. Узел-отправитель получает его, измеряет задержку между отправлением и получением пинга и формирует отчет примерно таким образом:

Reply from 192.168.0.5: bytes=32 time=16ms TTL=128

8. Переходим к моделированию работы сети по вариантам. Определите диапазон *IP*-адресов для своего варианта по шаблону 192.168.*N*.*, где *N* – номер варианта (по списку в журнале).

9. Выполните действия, описанные в разделе «Анализ работы сети» данной лабораторной работы, в соответствии со своим номером варианта.

10. Исследуйте и последовательно запишите, на каких уровнях модели *OSI* активизируется *ICMP*-пакет в процессе своего следования от узла-отправителя к узлу-адресату.

11. Исследуйте, как изменяется содержимое *ICMP*-пакета в процессе продвижения по сети. Проанализируйте эти изменения.

12. Составьте отчет о выполненной работе. Отчет должен содержать:

- 1) титульный лист с указанием названия лабораторной работы, фамилии студента, номера группы, варианта задания;
- 2) краткое изложение теоретических сведений по теме (2–3 страницы);
- 3) скриншоты экрана с результатами последовательно выполненных заданий (по своему варианту) и поясняющими комментариями к ним;
- 4) ответы на контрольные вопросы;
- 5) общие выводы по работе (заключение).

Контрольные вопросы

1. Поясните, что представляют собой логическое и физическое пространства в программе *Packet Tracer*. Чем обусловлено такое разделение?

2. Объясните, что такое режим симуляции, режим реального времени.

3. Расскажите, для чего предназначена команда *ping*. Что значит «пропинговать» узел?

4. Объясните, почему некоторые узлы сбрасывают пакеты, приходящие к ним от концентратора. Можно ли заставить их не делать этого и с какой целью?

5. Расскажите, что произойдет, если узел, на который отправлен пакет, в данный момент недоступен (отключен или отсутствует).

Лабораторная работа № 3

ФИЗИЧЕСКАЯ И ЛОГИЧЕСКАЯ СТРУКТУРИЗАЦИЯ СЕТИ. ПРОТОКОЛЫ, УРОВНИ, АДРЕСА

Цель работы: изучить принципы физической и логической структуризации сетей; получить понятие о сетевых протоколах и уровнях сети; научиться определять логические и аппаратные адреса узлов.

Теоретические сведения

Физическая и логическая топологии сети

В простой сети из нескольких компьютеров четко видно, как соединены между собой различные компоненты. Чем больше разрастается сеть, тем сложнее отслеживать местоположение каждого компонента и его связи с сетью. В проводной сети для подключения ко всем узлам используется множество кабелей и сетевых устройств.

При монтаже сетей составляют схему физической топологии, на которой указывают положение каждого узла и способ его подключения к сети. Кроме того, на схеме помечают все провода и сетевые устройства, соединяющие узлы.

На схеме топологии (рис. 35) физические устройства представлены в виде значков. Чтобы облегчить монтаж и устранение неполадок в будущем, важно своевременно обновлять схемы топологии.

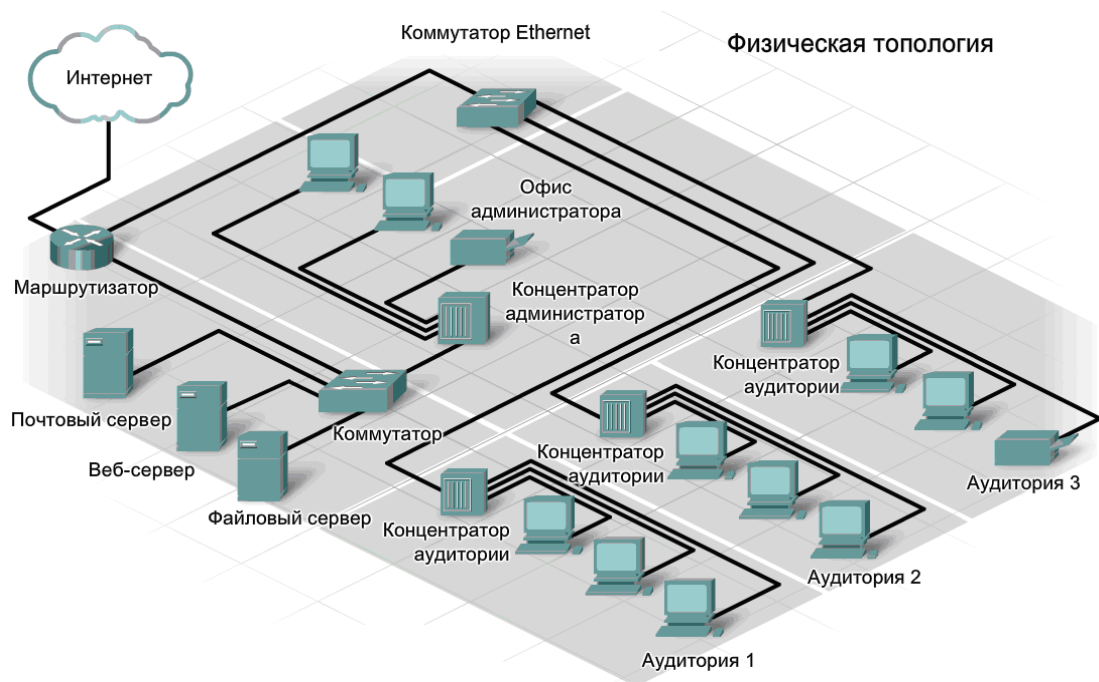


Рис. 35. Физическая топология сети

Помимо схемы физической топологии иногда приходится строить логическое представление топологии сети. На схеме логической топологии (рис. 36) узлы группируются по методам использования сети независимо от местоположения. На такой схеме можно указать имена и адреса узлов, информацию о группах и приложениях.

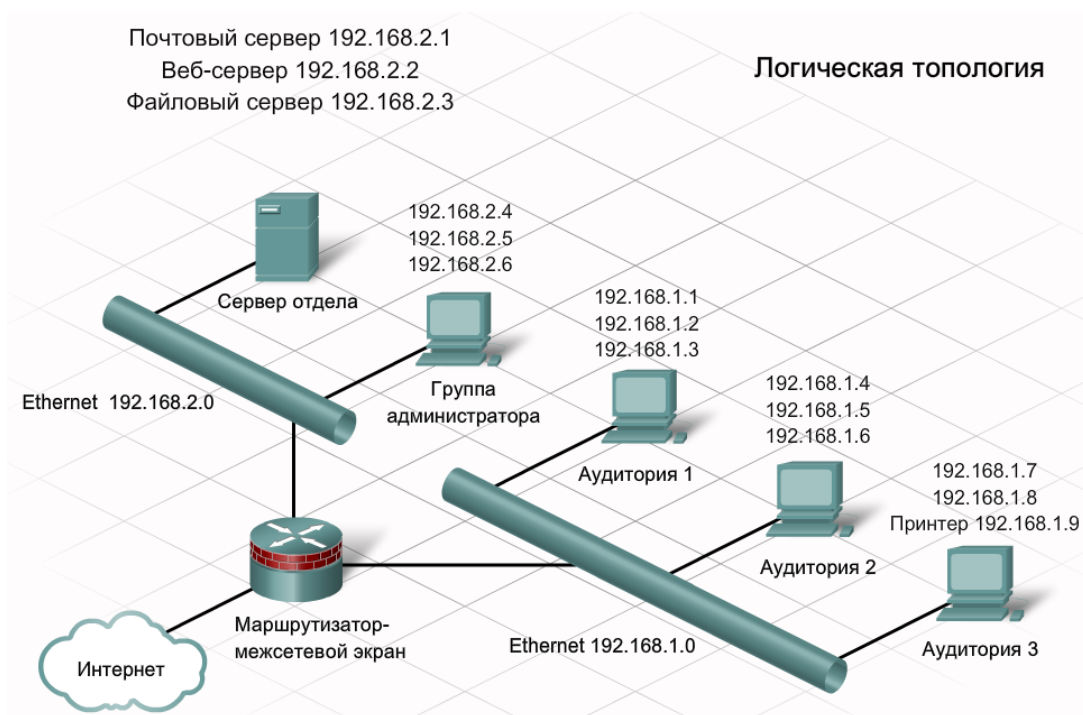


Рис. 36. Логическая топология сети

Основная задача любой сети – передача информации. Любой обмен информацией начинается с сообщения (информации), которое нужно передать от одного человека или устройства к другому. У всех методов обмена данными есть три общих элемента. Первый – это источник сообщения, или отправитель. Отправителем может быть человек или электронное устройство, которому нужно послать сообщение другому человеку или устройству. Второй элемент – это адресат, или получатель сообщения. Адресат получает и интерпретирует сообщение. Третий элемент – канал, это путь, по которому сообщение идет от источника к адресату.

Сетевые протоколы

При организации общения между двумя объектами необходимо учитывать правила, чтобы сообщения между ними были успешно доставлены и поняты. Эти правила составляют набор *протоколов*. К категории протоколов обмена информацией относятся, например:

- идентификация отправителя и получателя;
- выбранное средство или канал связи;
- режим обмена данными;
- общий язык;
- грамматическая структура и структура предложений;
- скорость и время доставки.

Выбор протоколов зависит от характеристик источника, канала и адресата сообщения. Протоколы подробно определяют способ передачи и доставки сообщения, а именно:

- формат сообщения;
- размер сообщения;
- синхронизацию;
- инкапсуляцию;
- кодирование;
- метод рассылки стандартного сообщения.

Один из первых этапов отправки сообщения – кодирование. *Кодирование* – это процесс преобразования передаваемого по сети сообщения в биты. Каждый бит кодируется набором звуков, световых волн или электрических импульсов, в зависимости от типа сети. *Декодирование* – это обратный процесс, т.е. расшифровка сообщения из битов.

При отправке сообщения от источника к адресату необходимо использовать определенный формат или структуру. Формат зависит от типа сообщения и канала доставки. Чтобы поместить сообщение в канал, его нужно преобразовать в формат, соответствующий этому каналу. Такая процедура называется *инкапсуляцией*. Для инкапсуляции каждого сообщения компьютера перед отправкой по сети используется особый формат, который называется *кадром*. Кадр действует примерно так же, как, например, почтовый конверт: в нем указаны адреса узла-источника и узла-назначения.

Формат и содержимое кадра зависят от типа сообщения и канала обмена данными. Если сообщение отформатировано неверно, узел назначения не сможет его успешно получить и обработать.

При передаче длинного сообщения от одного узла к другому по сети необходимо разделить его на части. Такое деление достигается с помощью кадров. Размеры кадров строго регулируются. Кроме всего прочего, они зависят от используемого канала. Слишком длинные или короткие кадры не доставляются.

Ограничения по размеру кадров заставляют узел-источник делить длинные сообщения на части, соответствующие требованиям к минимальному

и максимальному размеру. Каждая часть инкапсулируется в отдельный кадр с информацией об адресе и передается по сети. Узел-адресат распаковывает сообщения и собирает вместе для обработки и интерпретации.

Синхронизация передачи

Одним из факторов, которые влияют на качество приема и понимания сообщения, является синхронизация. Она решается с помощью метода доступа, управления потоками и использования тайм-аутов.

Метод доступа определяет, когда конкретный компьютер сможет отправить сообщение. Выбор времени зависит от среды. Например, в некоторых случаях компьютер может начать передачу в любой момент. Однако, если два узла начинают передавать одновременно, происходит информационная коллизия и обоим приходится начинать сначала. Чтобы узнать, когда начать отправку сообщений и как реагировать на ошибки, узлам в сети нужно определить метод доступа.

