

## Лабораторная работа

### Моделирование сетей связи с пакетной коммутацией

#### Цель работы

Исследовать процедуры моделирования сетей связи с пакетной коммутацией на примере программы Packet Tracer фирмы Cisco Systems.

#### Краткие теоретические сведения

Packet Tracer — симулятор сети передачи данных, выпускаемый фирмой Cisco Systems. Позволяет делать работоспособные модели сети, настраивать (командами Cisco IOS) маршрутизаторы и коммутаторы, взаимодействовать между несколькими пользователями (через облако). Включает в себя серии маршрутизаторов Cisco 1800, 2600, 2800 и коммутаторов 2950, 2960, 3560. Кроме того есть серверы DHCP, HTTP, TFTP, FTP, рабочие станции, различные модули к компьютерам и маршрутизаторам, устройства WiFi, различные кабели.

#### Оборудование рабочего места

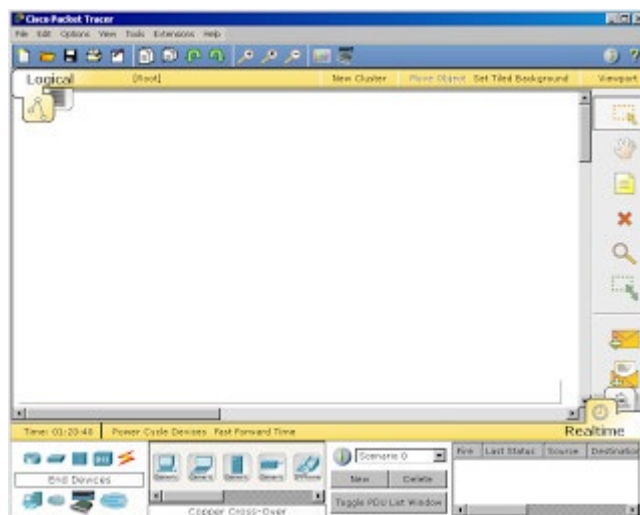
Интерактивный учебно-лабораторный класс (ауд. 217).

Программное обеспечение Packet Tracer

#### Задание

##### Часть 1

После запуска симулятора Packet Tracer, перед нами появляется довольно-таки приятное на вид главное окно программы.



## Главное окно Packet Tracer

Большую часть данного окна занимает рабочая область, в которой можно размещать различные сетевые устройства, соединять их различными способами и как следствие получать самые разные сетевые топологии.

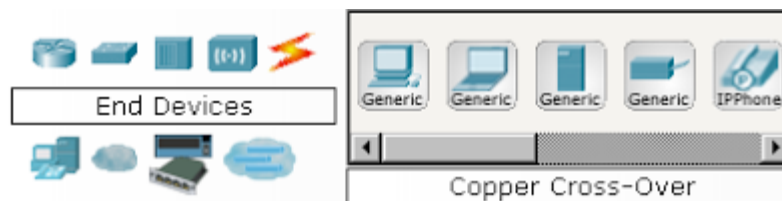
Сверху, над рабочей областью, расположена главная панель программы и ее меню. Меню позволяет выполнять сохранение, загрузку сетевых топологий, настройку симуляции, а также много других интересных функций. Главная панель содержит на себе наиболее часто используемые функции меню.



## Главное меню Packet Tracer

Справа от рабочей области, расположена боковая панель, содержащая ряд кнопок отвечающих за перемещение полотна рабочей области, удаление объектов и т.д.

Снизу, под рабочей областью, расположена панель оборудования.



## Панель оборудования Packet Tracer

Данная панель содержит в своей левой части типы доступных устройств, а в правой части доступные модели. При выполнении различных лабораторных работ, эту панель придется использовать намного чаще, чем все остальные. Поэтому рассмотрим ее более подробно.

При наведении на каждое из устройств, в прямоугольнике, находящемся в центре между ними будет отображаться его тип. Типы устройств, наиболее часто используемые в лабораторных работах Packet Tracer, представлены на рисунке.



### Основные типы устройств

Рассматривать конкретные модели устройств каждого типа, не имеет большого смысла. Так как их модель «красным по белому» написана прямо под иконкой каждого из устройств. Отдельного рассмотрения заслуживают типы соединений. Перечислим наиболее часто используемые из них (рассмотрение типов подключений идет слева направо, в соответствии с приведенным снизу рисунком).



### Типы соединений устройств в Packet Tracer

- Автоматический тип – при данном типе соединения PacketTracer автоматически выбирает наиболее предпочтительные тип соединения для выбранных устройств
- Консоль – консольные соединения
- Медь Прямое – соединение медным кабелем типа витая пара, оба конца кабеля обжаты в одинаковой раскладке. Подойдет для следующих соединений: коммутатор – коммутатор, коммутатор – маршрутизатор, коммутатор – компьютер и др.
- Медь кроссовер – соединение медным кабелем типа витая пара, концы кабеля обжаты как кроссовер. Подойдет для соединения двух компьютеров.
- Оптика – соединение при помощи оптического кабеля, необходимо для соединения устройств имеющих оптические интерфейсы.

- Телефонный кабель – обыкновенный телефонный кабель, может понадобиться для подключения телефонных аппаратов.
- Коаксиальный кабель – соединение устройств с помощью коаксиального кабеля.

Попробуем собрать простенькую схему, симулирующую подключения компьютеров к коммутаторам, используя PacketTracer. Для этого:

- Кликните левой кнопкой мыши на значок коммутаторы в панели устройств



Значок коммутатора на панели оборудования  
Packet Tracer

- Кликните левой кнопкой мыши на значок конкурентной модели коммутатора, например 2960



Значок конкретной модели коммутатора

- После этого кликните левой кнопкой мыши в рабочей области программы, там должен появиться коммутатор.