Синхронизация влияет и на количество отправляемой информации, и на скорость доставки. Узел-отправитель может передавать сообщения быстрее, чем узел назначения их принимает и обрабатывает. *Управление потоком* позволяет узлу-источнику и узлу назначения согласовать время для успешного обмена данными.

Если сетевой узел отправил кадр и не получил подтверждение, что кадр доставлен за приемлемое время (называемое *тайм-аутом*), он предполагает, что ответа не будет и предпринимает соответствующие действия. Он может повторить отправку кадра или просто продолжить передачу. Это зависит от характера передаваемой информации и параметров настройки узла.

Бывает, что информацию нужно передать только одному узлу. Такой метод рассылки «один к одному» называется одноадресным, или *point-to-point (PPP)*. Если сообщение нужно передать группе получателей, такой метод называется «один ко многим», или «один ко всем», т.е. это означает, что рассылка многоадресная, или *широковещательная*. Широковещательная рассылка предусматривает одновременную отправку одного и того же сообщения группе узлов. Предусмотрены правила рассылки сообщений с подтверждением и без него.

Ethernet

Итак, любой обмен данными между элементами сети подчиняется заранее установленным правилам, или протоколам. Эти протоколы зависят от характеристик источника, канала и адресата. Они четко определяют форматы

и размер сообщений, синхронизацию, характеристики инкапсуляции, кодирования и метод рассылки стандартного сообщения.

В локальных проводных сетях чаще всего используется протокол *Ethernet*. Он определяет многие аспекты обмена данными в локальной сети, например: формат и размер сообщения, синхронизацию, кодирование и методы рассылки сообщений.

В самом начале распространения сетевых технологий различные производители предлагали собственные, *проприетарные* методы связи сетевых устройств и сетевые протоколы, например *IBM*, *NCR*, *Xerox*, *DEC*, *HP*, *IOLA* и др. При этом оборудование от одного поставщика не в состоянии было обмениваться данными с оборудованием другого. Однако по мере распространения сетей разрабатывались стандартные правила работы сетевого оборудования различных производителей. В результате:

- упростилась конструкция сетей;
- упростилась разработка продуктов;
- появились новые возможности для конкуренции;
- появилась возможность связывать разные устройства;
- упростилось обучение;
- расширился выбор поставщиков.

Официально принятого протокола локальных сетей до сих пор не существует, но с течением времени в силу наибольшего распространения стандартом де-факто стал *Ethernet*.

MAC-адреса

Для любого обмена данными необходим способ идентификации источника и адресата. Каждому подключенному к *Ethernet* узлу присваивается физический адрес, который служит его идентификатором в сети. В процессе изготовления всем сетевым интерфейсам *Ethernet* даются физические адреса. Они называются адресами управления доступом к среде (MAC-адресами). MAC-адрес идентифицирует каждый узел источника и каждый узел назначения в сети.

Сети *Ethernet* прокладываются с помощью медных или оптоволоконных кабелей, соединяющих узлы и сетевые устройства. Они представляют собой канал связи между узлами. Когда подключенный к *Ethernet* узел включается в обмен данными, он рассылает кадры со своим MAC-адресом в качестве источника и MAC-адресом предполагаемого

получателя. Все принимающие узлы декодируют кадр и считывают *MAC*-адрес назначения. Если он соответствует настроенному *MAC*-адресу сетевой интерфейсной платы, она обрабатывает и сохраняет сообщение. Если *MAC*-адрес назначения не соответствует *MAC*-адресу узла, сетевой адаптер игнорирует сообщение.

Определить *MAC*-адрес компьютера в сети *Ethernet*, как правило, можно с помощью следующей команды:

ipconfig /all

Большинство сетевых ОС поддерживают эту команду. На компьютере, работающем под управлением ОС *Windows*, для этого необходимо выполнить следующие действия.

На рабочем столе *Windows* нажмите кнопку *Пуск* и выберите команду *Выполнить* (либо нажмите клавиатурную комбинацию **Win R**). В диалоговом окне *Запуск программы* введите команду **cmd** и нажмите кнопку *ОК* (рис. 37).

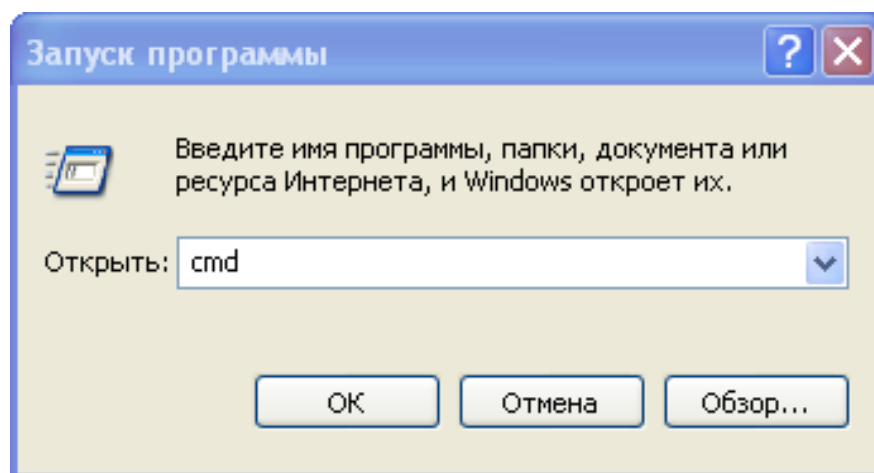


Рис. 37. Запуск программы

Откроется окно командной строки *Windows*. В командной строке введите команду **ipconfig /all** и нажмите **Enter**. На рис. 38 показан типичный результат работы такой команды, однако на разных компьютерах может отображаться разная информация.


```

% Командная строка
C:\>ipconfig /all

Настройка протокола IP для Windows

    Имя компьютера . . . . . : k909
    Основной DNS-суффикс . . . . . :
    Тип узла . . . . . : неизвестный
    IP-маршрутизация включена . . . . : нет
    WINS-прокси включен . . . . . : нет
    Порядок просмотра суффиксов DNS . : Home

Подключение по локальной сети - Ethernet адаптер:

    Состояние сети . . . . . : сеть отключена
    Описание . . . . . : VIA PCI 10/100Mb Fast Ethernet адаптер
    Физический адрес. . . . . : 00-11-5B-6B-88-DB

Беспроводное сетевое соединение - Ethernet адаптер:

    DNS-суффикс этого подключения . . : Home
    Описание . . . . . : D-Link Wireless 108G DWA-520 Desktop Adapter
    Физический адрес. . . . . : 00-1C-F0-D7-13-94
    DHCP включен. . . . . : да
    Автонастройка включена . . . . . : да
    IP-адрес . . . . . : 192.168.1.4
    Маска подсети . . . . . : 255.255.255.0
    Основной шлюз . . . . . : 192.168.1.1
    DHCP-сервер . . . . . : 192.168.1.1
    DNS-серверы . . . . . : 192.168.1.1
    Аренда получена . . . . . : 26 сентября 2016 г. 19:57:27
    Аренда истекает . . . . . : 27 сентября 2016 г. 19:57:27

```

Рис. 38. Результат работы команды **ipconfig**

В полученной сводке найдите физический *MAC*-адрес своего компьютера. *MAC*-адреса компьютеров обычно представлены набором из 6 двухразрядных шестнадцатеричных чисел, разделенных тире или двоеточием, например: **00-11-5B-6B-88-DB**.

Формат кадров Ethernet

Стандартные протоколы *Ethernet* определяют многие аспекты сетевого обмена данными, включая формат и размер кадра, синхронизацию и кодировку.

Когда подключенные к сети *Ethernet* узлы отправляют сообщения, они форматируют их в соответствии со стандартами макета кадра. Кадры иначе называют протокольными блоками данных (*PDU*).

Формат кадров *Ethernet* определяет положение *MAC*-адресов получателя и источника и дополнительную информацию, в том числе такую как:

- преамбула для последовательности и синхронизации;
- разделитель начала кадра;
- длина и тип кадра;
- контрольная последовательность кадра (для обнаружения ошибок передачи).

Максимальный размер кадров *Ethernet* (начиная с поля *MAC*-адреса назначения и заканчивая контрольной последовательностью кадра) составляет 1518 байт, а минимальный – 64 байта. Не входящие в этот диапазон кадры не обрабатываются. Помимо форматов, размеров и синхронизации кадра стандарты *Ethernet* определяют кодирование битов кадра при передаче по каналу. По медному кабелю биты передаются в виде электрических импульсов, по оптоволоконному кабелю – в виде световых импульсов (рис. 39).

Преамбула	Разделитель начала кадра	MAC-адрес назначения	MAC-адрес источника	Длина / тип	Инкапсулированные данные	Контрольная последовательность кадра
7	1	6	6	2	с 46 по 1500	4

Рис. 39. Структура кадра *Ethernet* (указано количество байтов для каждой секции)

Иерархическая структура сетей Ethernet

MAC-адрес узла идентифицирует конкретный узел, но не указывает, в каком конкретно месте сети он находится. При наличии большого количества узлов определение местонахождения какого-либо из них крайне сложно.

Кроме того, при обмене данными между узлами технология *Ethernet* генерирует большой объем широковещательного трафика. Широковещательные рассылки отправляются всем узлам, подключенным к одной сети. Они занимают часть полосы пропускания и значительно замедляют работу сети.

Поэтому большие сети *Ethernet*, состоящие из многих узлов, неэффективны. Крупные сети лучше разделить на более мелкие и более управляемые части. Один из способов деления предполагает использование модели иерархической архитектуры сети. Такая конструкция позволяет группировать устройства по нескольким сетям, организуя уровни. Они состоят из меньших, более управляемых групп, в которых локальный трафик остается локальным. На верхний уровень попадает только трафик, предназначенный для других сетей.

Иерархическая, многоуровневая конструкция повышает эффективность, оптимизирует систему и увеличивает скорость. Она дает возможность масштабировать сеть по мере необходимости, позволяя добавлять локальные сети, не снижая эффективности уже существующих.

В иерархической конструкции существует три основных уровня (рис. 40):

- уровень доступа – соединяет узлы в локальной сети *Ethernet*;
- уровень распределения – соединяет небольшие локальные сети;
- центральный уровень – осуществляет высокоскоростное соединение между устройствами уровня распределения.

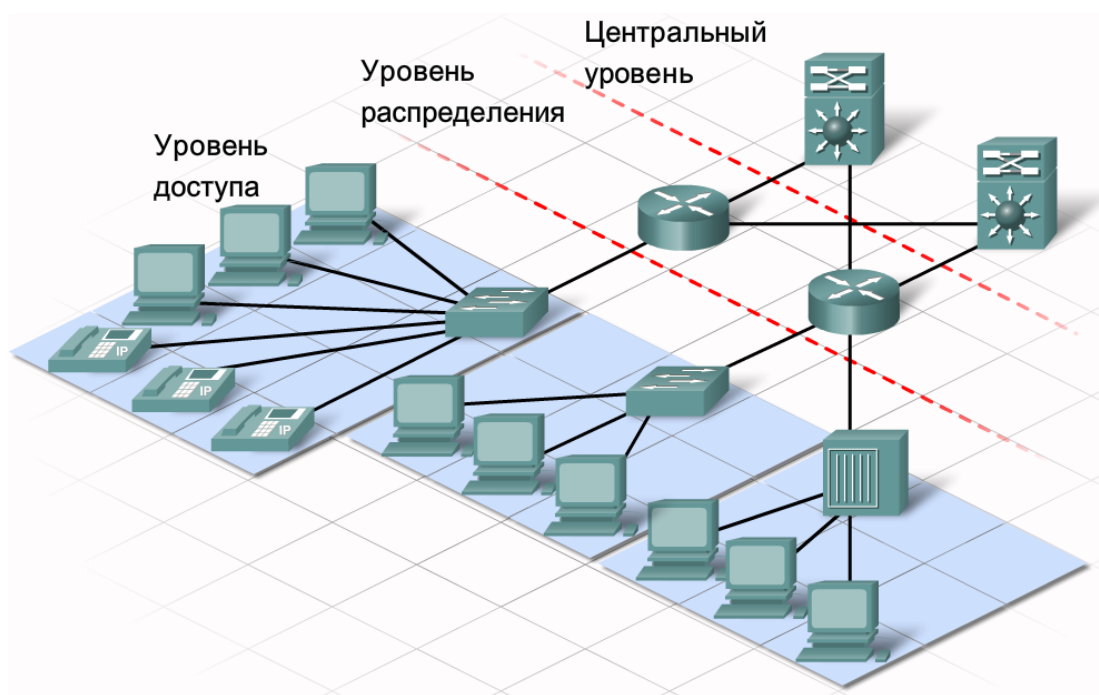


Рис. 40. Иерархическая конструкция сети *Ethernet*

В такой конструкции необходима схема логической адресации, позволяющая определить положение узла. Такая схема адресации называется межсетевым протоколом, а адреса – *IP*-адресами.