Коммутатора в рабочей области Packet Tracer

- Повторите, приведенную выше, последовательность действий и разместите в рабочей области еще один коммутатор.
- После этого попытаемся их соединить, для этого щелкните левой кнопкой мыши на значок типа соединения в панели устройств.



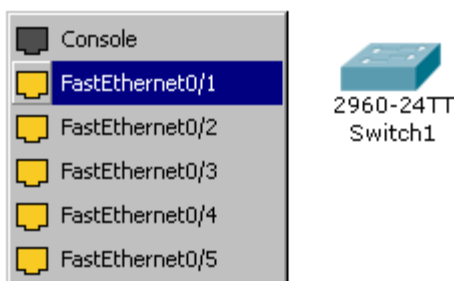
Значок типа соединения на  
панели оборудования

- Кликните левой кнопкой мыши по иконке типа соединения – медь прямое.



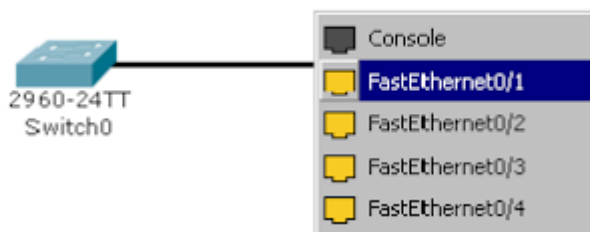
Значок Прямое соединения витой парой

- Кликните левой кнопкой мыши по первому коммутатору. Откроется окно выбора сетевого интерфейса (номера порта коммутатора) к которому вы хотите подключить кабель. Выберите один из них и кликните левой кнопкой мыши по его названию.



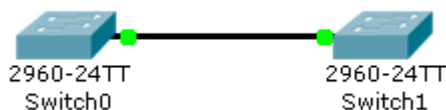
Выбираем интерфейс для соединения на первом коммутаторе

- После этого кликните левой кнопкой мыши по второму коммутатору, и аналогичным образом выберите интерфейс для подключения на нем.



Выбираем интерфейс на втором коммутаторе

- После выполнения данных действий у вас должна получиться топология представленная на рисунке.



## Два коммутатора, соединенные между собой витой парой

- Далее подключим к каждому из коммутаторов по компьютеру. Для этого кликните мышкой по иконке оконечное оборудование в панели устройств.



### Иконка Оконечное оборудование

- Выберите тип оконечного оборудования как персональный компьютер.



### Иконка персональный компьютер

- Разместите персональный компьютер в рабочей области, и подсоедините его к одному из коммутаторов (Соединение компьютер – коммутатор, выполняется аналогично соединению коммутатор-коммутатор).
- Разместите в рабочей области еще один персональный компьютер, но подключите его к другому коммутатору. В результате у вас должна получиться сетевая топология, представленная на рисунке.

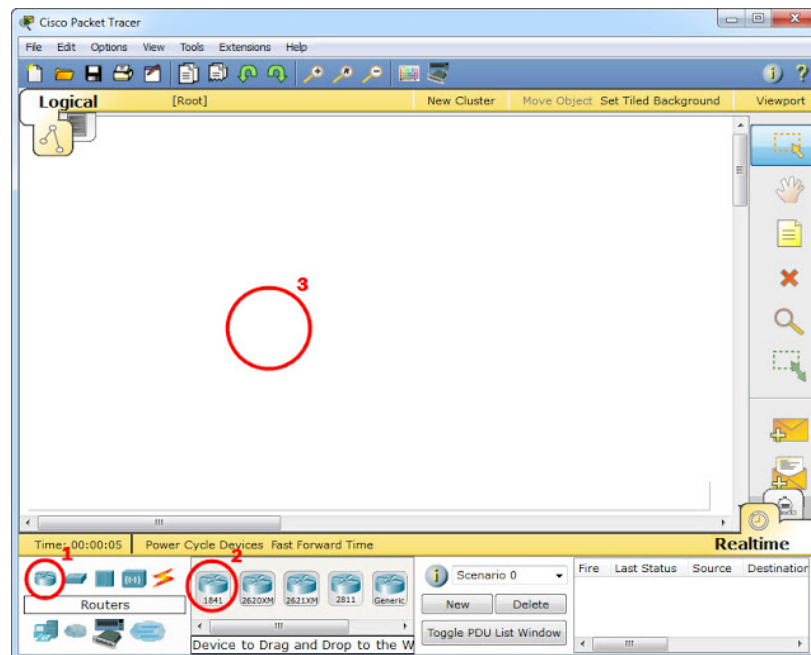


## Пример сетевой топологии в Packet Tracer

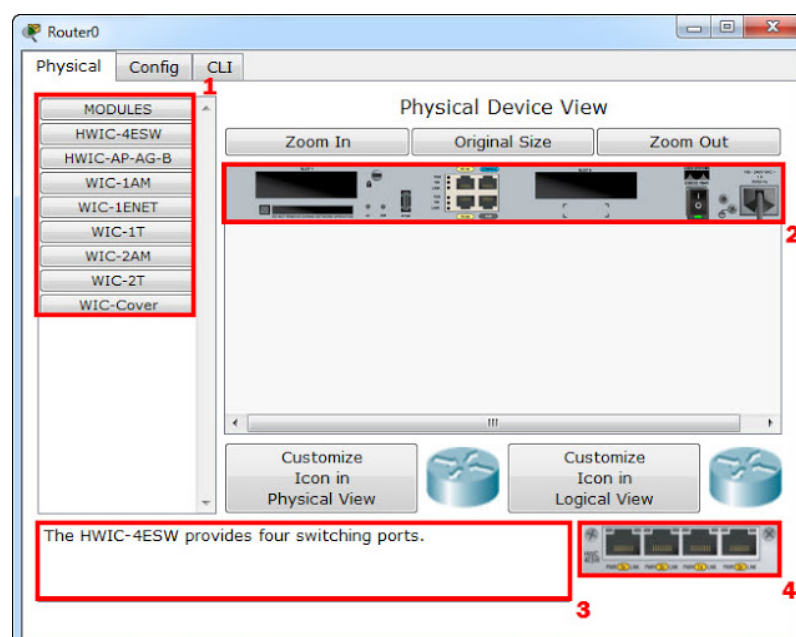
### Часть 2

#### Добавление маршрутизатора в проект

Для того чтобы добавить маршрутизатор в проект сети, необходимо выбрать левым кликом мыши данный тип оборудования, также выбрать модель и добавить в проект, кликнув на рабочем поле программы. Весь процесс добавления нового юнита делается в три клика.

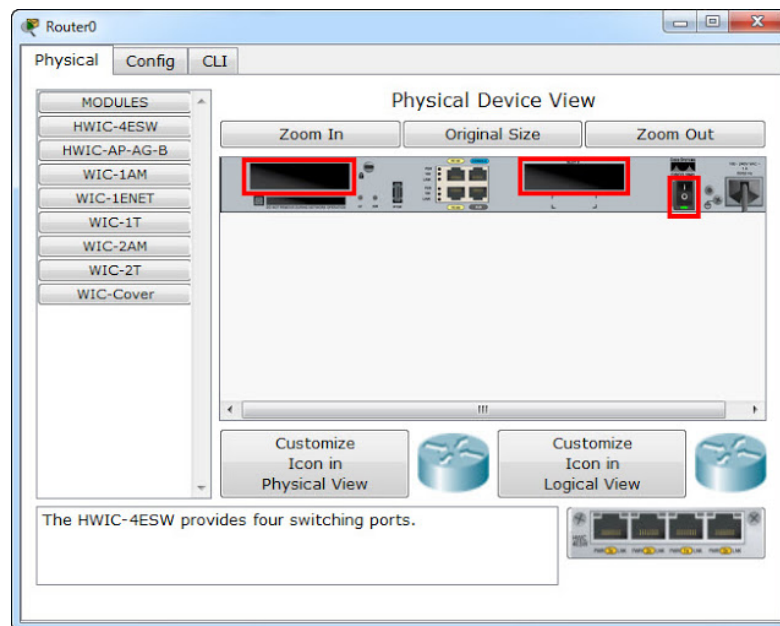


После того как оборудование добавлено в проект, можно открыть окно параметров данного устройства которое предоставляет возможность доступа к аппаратной конфигурации данного юнита, а также возможность конфигурации средствами IOS CLI или меню. Так как знание команд CLI является необходимым для успешной сдачи CCNA, возможности управления конфигурацией маршрутизаторов и коммутаторов средствами меню лучше не использовать.



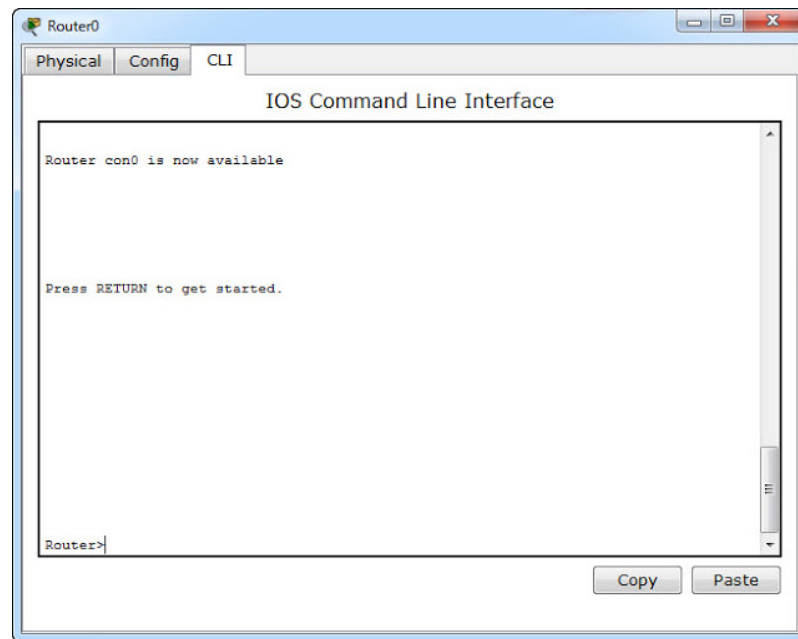
Активная по умолчанию вкладка "Physical" позволяет управлять аппаратной конфигурацией маршрутизатора и отображает следующие элементы:

- список доступных для установки модулей;
- внешний вид оборудования;
- описание выбранного модуля;
- внешний вид выбранного модуля.



В тех случаях, когда для выполнения лабораторной работы нужен, отсутствующий по умолчанию, интерфейс, данная вкладка позволяет добавить любой из доступных. Кроме того, свободные слоты могут быть закрыты фальшь панелями ( WIC-Cover). Любые операции по добавлению или удалению модулей производятся только на выключенном оборудовании. В связи с этим, виртуальное оборудование должно быть отключено от сети кнопкой питания перед установкой или удалением сетевого модуля.