В отличие от физического *MAC*-адреса, который прописан в сетевых устройствах аппаратно, *IP*-адрес присваивается логически, в зависимости от местонахождения узла. *IP*-адрес, или сетевой адрес, присваивается узлу сетевым администратором на основе характеристик локальной сети.

IP-адреса состоят из двух частей (рис. 41). Одна из них является идентификатором локальной сети. Сетевая часть *IP*-адреса общая у всех узлов в одной локальной сети. Вторая часть *IP*-адреса является идентификатором конкретного узла. Относящаяся к узлу часть *IP*-адреса в одной локальной сети не повторяется.

Определить *IP*-адрес компьютера можно, как правило, командой **ipconfig /all**. Поскольку вы уже знакомы с этой командой, изучите информацию на экране, полученную с ее помощью.

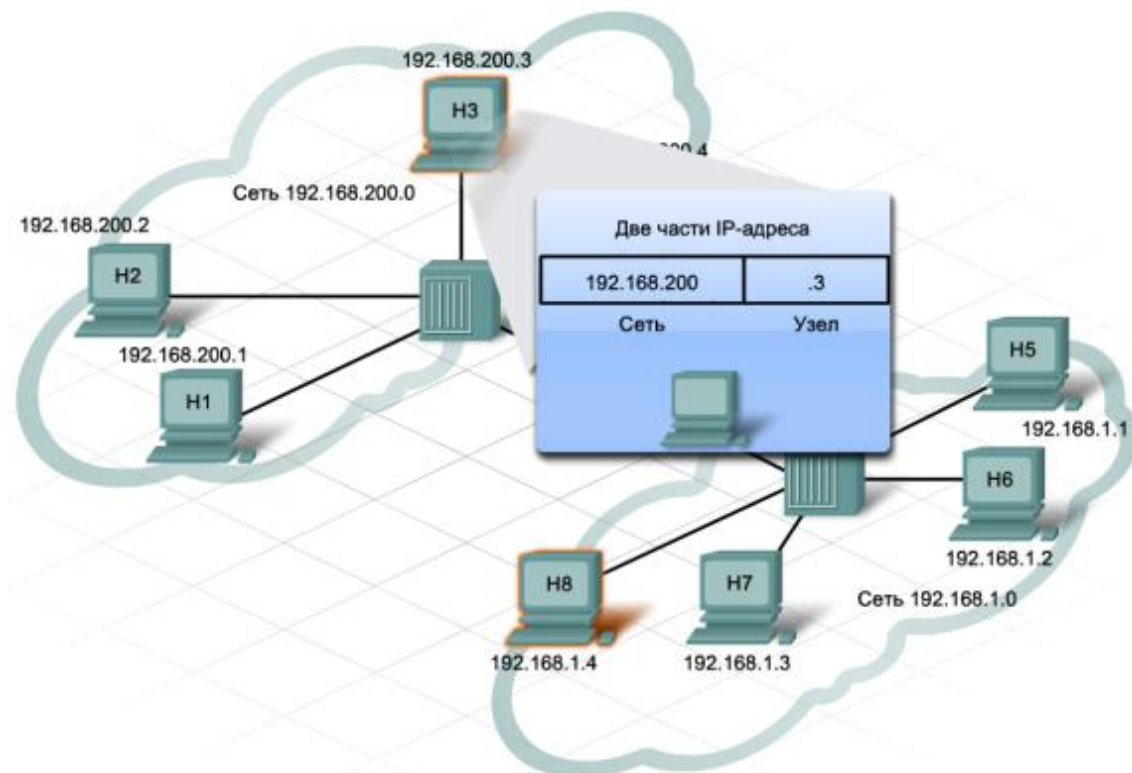


Рис. 41. Две части IP-адреса

Уровни сети

IP-трафик распределяется в зависимости от характеристик и устройств каждого из трех уровней: доступа, распределения и центрального. IP-адрес позволяет определить, останется ли трафик локальным или переместится на следующий уровень иерархической сети.

Уровень доступа соединяет устройства конечных пользователей с сетью и позволяет нескольким узлам подключаться к другим узлам через сетевое устройство (обычно это концентратор или коммутатор). Обычно сетевая часть IP-адреса всех устройств одного и того же уровня доступа совпадает.

Если сообщение предназначено локальному узлу, оно остается на локальном уровне (это зависит от сетевой части IP-адреса). Если сообщение предназначено для другой сети, оно передается на уровень распределения. Концентраторы и коммутаторы обеспечивают связь с устройствами уровня распределения (как правило, это маршрутизаторы).

Уровень распределения соединяет разные сети и контролирует потоки информации между сетями. Обычно коммутаторы этого уровня мощнее, чем на уровне доступа. Кроме того, для маршрутизации данных между сетями используются маршрутизаторы. Устройства уровня распределения контролируют тип и объем трафика, идущего с уровня доступа к центральному уровню.

Центральным уровнем называется магистральный высокоскоростной уровень с дублирующими (резервными) соединениями. На этом уровне большие объемы данных передаются между несколькими сетями. Обычно на центральном уровне находятся высокоскоростные коммутаторы и маршрутизаторы. Основная задача центрального уровня – быстрая передача данных.

Уровень доступа

В сети *Ethernet* каждый узел может напрямую соединяться с сетевым устройством уровня доступа с помощью двухточечного кабеля. Такие кабели производятся в соответствии с конкретными стандартами *Ethernet*. Каждый кабель вставляется в разъем сетевого адаптера узла и в порт сетевого устройства. Для подключения узлов на уровне доступа (включая концентраторы и коммутаторы *Ethernet*) используется несколько типов сетевых устройств.

Концентратор

Концентратор (хаб) – простое устройство, не оборудованное необходимыми электронными компонентами для передачи сообщений между узлами в сети (рис. 42). Концентратор не в состоянии определить, какому узлу предназначено конкретное сообщение. Он просто принимает электронные сигналы одного порта и воспроизводит (или ретранслирует) его сообщение для всех остальных портов.

Сетевой адаптер конечного узла принимает только сообщения, адресованные на правильный *MAC*-адрес. Узлы игнорируют сообщения, которые адресованы не им. Только узел, которому адресовано данное сообщение, обрабатывает его и отвечает отправителю.

Для отправки и получения сообщений все порты концентратора *Ethernet* подключаются к одному и тому же каналу. Концентратор называется устройством с общей полосой пропускания, поскольку все узлы в нем работают на одной полосе одного канала.

Через концентратор *Ethernet* можно одновременно отправлять только одно сообщение. Если два или более узла, подключенные к одному концентратору, попытаются одновременно отправить сообщение, произойдет *коллизия*, т.е. столкновение электронных сигналов, из которых состоит сообщение.

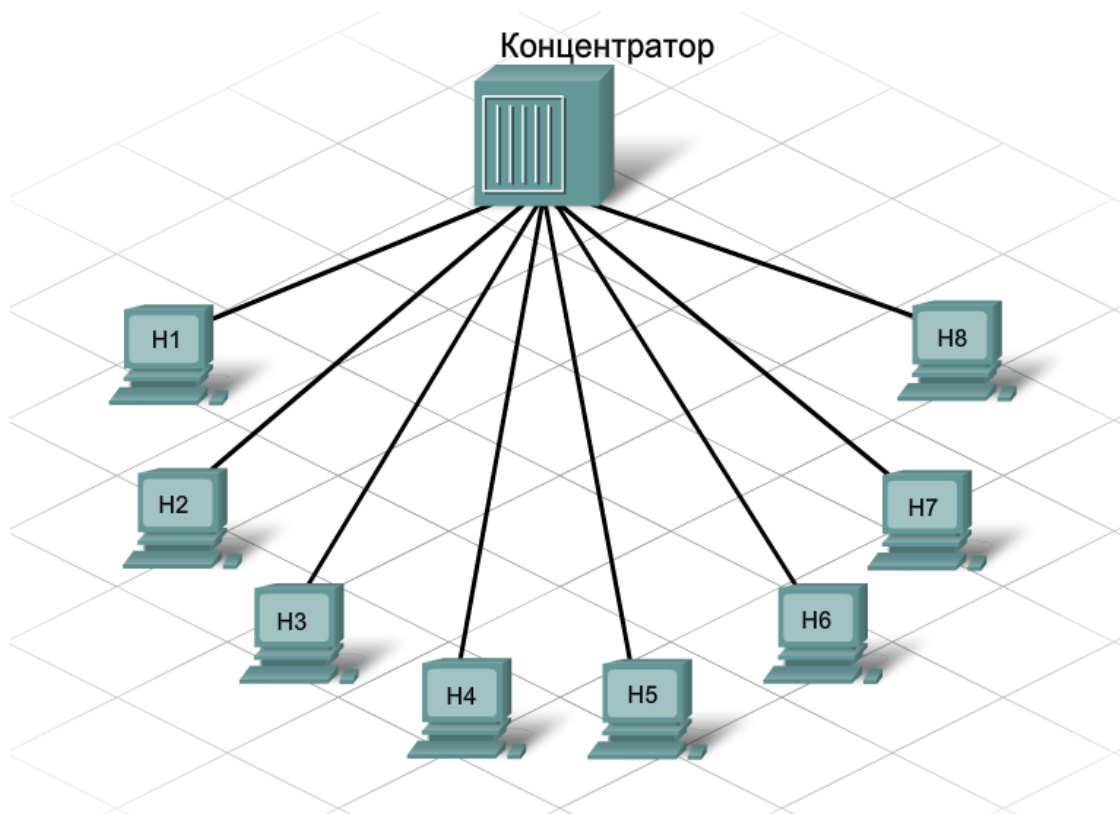


Рис. 42. Концентратор

Кадры, попавшие в коллизию, искажаются. Узлы не смогут их прочесть. Концентратор не декодирует сообщение, поэтому он не обнаруживает, что оно искажено, и повторяет его всем портам. Область сети, в которой узел может получить искаженное при столкновении сообщение, называется *доменом коллизий*.

Внутри этого домена узел, получивший искаженное сообщение, обнаруживает, что произошла коллизия. Каждый отправляющий узел какое-то время ждет и затем пытается снова отправить или переправить сообщение. По мере того, как количество подключенных к концентратору узлов растет, растет и вероятность столкновения. Чем больше столкновений, тем больше повторов. При этом сеть нагружается и скорость передачи сетевого трафика падает. Поэтому размер домена коллизий необходимо ограничить.