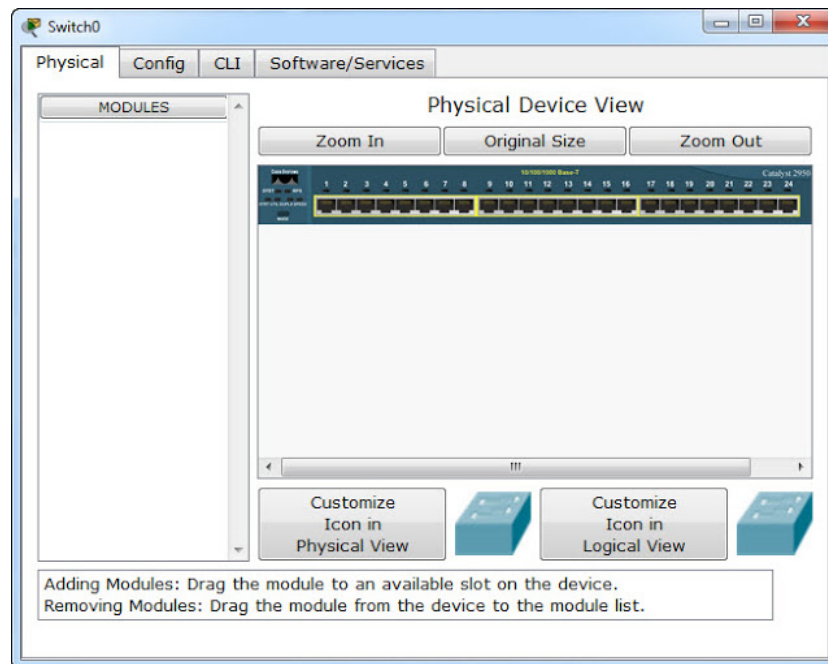


Вкладка CLI предоставляет доступ к консоли Console 0 маршрутизатора. По умолчанию на доступ к консоли пароль не установлен.

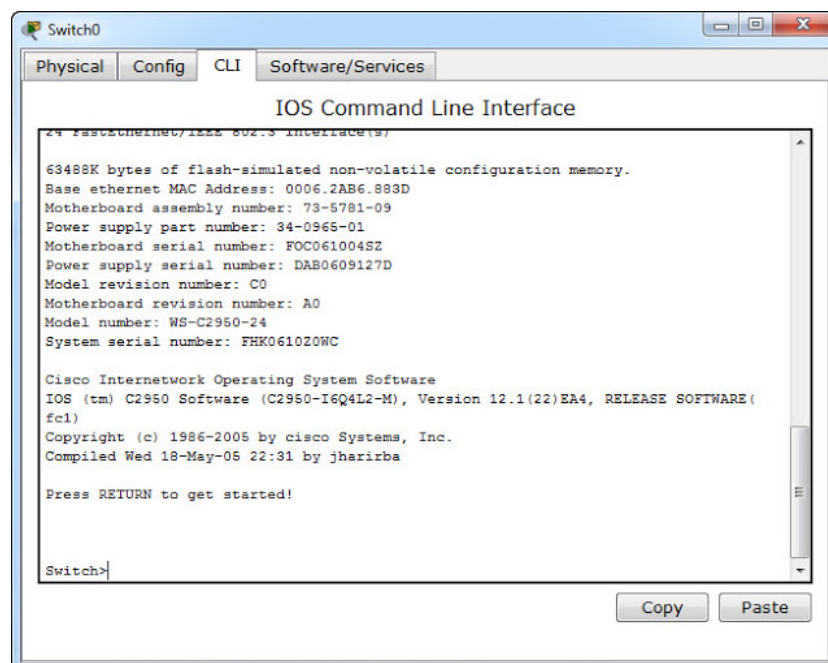
### Добавление коммутатора в проект

Добавление коммутатора в проект практически идентично добавлению маршрутизатора. Отличия в начальной стадии при выборе панели и в самом окне параметров добавленного устройства.





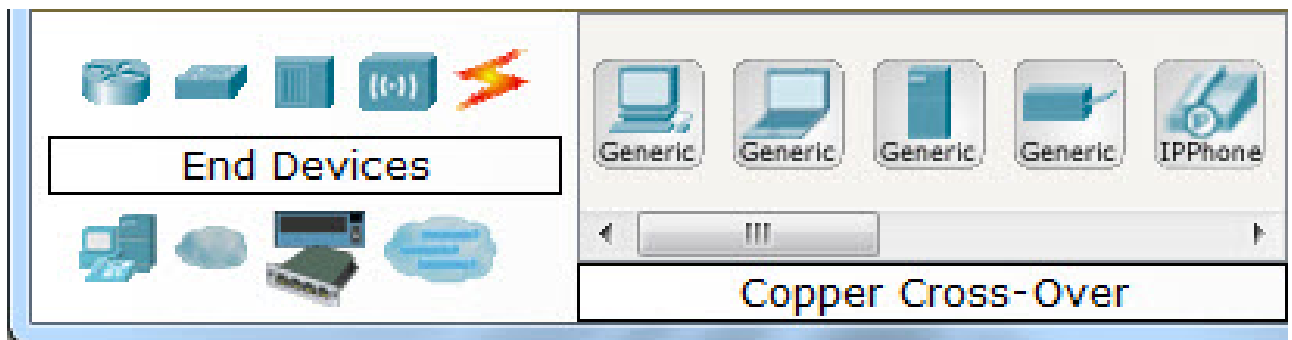
В параметрах отсутствует возможность изменения аппаратной конфигурации оборудования, так как в большинстве своем коммутаторы, доступные в Cisco Packet Tracer, не являются модульными. Исключение составляет специальный тип оборудования Genetec, который практически не используется или используется редко.



Так же как и для маршрутизаторов, вкладка CLI предоставляет доступ к консоли Console 0. По умолчанию на доступ к консоли пароль не установлен.

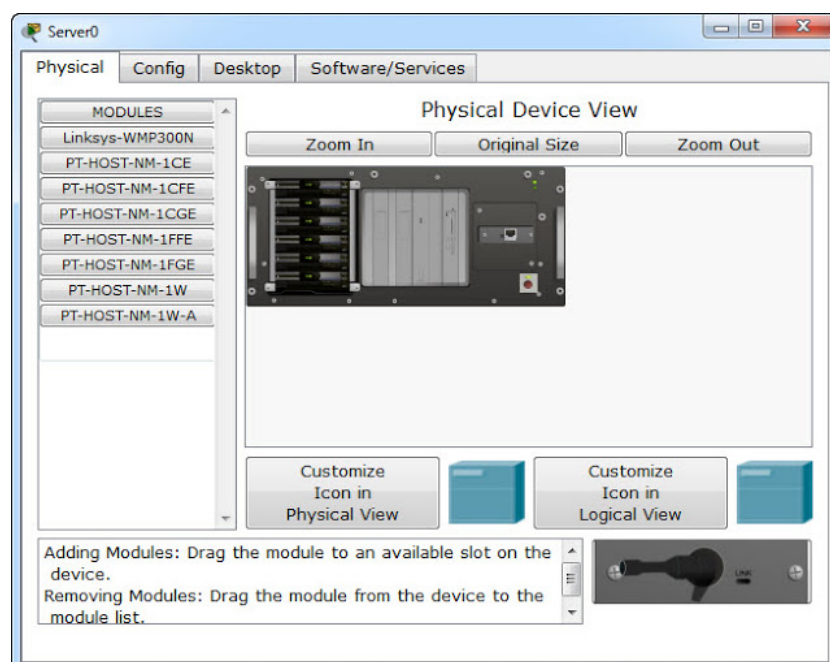
### Добавление конечных узлов сети

Конечные узлы сети, такие как сервера, рабочие станции и ноутбуки, добавляются в топологию идентично другим устройствам проекта.

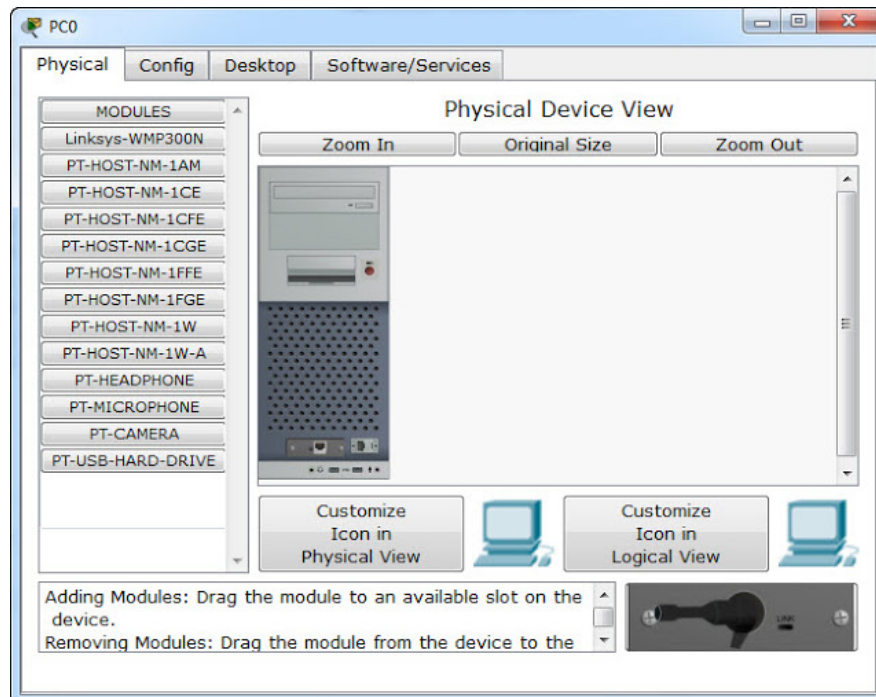


Окно параметров конечных устройств варьируется в зависимости от самого устройства. Некоторые различия сразу заметны .