Коммутатор

Коммутатор (мост, свитч) также используется на уровне доступа. Как и концентратор, коммутатор соединяет несколько узлов с сетью. В отличие от концентратора, коммутатор в состоянии передать сообщение конкретному узлу. Когда узел отправляет сообщение другому узлу через коммутатор, тот принимает и декодирует кадры и считывает физический (MAC) адрес сообщения.

В таблице коммутатора, которая называется таблицей *MAC*-адресов, находится список активных портов и *MAC*-адресов подключенных к ним узлов (рис. 43). Когда узлы обмениваются сообщениями, коммутатор проверяет, есть ли в таблице *MAC*-адрес. Если да, то коммутатор устанавливает между портом источника и портом назначения временное соединение, которое называется канал. Этот новый канал представляет собой назначенный канал, по которому два узла обмениваются данными. Другие узлы, подключенные к коммутатору, работают на разных полосах пропускания канала и не принимают сообщения, адресованные не им. Для каждого нового соединения между узлами создается новый канал. Такие отдельные каналы позволяют устанавливать несколько соединений одновременно без возникновения коллизий.

Таблица <i>MAC</i> -адресов			
fa0/1	fa0/2	fa0/3	fa0/4
260d.8c01.0000	260d.8c01.1111	260d.8c01.2222	260d.8c01.3333
fa0/5	fa0/6	fa0/7	fa0/8
260d.8c01.4444	260d.8c01.5555		260d.8c01.7777

Рис. 43. Таблица *MAC*-адресов коммутатора

Если *MAC*-адреса назначения нет в таблице, коммутатор не может создать отдельный канал, поскольку не имеет соответствующей информации. В этом случае он передает широковещательное сообщение всем подключенным узлам. Каждый узел сравнивает *MAC*-адрес назначения сообщения со своим *MAC*-адресом, и узел, которому оно адресовано, отвечает на него.

Когда узел отправляет сообщение или отвечает на широковещательное сообщение, коммутатор сохраняет в таблице соответствие адреса узла и порта, к которому тот подключен. Таблица динамически обновляется каждый раз, когда появляется новый *MAC*-адрес источника.

В случае, если к порту коммутатора подключен концентратор, *MAC*-адреса всех узлов, подключенных к концентратору, связываются с одним портом (рис. 44). Бывает, что один узел подключенного концентратора отправляет сообщения другому узлу того же устройства. В этом случае коммутатор принимает кадр и проверяет местонахождение узла назначения по таблице. Если узлы источника и назначения подключены к одному порту, коммутатор отклоняет сообщение.

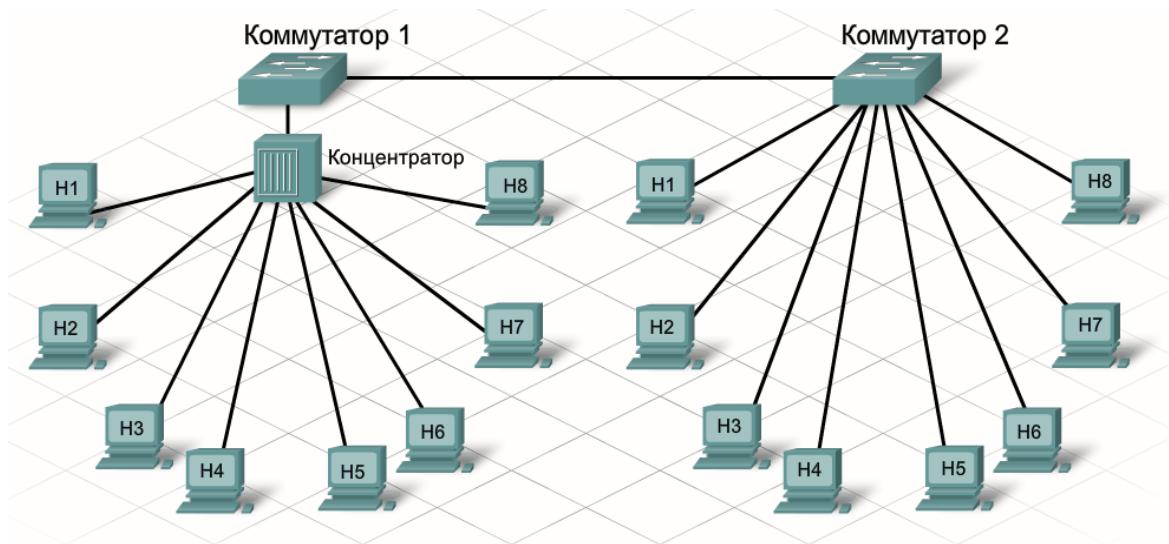


Рис. 44. Подключение концентратора к коммутатору

Если концентратор подключен к порту коммутатора, возможны коллизии. Концентратор передает поврежденные при столкновении сообщения всем портам. Коммутатор принимает поврежденное сообщение, но, в отличие от концентратора, не переправляет его. В итоге у каждого порта коммутатора создается отдельный домен коллизий.

Широковещательная рассылка

Если узлы подключаются через коммутатор или концентратор, образуется единая локальная сеть. В локальной сети одному узлу часто приходится делать широковещательные рассылки (например, если нужно найти информацию, не зная точно, на каком узле она находится, или же нужно своевременно предоставить информацию всем остальным узлам в той же сети).

В сообщении может быть только один *MAC*-адрес назначения, поэтому сообщение отправляется на уникальный *MAC*-адрес, который опознают все узлы. В действительности *MAC*-адрес широковещательной рассылки представляет собой 48-битный адрес, в который входят все остальные адреса. В шестнадцатеричной форме он выглядит как **FFFF.FFFF.FFFF**.

Когда узел получает сообщение на адрес широковещательной рассылки, он его принимает и обрабатывает так же, как и те, что адресованы ему. Когда узел отправляет широковещательное сообщение, концентраторы и коммутаторы передают его всем подключенным к одной локальной сети узлам. Поэтому локальная сеть называется также *доменом широковещательной рассылки*.

Если к одному и тому же домену широковещательной рассылки подключается слишком много узлов, объем широковещательного трафика становится недопустимо большим. Количество узлов и объем сетевого трафика, который поддерживает локальная сеть, ограничивается возможностями используемых концентраторов и коммутаторов. По мере расширения сети и добавления узлов растет и объем сетевого трафика, включая широковещательные рассылки. Для повышения эффективности часто приходится делить одну локальную сеть, или домен широковещательной рассылки, на несколько сетей (рис. 45).

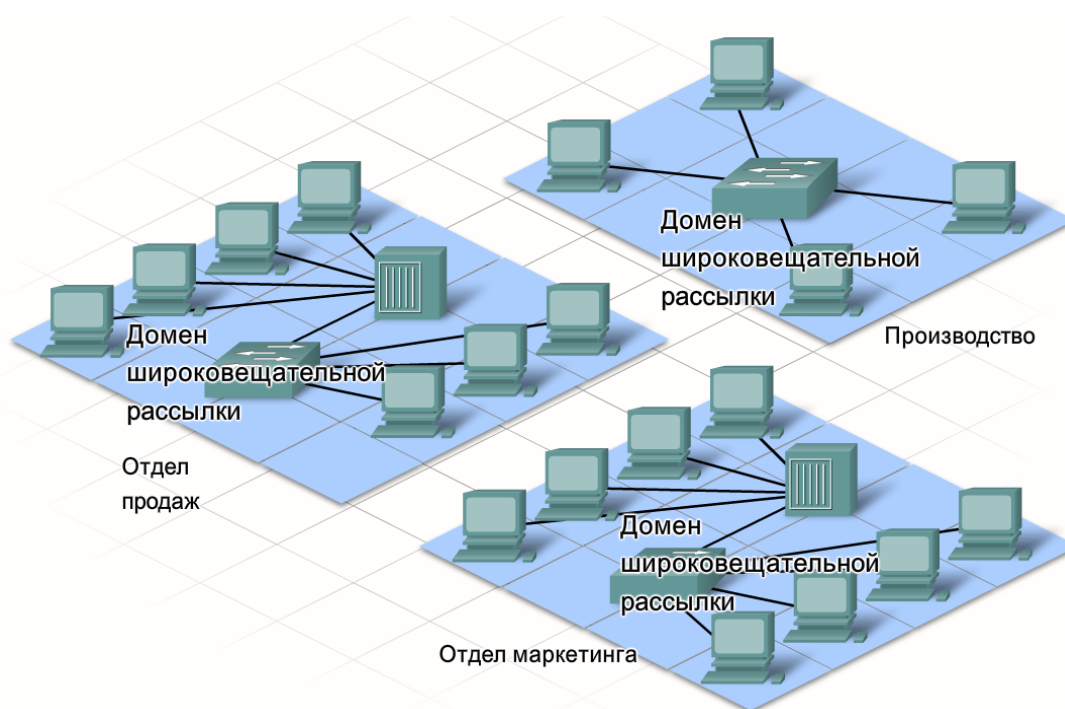


Рис. 45. Деление большой сети на малые

Преобразование адресов

В локальной сети *Ethernet* сетевая интерфейсная плата принимает кадр только в том случае, если он отправлен на *MAC*-адрес широковещательной рассылки или *MAC*-адрес сетевого адаптера. При этом большинство сетевых приложений находят серверы и клиентов только по логическому *IP*-адресу. Определение *MAC*-адреса любого узла из той же локальной сети можно сделать с помощью протокола разрешения адресов (*ARP*).

При наличии *IP*-адреса узла *ARP* определяет и сохраняет *MAC*-адрес узла в локальной сети в три этапа.

1. Отправляющий узел создает и отправляет кадр по *MAC*-адресу широковещательной рассылки. В кадре находится сообщение с *IP*-адресом узла назначения.

2. Каждый сетевой узел получает этот кадр и сравнивает *IP*-адрес из сообщения со своим. Узел с соответствующим *IP*-адресом посылает отправителю свой *MAC*-адрес.

3. Узел-отправитель получает сообщение и сохраняет *MAC*-адрес и *IP*-адрес в таблице *ARP*.

Когда *MAC*-адрес назначения оказывается в таблице *ARP* отправителя, появляется возможность отправлять кадры напрямую, минуя запрос *ARP*.

Воспользоваться возможностями *ARP* можно следующим образом. При помощи команды **ping** пропикуйте узел, *MAC*-адрес которого желаете выяснить, чтобы убедиться, что он подключен, а также, при необходимости, инициировать обновление таблицы адресов. При этом необходимо помнить, что узел должен находиться в этой же локальной сети, поскольку *MAC*-адреса не транслируются за пределы коммутатора.

В некоторых случаях, когда *IP*-адрес узла неизвестен, но известно его имя в сети, можно вместо *IP*-адреса указать имя. В этом случае команда **ping** осуществит неявное преобразование адресов и продемонстрирует соответствующий *IP*-адрес (рис. 46).

```

%% Командная строка
Microsoft Windows XP [Версия 5.1.2600]
(C) Корпорация Майкрософт, 1985-2001.
C:\>ping k909

Обмен пакетами с k909 [192.168.1.4] по 32 байт:

Ответ от 192.168.1.4: число байт=32 время<1мс TTL=128
Ответ от 192.168.1.4: число байт=32 время<1мс TTL=128
Ответ от 192.168.1.4: число байт=32 время<1мс TTL=128
Ответ от 192.168.1.4: число байт=32 время<1мс TTL=128

Статистика Ping для 192.168.1.4:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь),
    Приблизительное время приема-передачи в мс:
        Минимальное = 0мсек, Максимальное = 0 мсек, Среднее = 0 мсек

```

Рис. 46. Результат работы команды **ping** с преобразованием адресов

Выяснить *IP*-адрес по имени узла можно также явным образом при помощи команды **nslookup name** (рис. 47).

```

C:\Users\Tujger>nslookup tujger
ТхЕтхЕ: gatewayd.del
Address: 192.168.3.253

Не заслуживающий доверия ответ:
Ь : tujger.del
Address: 192.168.3.116

```

Рис. 47. Результат работы команды **nslookup**

В обоих случаях для возможности определения *IP*-адреса в данной подсети должен быть настроен *DNS*-сервер.