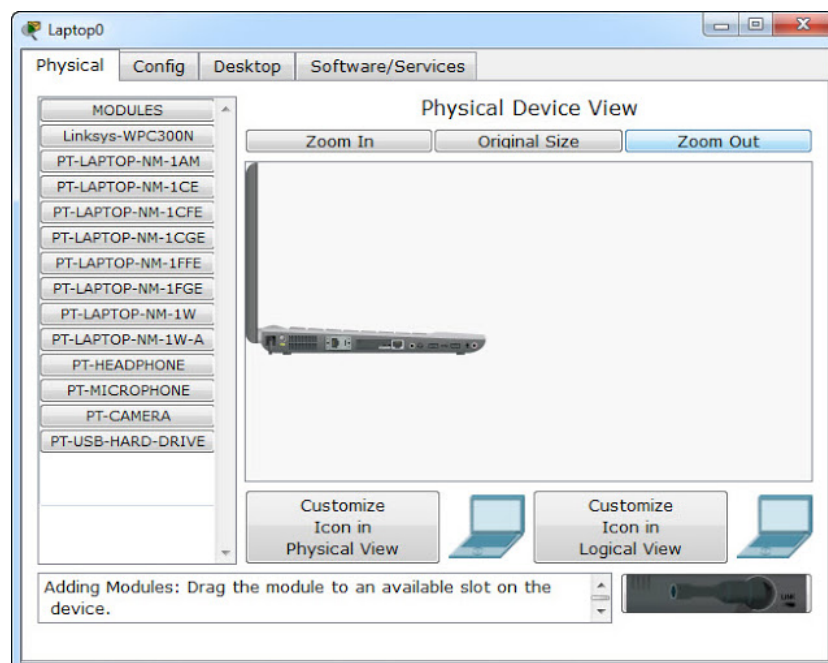
### Сервер



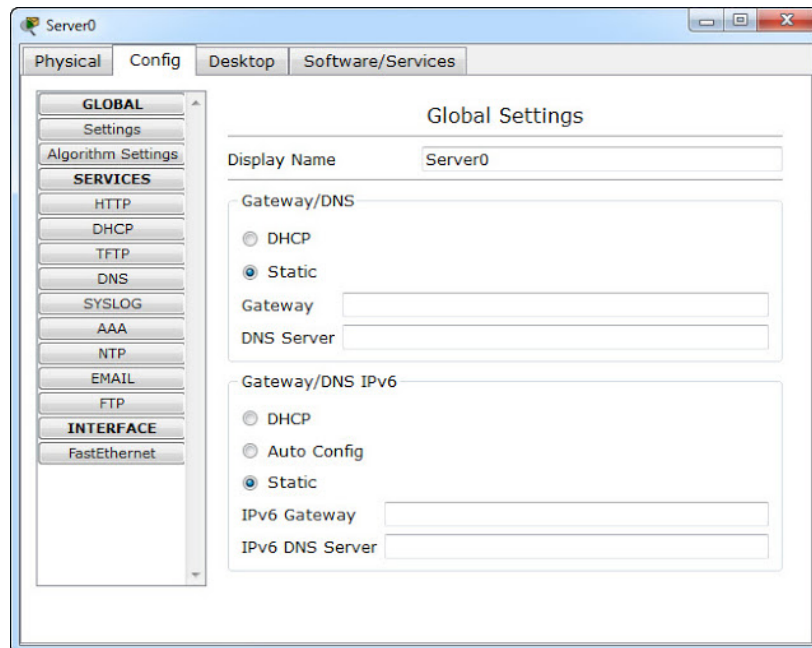
## Персональный компьютер



## Лэптоп

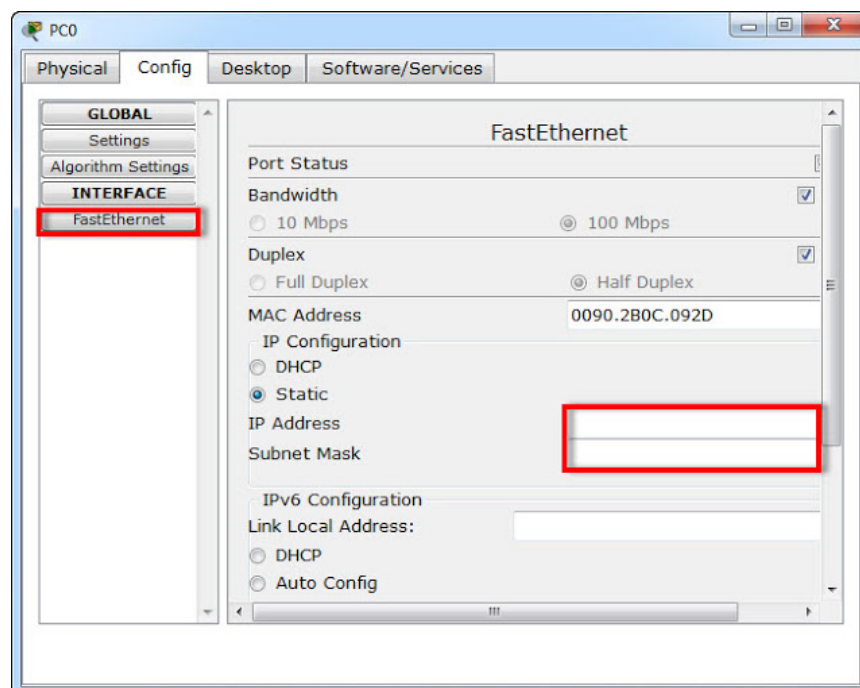
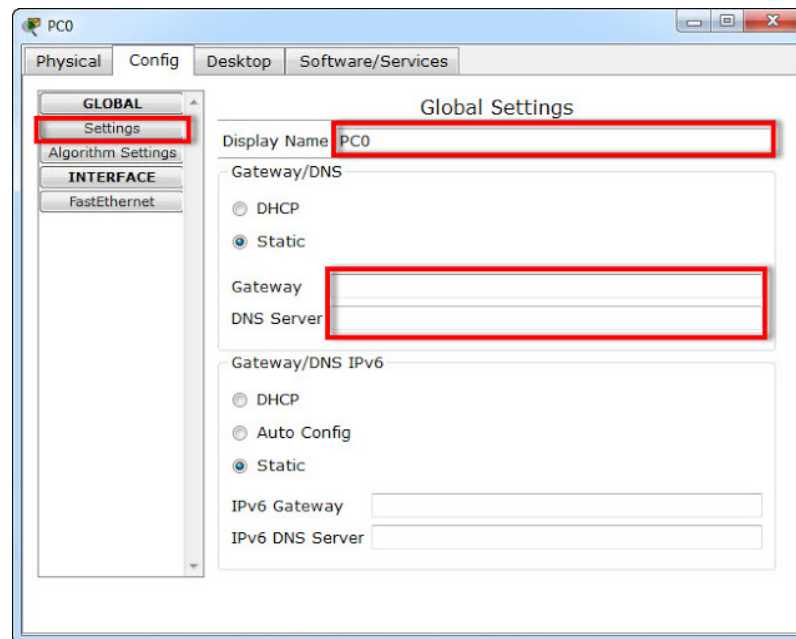


Параметры сетевых интерфейсов для данного вида устройств устанавливаются через меню на вкладке Config. Данная вкладка практически идентична для перечисленных выше устройств.