Запомните полученный *IP*-адрес и используйте его при вызове команды **arp**. Эта команда имеет несколько режимов работы, для активации которых используется разный синтаксис. Например, для принудительного добавления *ARP*-записи в локальную таблицу адресов операционной системы используется:

arp -s inet_addr eth_addr ,

где **inet_addr** – *IP*-адрес, а **eth_addr** – *MAC*-адрес, которые нужно сохранить в таблице.

Команда:

arp -d inet_addr ,

наоборот, удаляет запись с указанным *IP*-адресом.

Для выяснения *MAC*-адреса узла используется команда:

arp -a inet_addr

В результате выполнения команды можно увидеть физический адрес целевого узла, а также его тип – динамический (распределяемый службой адресов *DHCP*) или статический (принудительно установленный администратором сети) – рис. 48.

```
C:\>arp -a 192.168.3.116
```

```
Интерфейс: 192.168.2.153 --- 0xd
    адрес в Интернете      Физический адрес      Тип
    192.168.3.116          00-40-f4-ca-fb-e3     динамический
```

Рис. 48. Результат работы команды **arp**

Если команду **arp -a** запустить без указания конкретного адреса, будет выведен список всех *MAC*-адресов, известных операционной системе.

Задание на лабораторную работу

Изучите принципы физической и логической структуризации сетей; получите понятие о сетевых протоколах и уровнях сети; определите логические и аппаратные адреса узлов.

Работа выполняется по вариантам на отдельном компьютере.

Методические указания по выполнению лабораторной работы

1. Изучите теоретическую часть лабораторной работы.
2. Средствами операционной системы *Windows* определите следующие параметры вашего компьютера: *MAC*-адрес; *IP*-адрес.
3. Выберите любой из компьютеров, объединенных локальную сеть учебного компьютерного класса. Определите аппаратный адрес выбранного компьютера.
4. Выполните полный *dump* таблицы адресов. Сравните количество динамических и статических адресов.
5. Создайте в среде моделирования *Packet Tracer* сетевую топологию, изображенную на рис. 49. Диапазон адресов своего варианта определите по шаблону 192.168.*N*.*, где *N* – номер варианта (по списку в журнале).

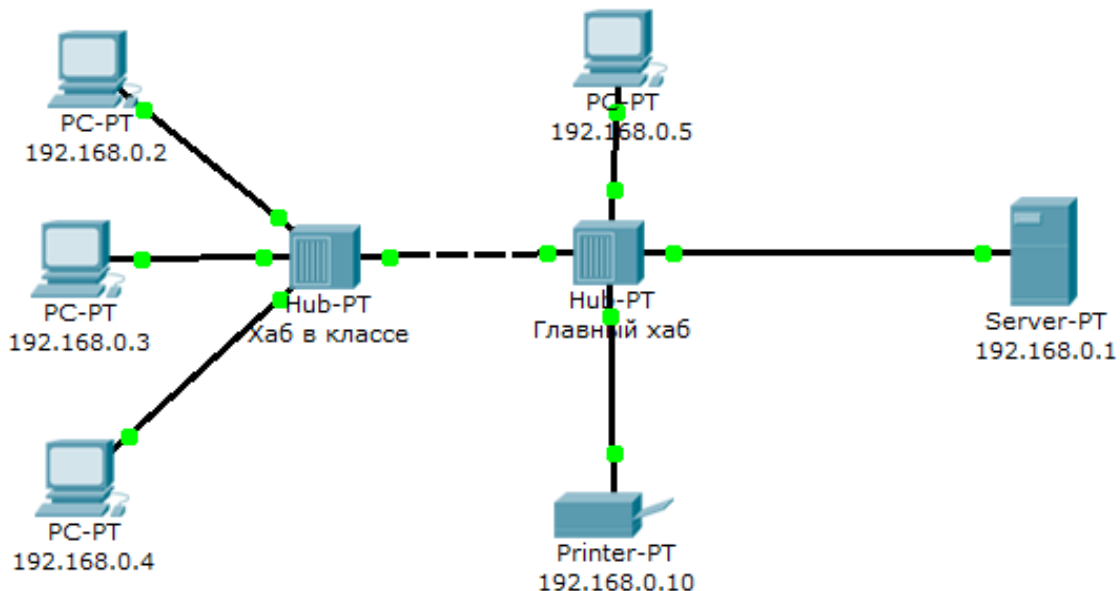


Рис. 49. Простая сеть: два концентратора, четыре компьютера, сервер, принтер

6. Определите *MAC*-адреса всех узлов в этой сети.

Примечание. Обратите внимание: замена *MAC*-адреса сетевого устройства невозможна, поскольку аппаратный адрес записывается в него при изготовлении. Более того, у каждого устройства этот адрес уникален. Однако в некоторых случаях может понадобиться представить это же устройство в сети под другим аппаратным адресом. Как правило, такая возможность имеется. Например, в ОС *Windows* определение *MAC*-адреса осуществляется не прямым обращением к сетевому устройству, а с помощью специальной системной библиотеки. При этом она может вернуть не тот адрес, который действительно *прошит* в сетевом адаптере, а другой, принудительно заданный администратором

компьютера или коммутационного устройства в его настройках. Отсюда вытекает следующее задание: на компьютере найдите способ подменить *MAC*-адрес сетевого адаптера и опишите его.

7. Составьте отчет о выполненной работе. Отчет должен содержать:

- 1) титульный лист с указанием названия лабораторной работы, фамилии студента, номера группы, варианта задания;
- 2) краткое изложение теоретических сведений по теме (2–3 страницы);
- 3) скриншоты экрана с результатами последовательно выполненных заданий (по своему варианту) и поясняющими комментариями к ним;
- 4) ответы на контрольные вопросы;
- 5) общие выводы по работе (заключение).

Контрольные вопросы

1. Объясните, по какой причине физическая топология может не совпадать с логической.
2. Дайте определение термину «сетевой протокол».
3. Поясните, что такое инкапсуляция.
4. Расскажите, чем обусловлены строгие требования к формату и размеру кадра.
5. Назовите два основных метода рассылки сообщений.
6. Объясните, что такое *MAC*-адрес. Откуда он берется?
7. Расскажите, для каких целей может понадобиться подмена *MAC*-адреса устройства.
8. Объясните, что такое *IP*-адрес. Откуда он берется? Из каких частей состоит?
9. Назовите основные уровни сети.
10. Расскажите, что представляет собой концентратор. Каков принцип его работы? Чем отличается концентратор от хаба и коммутатора?
11. Назовите причину, по которой концентраторы не имеют *IP*-адреса.
12. Поясните, что такое коллизия. Что собой представляет домен коллизий? Чем он отличается от домена широковещательной рассылки?
13. Опишите принцип, согласно которому коммутатор распределяет движение пакетов по сети.
14. Откройте анализ *ICMP*-пакета в процессе продвижения по сети, сделанный Вами на предыдущей лабораторной работе. Объясните, с чем связаны изменения *MAC*-адресов, записанных в заголовках кадров.

Лабораторная работа № 4

МАРШРУТИЗАЦИЯ В СЕТЯХ

Цель работы: изучить принципы маршрутизации пакетов в локальных сетях; научиться настраивать простые статические маршруты; ознакомиться с понятием «планирование сети».

Теоретические сведения

Уровень распределения

По мере расширения часто приходится делить одну локальную сеть на несколько сетей уровня доступа. Это можно сделать по-разному, на основе разных критериев, в том числе таких:

- физическое местоположение;
- логическая функция;
- требования безопасности;
- требования приложений.

Уровень распределения соединяет независимые локальные сети и контролирует обмен трафиком (рис. 50). Он отвечает за то, чтобы трафик между узлами локальной сети оставался локальным. Наружу передается только трафик, направленный в другие сети. Кроме того, уровень распределения может фильтровать входящий и исходящий трафик в целях безопасности и управления.

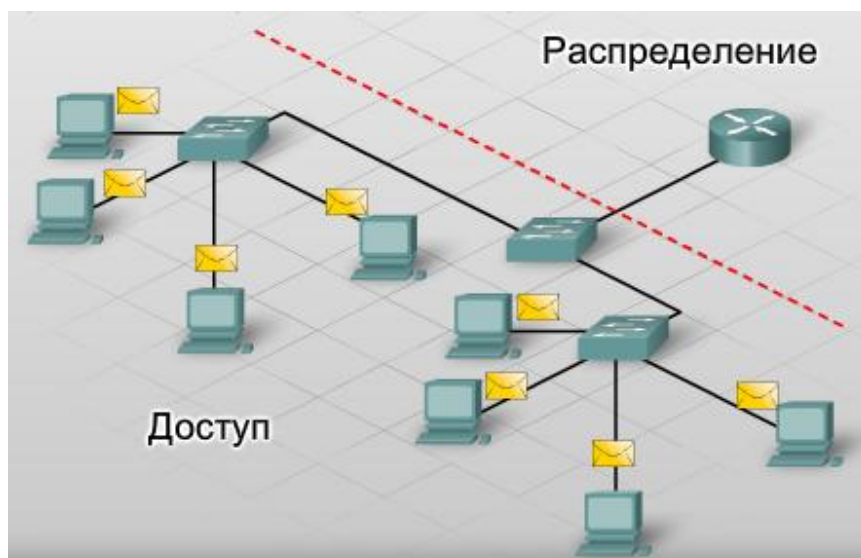


Рис. 50. Уровень распределения

Сетевые устройства уровня распределения призваны связывать не отдельные узлы, а сети. Отдельные узлы подключаются к сети через устройства уровня доступа, например коммутаторы и концентраторы. Устройства уровня доступа связываются друг с другом через устройства уровня распределения (например, маршрутизаторы).

Маршрутизатор

Маршрутизатор (роутер) – сетевое устройство, связывающее локальные сети. На уровне распределения маршрутизаторы направляют трафик и выполняют другие функции, важные для эффективной работы сети. Как и коммутаторы, маршрутизаторы могут декодировать и читать полученные сообщения. Но, в отличие от коммутаторов, которые декодируют только кадр с *MAC*-адресом, маршрутизаторы декодируют весь пакет, находящийся внутри кадра.

В пакете содержатся *IP*-адреса отправителя и адресата и данные пересылаемого сообщения. Маршрутизатор считывает сетевую часть *IP*-адреса назначения и определяет, по какой из подключенных сетей лучше всего переслать сообщение адресату.

Если сетевая часть *IP*-адресов отправителя и адресата не совпадает, для пересылки сообщения необходимо использовать маршрутизатор. Например, если узел, находящийся в сети **1.1.1.0**, отправляет сообщение узлу в сети **5.5.5.0**, то оно переправляется маршрутизатору. Он получает сообщение, распаковывает и считывает *IP*-адрес назначения. Затем он определяет, куда переправить сообщение. Затем маршрутизатор снова инкапсулирует пакет в кадр и переправляет его по назначению.

Каждый порт (интерфейс) маршрутизатора связан со своей локальной сетью. У каждого маршрутизатора есть таблица локально подключенных сетей и их интерфейсов. Кроме того, в этих *таблицах маршрутизации* бывает информация о маршрутах (путях) для подключения к другим локально подключенным удаленным сетям.

Получив кадр, маршрутизатор декодирует его и получает пакет с *IP*-адресом назначения. Этот адрес он сравнивает с данными всех сетей из таблицы маршрутизации. Если адрес сети назначения есть в таблице, маршрутизатор инкапсулирует пакет в новый кадр и отправляет. Новый кадр направляется в сеть назначения через соответствующий порт. Процесс перенаправления пакетов в сеть назначения называется маршрутизацией.

Интерфейсы маршрутизатора не пересылают широковещательные сообщения локальной сети.

Метод, с помощью которого узел отправляет сообщения адресату в удаленной сети, отличается от метода отправки в той же локальной сети. При отправке узлу, подключенному к той же сети, сообщение направляется напрямую, с использованием *MAC*-адреса узла назначения.

С другой стороны, если узлу нужно отправить сообщение в удаленную сеть, приходится использовать маршрутизатор. Узел включает в пакет *IP*-адрес узла назначения. При этом в качестве адреса назначения указывается *MAC*-адрес маршрутизатора. Таким образом, маршрутизатор получает и принимает кадр по *MAC*-адресу.

Каждый узел получает *IP*-адрес маршрутизатора на основе адреса основного шлюза, выбранного в настройках *TCP/IP* (рис. 51). Адрес основного шлюза – это адрес интерфейса маршрутизатора, подключенного к той же локальной сети. Для отправки сообщений маршрутизатору все узлы в локальной сети используют адрес основного шлюза. По нему узел определяет *MAC*-адрес, используя протокол *ARP*.

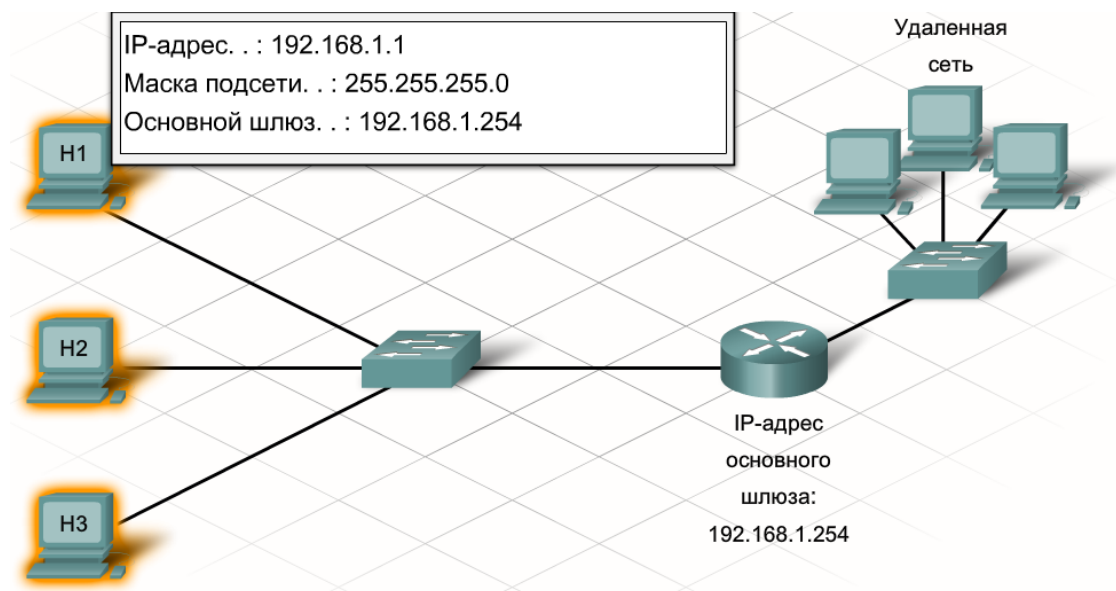


Рис. 51. Маска подсети и адрес шлюза

Для каждого узла в локальной сети важно правильно настроить основной шлюз. Если в настройках *TCP/IP* узла основной шлюз не указан или указан неверно, сообщения, адресованные узлам в удаленных сетях, не будут доставлены.

Таблицы маршрутизации

Таким образом, маршрутизаторы перемещают данные между локальной и удаленной сетью. Информация хранится в таблицах *ARP* и таблицах маршрутизации (рис. 52). В таблицах маршрутизации нет адресов отдельных узлов. В них хранятся адреса сетей и оптимальные пути к ним. Данные вносятся в таблицы маршрутизации двумя способами:

- динамическое обновление данных, полученных от других сетевых маршрутизаторов;
- ручной ввод, выполняемый сетевым администратором.

С помощью таблиц маршрутизаторы определяют порт, на который следует передать пакет (рис. 52).

Тип	Сеть	Порт
C	10.0.0.0/8	FastEthernet0/0
C	172.16.0.0/16	FastEthernet0/1

Рис. 52. Пример таблицы маршрутизации

Если маршрутизатор не может определить адресата сообщения, оно сбрасывается. Чтобы предотвратить сброс, вызванный отсутствием в таблице маршрутизации пути к конкретному адресату, сетевые администраторы вводят в нее маршрут по умолчанию. Он представляет собой порт, через который маршрутизатор передает пакет с неизвестным *IP*-адресом сети назначения. Обычно он ведет к другому маршрутизатору, который может передать пакет в сеть назначения.

Маршрутизатор перенаправляет кадр либо в непосредственно подключенную сеть, где находится узел назначения, либо другому маршрутизатору, который находится на пути к этой сети. Инкапсулируя кадр для отправки через интерфейс *Ethernet*, маршрутизатор должен добавить *MAC*-адрес назначения. В первом случае это будет *MAC*-адрес узла-получателя. Если пакет передается другому маршрутизатору, то будет использован *MAC*-адрес этого маршрутизатора, взятый из таблиц *ARP*. Каждый интерфейс маршрутизатора является частью локальной сети, к которой он подключен, и поддерживает соответствующую таблицу *ARP* для данной сети (рис. 53).

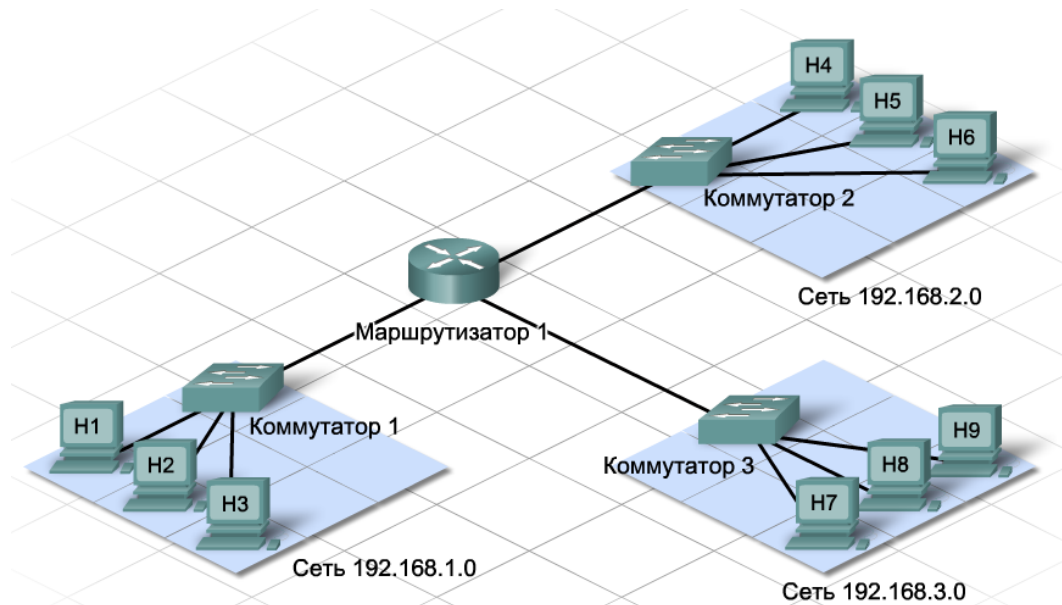


Рис. 53. Маршрутизатор как участник нескольких сетей

Локальная сеть

Термин «локальная сеть» (*Local Area Network – LAN*) относится к группе взаимосвязанных локальных сетей, которыми управляет один и тот же администратор. *LAN* допускает наличие взаимосвязанных сетей, состоящих из сотен узлов, установленных в разных зданиях. Кроме того, обычно в *LAN* используются беспроводные протоколы или *Ethernet* и поддерживается высокая скорость передачи данных.

Частные *LAN*, принадлежащие организации и доступные только для ее членов, сотрудников и прочих допущенных лиц, часто называют *интранет*, или внутренние сети.

В *LAN* все узлы могут находиться в одной локальной сети или же распределяться между несколькими сетями, связанными на уровне распределения. Это зависит от желаемого результата. Если все узлы находятся в одной сети, они образуют один домен широковещательной рассылки и находят друг друга с использованием протокола *ARP*.

При простом проекте сети, возможно, лучше оставить все узлы в одной локальной сети. Однако по мере роста размера сети трафик увеличивается, а эффективность и скорость сети снижаются. В таком случае некоторые узлы стоит переместить в удаленную сеть. Это снизит эффект от увеличения трафика, однако в этом случае узлы из разных сетей не смогут обмениваться данными без использования маршрутизации. Маршрутизаторы усложняют конфигурацию сети и в некоторых случаях создают запаздывание (временные задержки) при обмене пакетами между сетями.