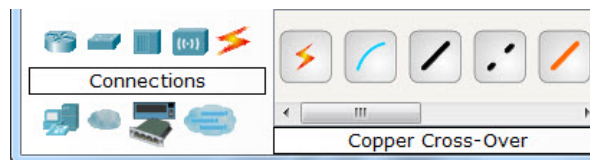
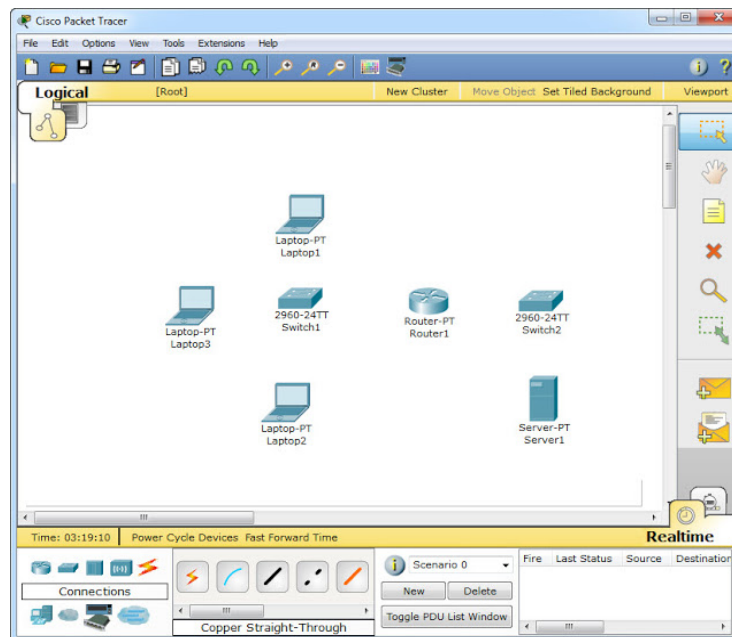
Исключение составляет наличие дополнительных меню настройки сетевых сервисов ( HTTP, DHCP, TFTP, DNS, SYSLOG, AAA, NTP, EMAIL, FTP ) у устройства сервис. Данные сервисы далеки по своей функциональности от реальных, но обеспечивают базовую функциональность, необходимую для тестирования.

Сетевые параметры указываются в меню Settings и меню свойств сетевого интерфейса (FastEthernet) вкладки Config.

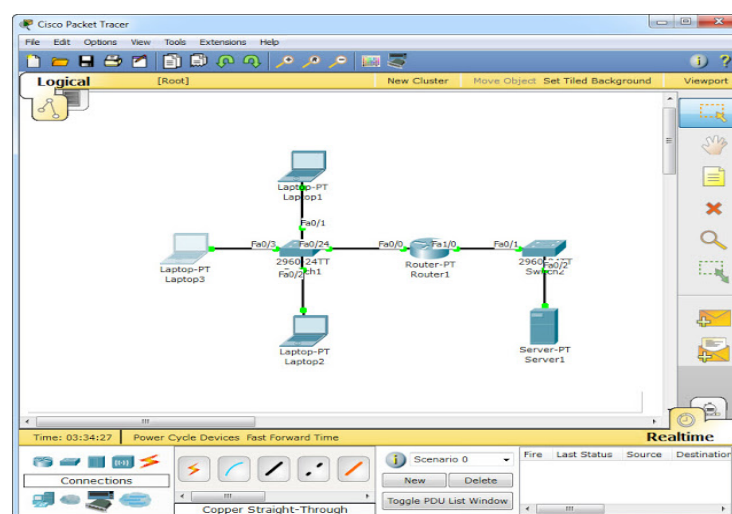


## Соединение устройств

После того как все необходимые для выбранного сценария лабораторной работы устройства добавлены в проект, необходимо все единицы оборудования соединить между собой согласно сценария. Для этого используется меню Conenctions.



Выбор кабеля зависит от подключаемого оборудования и технологии подключения. В этом конкретном случае это будет Copper Cross-Over. Каждый раз при соединении оборудования будет предлагаться выбор интерфейса, если таковые есть в наличии и не участвуют в другом подключении. В конечном итоге получается следующий проект сети.



## **Содержание и оформление отчета**

Отчет по данной работе не оформляется, так как все отчеты сохраняются на сервере.

## **Порядок проведения работы**

После проведения всех исследований, данный факт регистрируется преподавателем. По окончании занятия преподаватель сохраняет результаты работы студента в файл. Если студент не успел за время занятия выполнить все задания, он выполняет их самостоятельно во внеучебное время.

## **Информационные источники**

Техническое описание Cisco Packet Tracer

---



## Лабораторная работа № 1

**Цель работы.** Изучить основные функциональные возможности программного сетевого эмулятора Packet Tracer Cisco Systems.

### Введение

Packet Tracer является интегрированной средой моделирования, визуализации, совместной работы и оценки состояния окружающей среды. Packet Tracer помогает студенту и преподавателю создавать сетевые модели, осуществлять визуализацию и анимацию передачи информации в сети. Как и любое моделирование, Packet Tracer опирается на упрощенные модели сетевых устройств и протоколов. Реальные компьютерные сети остаются эталоном для понимания поведения сети и развития навыков для их построения. Packet Tracer был создан, чтобы помочь решить проблему обеспечения доступа к сетевому оборудованию для студентов и преподавателей.