Настройка маршрутизации

Когда маршрутизатор получает пакет, адресованный известному ему узлу, т.е. такому, для которого имеется *ARP*-запись, он просто передает этот пакет данному узлу. В остальных случаях маршрутизатор анализирует адрес сети и отправляет пакет на тот адрес, которому поставлена в соответствие подсеть адресата. При отсутствии такой настройки выдается сообщение об ошибке (рис. 54).

```
PC>ping 192.168.2.4

Pinging 192.168.2.4 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 192.168.2.4:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Рис. 54. Сообщение об ошибке при недоступности сети назначения

Чтобы настроить адресацию между подсетями, на маршрутизаторе необходимо настроить статический маршрут. Для этого в *CLI* маршрутизатора выполните следующую команду:

ip route net_addr net_mask next_hop

Здесь:

- **net_addr** – маска адресов, обращение к которым подлежит маршрутизации;
- **net_mask** – маска подсети, ограничивающая пределы маски адресов;
- **next_hop** – *IP*-адрес, на который маршрутизируются пакеты с данного маршрутизатора.

Например:

ip route 192.168.2.0 255.255.255.0 192.168.10.2

В этом случае все пакеты, адресованные на любой *IP*-адрес из диапазона **192.168.2.1–192.168.2.254**, будут отправлены маршрутизатором на известное ему устройство, имеющее *IP*-адрес **192.168.10.2** (рис. 55).

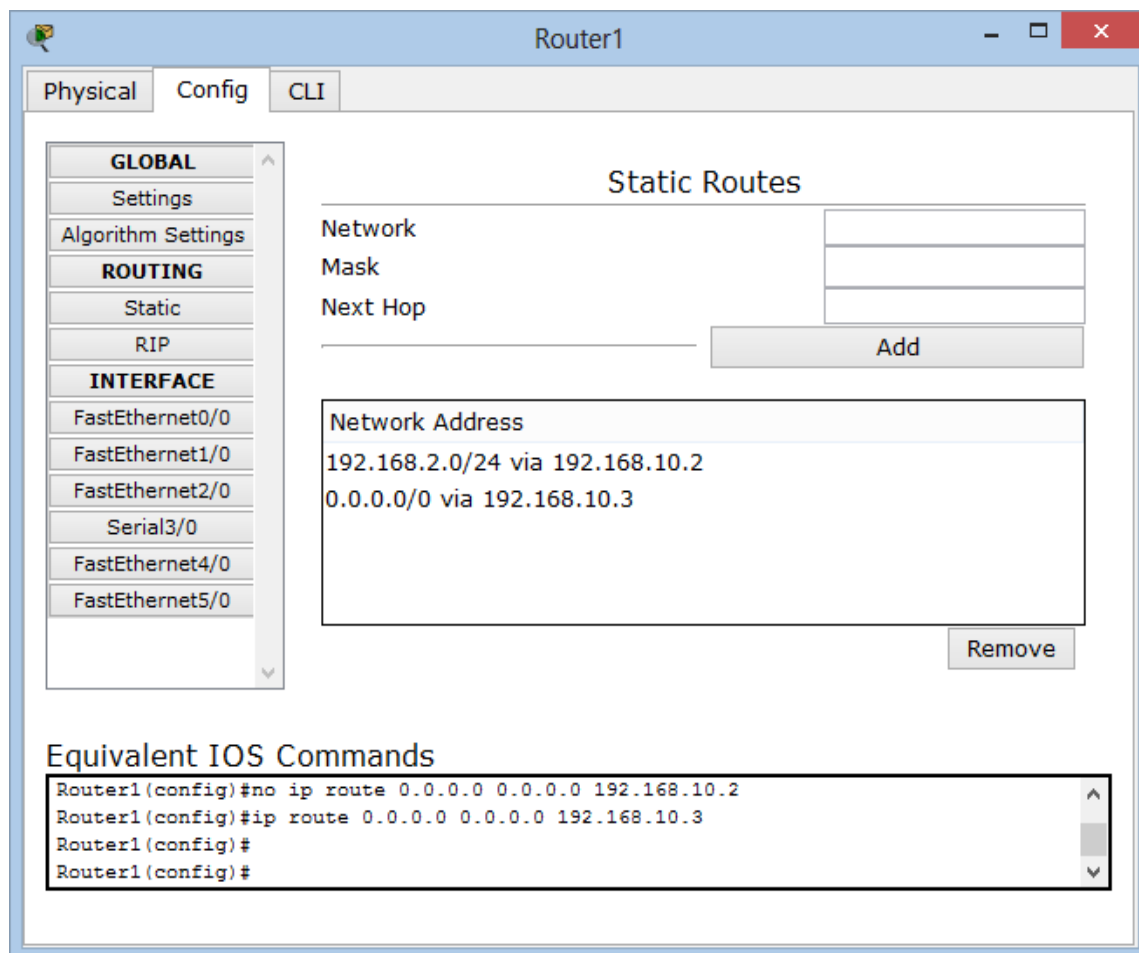


Рис. 55. Окно маршрутизатора с настроенной статической маршрутизацией (вторая запись означает, что все пакеты, не подошедшие под первый шаблон, передаются на адрес **192.168.10.3**)

Если имеется только один возможный путь маршрутизации, то можно указать следующую запись:

ip route 0.0.0.0 0.0.0.0 192.168.10.2

Следует отметить, что вид маски подсети зависит от того, какой диапазон *IP*-адресов задается, а также позволяет ограничить количество адресов, которое маршрутизатору необходимо иметь в виду, чтобы не осуществлять лишние запросы. Различные маски подсети и диапазоны представлены в табл. 4.

Таблица 4

Возможные маски подсети в IPv4

IP/маска	До последнего IP в подсети	Маска подсети	Количество доступных адресов	Класс сети
a.b.c.d/32	+0.0.0.0	255.255.255.255	1	1/256 C
a.b.c.d/31	+0.0.0.1	255.255.255.254	2	1/128 C
a.b.c.d/30	+0.0.0.3	255.255.255.252	4	1/64 C
a.b.c.d/29	+0.0.0.7	255.255.255.248	8	1/32 C
a.b.c.d/28	+0.0.0.15	255.255.255.240	16	1/16 C
a.b.c.d/27	+0.0.0.31	255.255.255.224	32	1/8 C
a.b.c.d/26	+0.0.0.63	255.255.255.192	64	1/4 C
a.b.c.d/25	+0.0.0.127	255.255.255.128	128	1/2 C
a.b.c.0/24	+0.0.0.255	255.255.255.000	256	1 C
a.b.c.0/23	+0.0.1.255	255.255.254.000	512	2 C
a.b.c.0/22	+0.0.3.255	255.255.252.000	1024	4 C
a.b.c.0/21	+0.0.7.255	255.255.248.000	2048	8 C
a.b.c.0/20	+0.0.15.255	255.255.240.000	4096	16 C
a.b.c.0/19	+0.0.31.255	255.255.224.000	8192	32 C
a.b.c.0/18	+0.0.63.255	255.255.192.000	16 384	64 C
a.b.c.0/17	+0.0.127.255	255.255.128.000	32 768	128 C
a.b.0.0/16	+0.0.255.255	255.255.000.000	65 536	256 C = 1 B
a.b.0.0/15	+0.1.255.255	255.254.000.000	131 072	2 B
a.b.0.0/14	+0.3.255.255	255.252.000.000	262 144	4 B
a.b.0.0/13	+0.7.255.255	255.248.000.000	524 288	8 B
a.b.0.0/12	+0.15.255.255	255.240.000.000	1 048 576	16 B
a.b.0.0/11	+0.31.255.255	255.224.000.000	2 097 152	32 B
a.b.0.0/10	+0.63.255.255	255.192.000.000	4 194 304	64 B
a.b.0.0/9	+0.127.255.255	255.128.000.000	8 388 608	128 B
a.0.0.0/8	+0.255.255.255	255.000.000.000	16 777 216	256 B = 1 A
a.0.0.0/7	+1.255.255.255	254.000.000.000	33 554 432	2 A
a.0.0.0/6	+3.255.255.255	252.000.000.000	67 108 864	4 A
a.0.0.0/5	+7.255.255.255	248.000.000.000	134 217 728	8 A
a.0.0.0/4	+15.255.255.255	240.000.000.000	268 435 456	16 A
a.0.0.0/3	+31.255.255.255	224.000.000.000	536 870 912	32 A
a.0.0.0/2	+63.255.255.255	192.000.000.000	1 073 741 824	64 A
a.0.0.0/1	+127.255.255.255	128.000.000.000	2 147 483 648	128 A
0.0.0.0/0	+255.255.255.255	000.000.000.000	4 294 967 296	256 A

Узнать маршрут следования пакета по сети можно при помощи утилиты *TraceRoute*:

tracert inet_addr

Протокол работы утилиты содержит список транзитных адресов и время отклика каждого из них (пинг). При этом максимальное число шагов (хопов) обычно устанавливается равным 30, так как считается, что этого объективно достаточно, чтобы соединить два любых узла в глобальной сети, если их вообще можно соединить. Обычно количество хопов варьируется от 5 до 20 (рис. 56).

```
PC>tracert 192.168.2.4

Tracing route to 192.168.2.4 over a maximum of 30 hops:

  1    10 ms     6 ms     7 ms     192.168.1.1
  2    11 ms     10 ms    12 ms    192.168.10.2
  3    18 ms     19 ms    18 ms    192.168.2.4

Trace complete.
```

Рис. 56. Результат работы утилиты **tracert**

Планирование сети

Большинство локальных сетей создано на основе технологии *Ethernet*. В правильно разработанной и сконструированной сети она работает быстро и эффективно. Основная предпосылка для создания качественной сети – предварительное планирование.

Для начала собирается информация о том, как будет использоваться сеть. Сюда входит:

- количество и тип подключаемых узлов;
- используемые приложения;
- требования к общему доступу и подключению к интернету;
- безопасность и охрана личной информации;
- ожидаемая степень надежности и время бесперебойной работы;
- требования к подключению, в частности выбор проводной или беспроводной связи.

При планировании сети необходимо учесть множество аспектов. Перед покупкой сетевого оборудования и подключением узлов следует построить схемы логической и физической топологии сети. В частности, необходимо учесть:

- физическую среду установки сети;
- контроль температуры, так как у всех устройств есть специфические требования к температуре и влажности;
- наличие и расположение розеток.

С точки зрения физической конфигурации сети учитываются (см. рис. 35):

- физическое расположение устройств;
- соединение устройств;
- расположение и длина всех кабелей;
- аппаратная конфигурация оконечных устройств.

С точки зрения логической конфигурации (см. рис. 36):

- местоположение и размер доменов широковещательных рассылок и коллизий;
- схема *IP*-адресации;
- схема именования;
- конфигурация общего доступа;
- разрешения.

После того, как требования к сети будут задокументированы, а схемы физической и логической топологии построены, следует протестировать конструкцию сети. Один из способов проверки конструкции сети – создание рабочей модели, или прототипа.

По мере увеличения размера и сложности сети возрастает важность использования прототипов. С помощью прототипа сетевой администратор может выяснить, будет ли запланированная сеть работать так, как ожидается, не тратя лишние деньги на оборудование и монтаж. Все аспекты процесса испытания на прототипе тоже должны быть задокументированы.

Существуют различные средства и технологии создания моделей сети, включая установку реального оборудования в лаборатории и моделирование при помощи программных симуляторов, таких как *Cisco Packet Tracer*, *Huawei Enterprise Network Simulation Platform (eNSP)*, *Boson NetSim* или эмуляторов (*GNS3/dynamips*) и др.

Задание на лабораторную работу

Изучите принципы маршрутизации пакетов в локальных сетях; настройте простые статические маршруты; ознакомьтесь с понятием «планирование сети».

Работа выполняется по вариантам на отдельном компьютере.

Методические указания по выполнению лабораторной работы

1. Изучите теоретическую часть лабораторной работы.
2. Запустите на компьютере программный симулятор *Packet Tracer*.
3. Создайте в среде моделирования *Packet Tracer* одноранговую сетевую топологию, изображенную на рис. 57. Диапазон адресов для своего варианта определите по шаблону 192.168.N.*, где N – номер варианта (по списку в журнале).

Пропингуйте с одного из узлов сети все остальные узлы.

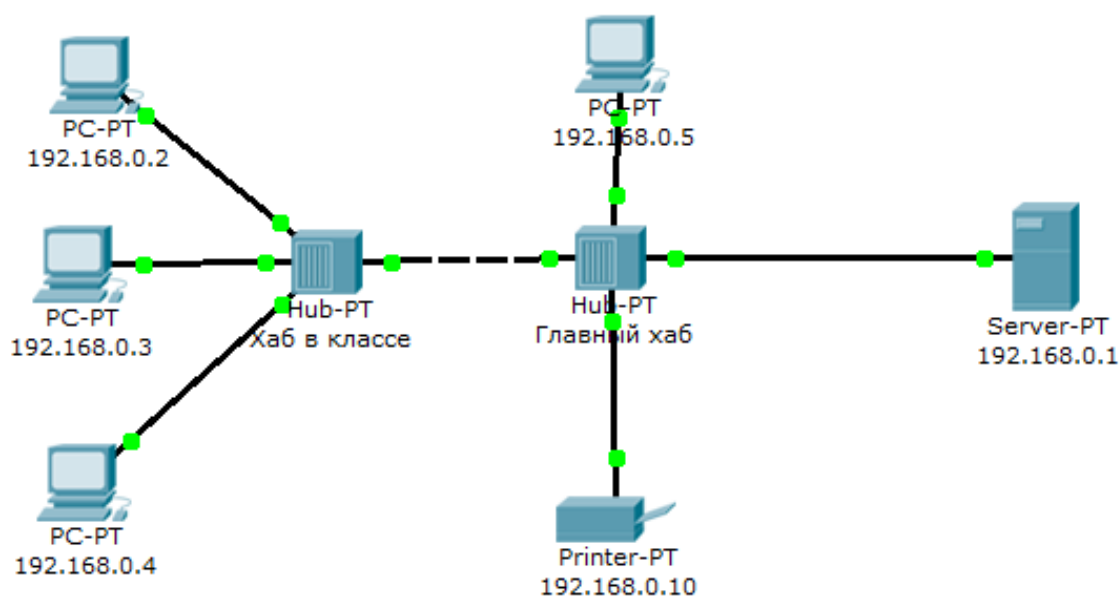


Рис. 57. Простая сеть: два концентратора, четыре компьютера, сервер, принтер

4. Присвойте одному из узлов IP-адрес, который принадлежит адресному пространству другой подсети (например, по формуле **192.168.N + 1.1**), маску подсети при этом не меняйте. Проанализируйте, как изменился обмен данными в сети.

5. Модифицируйте топологию сети в более сложную таким образом, чтобы в ней было две подсети, соединенные через два роутера, как на рис. 58.

Роутеры между собой соедините кросс-кабелем. Пропингуйте с любого одного узла все остальные. Проанализируйте поведение пакетов в случае, когда пинг не проходит.

6 Сконфигурируйте узлы таким образом, чтобы при обозначенных IP-адресах успешно пинговались все устройства. Для этого роутеры объедините в подсеть с маской **255.255.255.252** и определите на каждом из них статическую маршрутизацию.

7. Сделайте *трейсрут* (выполните утилиту *TraceRoute*) с любого одного узла на все остальные. Проанализируйте результат.

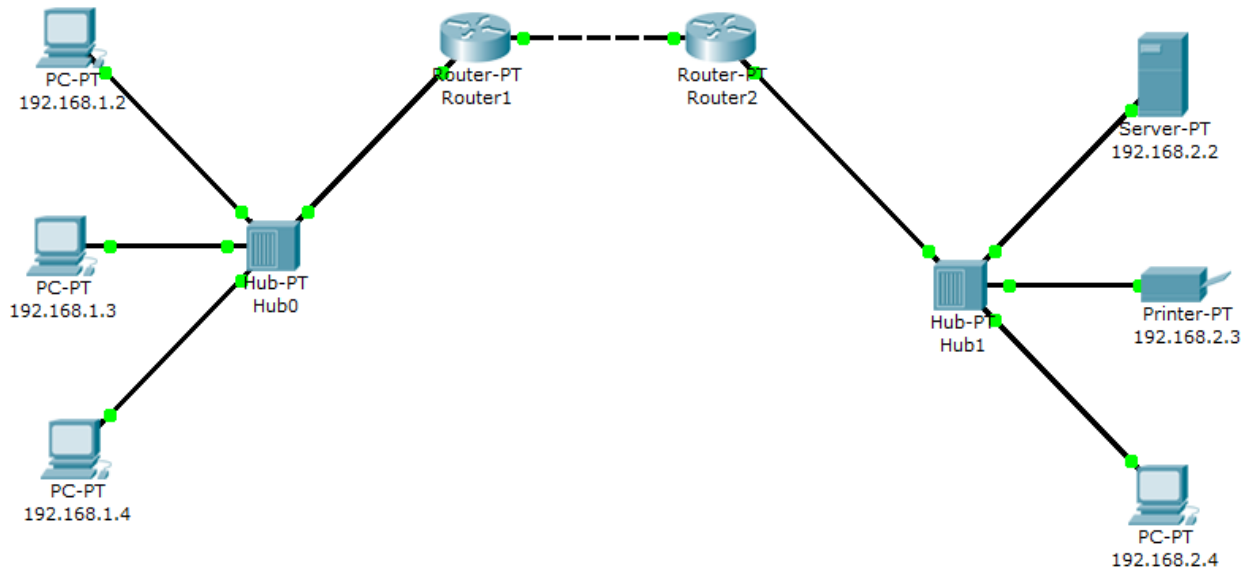


Рис. 58. Сеть: роутер, два концентратора, четыре компьютера, сервер, принтер

8. Сделайте трейсрут на какой-либо из известных вам сайтов в Интернете. Прокомментируйте полученные данные.

9. К уже имеющейся топологии добавьте беспроводную точку доступа. Если на роутере недостаточно портов *FastEthernet*, добавьте их через интерфейс программы. К точке доступа подключите компьютер с *Wi-Fi*-адаптером, как на рис. 59. Сконфигурируйте их так, чтобы сеть функционировала исправно. Проследите это на примере пинга с компьютера, подключенного к беспроводной точке доступа, до любого из остальных компьютеров сети.

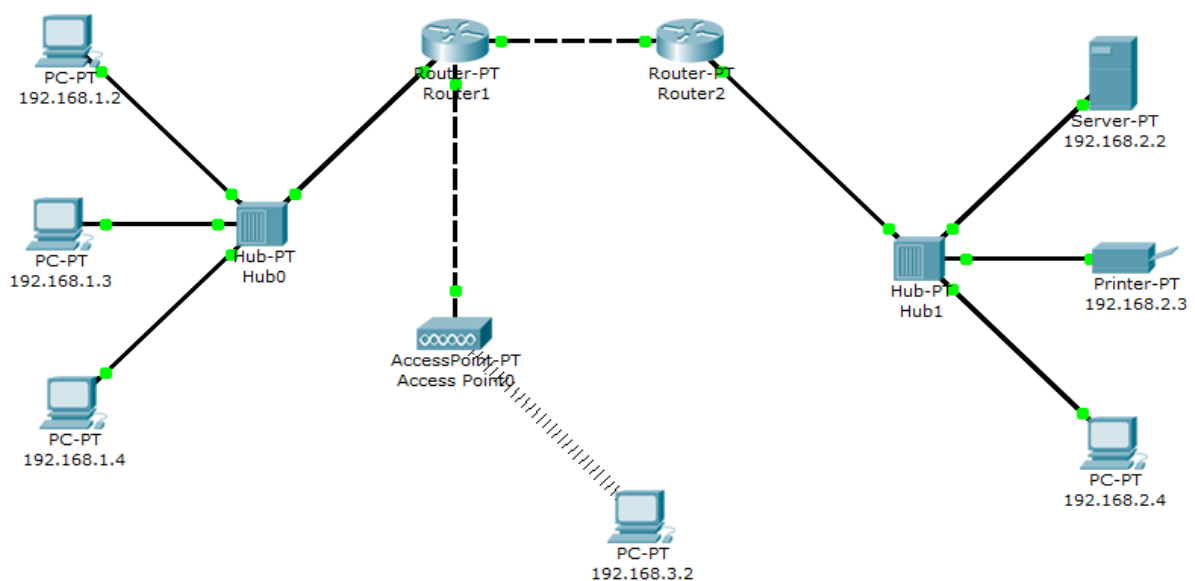


Рис. 59. Сеть: роутер, беспроводная точка доступа, два концентратора, пять компьютеров, сервер, принтер

10. Составьте отчет о выполненной работе. Отчет должен содержать:

- 1) титульный лист с указанием названия лабораторной работы, фамилии студента, номера группы, варианта задания;
- 2) краткое изложение теоретических сведений по теме (2–3 страницы);
- 3) скриншоты экрана с результатами последовательно выполненных заданий (по своему варианту) и поясняющими комментариями к ним;
- 4) ответы на контрольные вопросы;
- 5) общие выводы по работе (заключение).

Контрольные вопросы

1. Объясните, что такое уровень распределения. Какие функции на нем выполняются?
2. Дайте пояснение выражению «локальный трафик».
3. Объясните, что такое роутер. Чем отличается роутер от свитча?
4. Расскажите, как формируется таблица маршрутизации.
5. Объясните, что означают записи: 10.0.0.0/8; 192.168.1.13/32. Что означает маска подсети 255.255.255.252?
6. Поясните, есть ли различие в терминах «локальная сеть» и LAN.
7. Поясните, каким образом маршрутизатор отправляет пакеты узлам в своей подсети, в чужой подсети.
8. Напишите команду для настройки статической маршрутизации. Где необходимо выполнять эту команду?
9. Объясните, зачем нужно ограничивать количество узлов маской подсети.
10. Расскажите, каким образом можно проследить путь пакета до узла назначения.
11. Расскажите, с какой целью и почему необходимо осуществлять планирование сети.
12. Приведите последовательность этапов при планировании сети. Что при этом необходимо учесть?
13. Расскажите, что является главным критерием планирования сети.

СПИСОК ЛИТЕРАТУРЫ

1. Программа сетевой академии Cisco CCNA 1 и 2: вспомогательное руководство. – М.: Вильямс, 2007. – 1168 с.
2. Программа сетевой академии Cisco CCNA 3 и 4: вспомогательное руководство. – М.: Вильямс, 2007. – 944 с.
3. Мухутдинов Э.А. Мировые информационные ресурсы и сети: учеб. пособие / Э.А. Мухутдинов, С.Ю. Ситников, Е.А. Комиссарова. – Казань: КГЭУ, 2009. – 230 с.
4. Олифер В.Г. Компьютерные сети: Принципы, технологии, протоколы: учебник для вузов / В.Г. Олифер, Н.А. Олифер. – СПб.: Питер, 2016. – 992 с.
5. Олифер В.Г. Сетевые операционные системы: учебник для вузов / В.Г. Олифер, Н.А. Олифер. – СПб.: Питер, 2009. – 672 с.
6. Олифер В.Г. Основы компьютерных сетей: учеб. пособие / В.Г. Олифер, Н.А. Олифер. – СПб.: Питер, 2009. – 350 с.
7. Малюк А.А. Введение в информационную безопасность: учеб. пособие для вузов / А.А. Малюк, В.С. Горбатов, А.П. Дураковский и др. – М.: Горячая линия – Телеком, 2011. – 288 с.
8. Бройдо В.Л. Вычислительные системы, сети и телекоммуникации: учебник для вузов / В.Л. Бройдо, О.П. Ильина. – СПб.: Питер, 2011. – 560 с.
9. Пескова С.А. Сети и телекоммуникации: учебник / С.А. Пескова, А.В. Кузин, А.Н. Волков. – М.: Академия, 2009. – 352 с.
10. Таненбаум Э. Компьютерные сети / Э. Таненбаум, Д. Уэзеролл. – СПб.: Питер, 2013. – 960 с.
11. Насейкина Л.Ф. Сети и телекоммуникации: учеб. пособие: [Электрон. ресурс] / Л.Ф. Насейкина, Н.В. Деревянкина. – Оренбург, 2013. – Режим доступа: <http://11povt-osu.16mb.com/seti/> (дата обращения 31.05.2017).

СОДЕРЖАНИЕ

Введение	3
Лабораторная работа № 1. Знакомство с программной средой моделирования компьютерных сетей	4
Лабораторная работа № 2. Анализ функционирования компьютерной сети	19
Лабораторная работа № 3. Физическая и логическая структуризация сети. Протоколы, уровни, адреса	35
Лабораторная работа № 4. Маршрутизация в сетях	54
Список литературы	67

Учебное издание

**Ситников Сергей Юрьевич,
Ситников Юрий Кириллович,
Мухутдинов Эдуард Асгатович**

**ИНФОРМАЦИОННЫЕ СИСТЕМЫ И СЕТИ.
ЧАСТЬ I. ОСНОВЫ КОМПЬЮТЕРНЫХ СЕТЕЙ**

Лабораторный практикум

Кафедра информатики и информационно-управляющих систем КГЭУ

**Редактор издательского отдела *Н.А. Мустакимова*
Компьютерная верстка *Т.И. Лунченкова***

Подписано в печать 06.06.2017.

Формат 60х84/16. Гарнитура «Times». Вид печати РОМ.

Усл. печ. л. 3,95. Уч.-изд. л. 4,38. Тираж 500 экз. Заказ № 126/эл.

**Редакционно-издательский отдел КГЭУ
420066, Казань, Красносельская, 51**