#### Наименование

#### Описание

**LAN:** Ethernet (including CSMA/CD\*), 802.11 a/b/g/n wireless\*, PPPOE

**Switching:** VLANs, 802.1q, trunking, VTP, DTP, STP\*, RSTP\*, multilayer switching\*, Etherchannel, LACP, PAgP

**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)

**Routing:** static, default, RIPv1, RIPv2, EIGRP, single-area OSPF, multi-area OSPF, BGP, inter-VLAN routing, redistribution

#### Протоколы

**Other:** ACLs (standard, extended, and named), CDP, NAT (static, dynamic, inside/outside, and overload), NATv6

**WAN:** HDLC, SLARP, PPP\*, and Frame Relay\*

**Security:** IPsec, GRE, ISAKMP, NTP, AAA, RADIUS, TACACS, SNMP, SSH, SYSLOG, CBAC, Zone-based policy firewall, IPS

**QoS:** Layer 2 QoS, Layer 3 Diffserv QoS, FIFO Hardware queues, Priority Queuing, Custom Queuing, Weighted Fair Queuing, MQC, NBAR\*

**\* обозначает наложенные функциональные ограничения**  
Создание сетевой топологии

#### Логическое пространство

Устройства: 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.

Обмен данными происходит в режиме реального времени.

## Режим реального времени

Настраиваемая конфигурация: DHCP, DNS, HTTP, TFTP, Syslog, AAA, and NTP servers

Анимация передачи пакетов

Лист событий

## Режим симуляции

Имеется широкий выбор протоколов модели OSI.

Пользователь имеет возможность создания сценария передачи пакетов.

## Логическое пространство

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

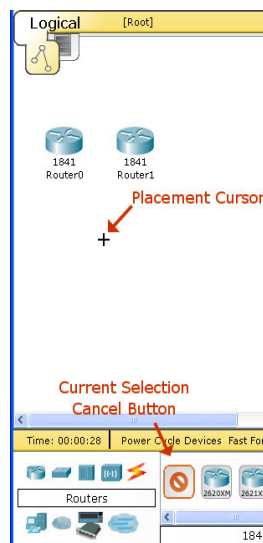


Рис.1. Выбор устройства

Большинство из устройств в Packet Tracer имеют модули расширения, необходимые для подключения дополнительных портов. Добавление модулей осуществляется в панели настройки устройства. При подключении нового модуля устройство должно быть отключено от электросети.

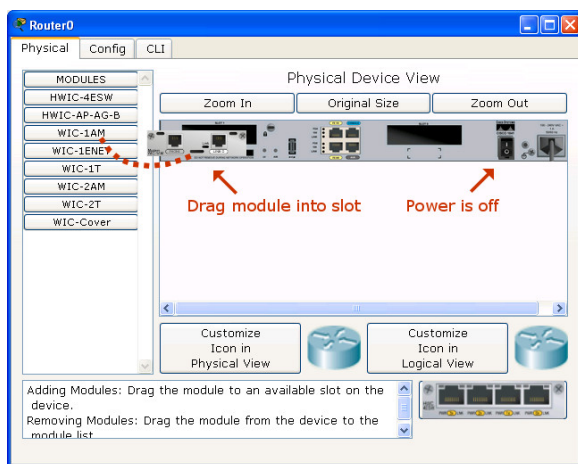


Рис.2. Добавление модулей расширения

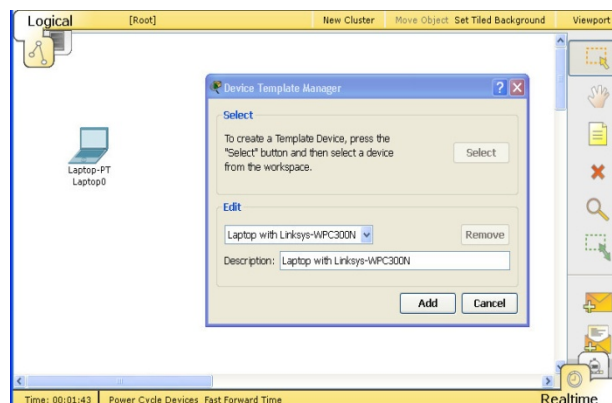


Рис.3. Создание шаблона устройства.

Packet Tracer предоставляет возможность создания шаблонов устройств. Для создания шаблона необходимо выбрать устройство, добавить необходимые модули расширения, затем перейти в **Custom Devices Dialog**. Затем добавить описание выбранного устройства, нажав на **Select**. Добавить новое созданное пользователем устройство можно через **Custom Made Devices**.

Для соединения устройств между собой необходимо выбрать подходящие кабели, расположенных на панели **Connections**. Затем нужно щелкнуть правой кнопкой мыши по одному из устройств и выбрать порт подключения. Аналогичные действия выполнить для второго устройства.

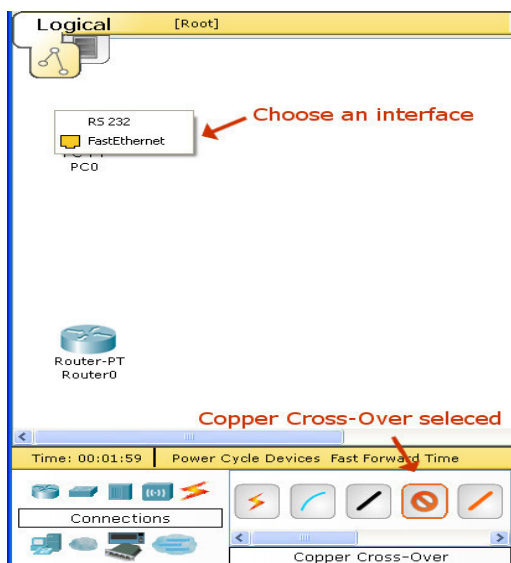


Рис. 4. Создание соединений.

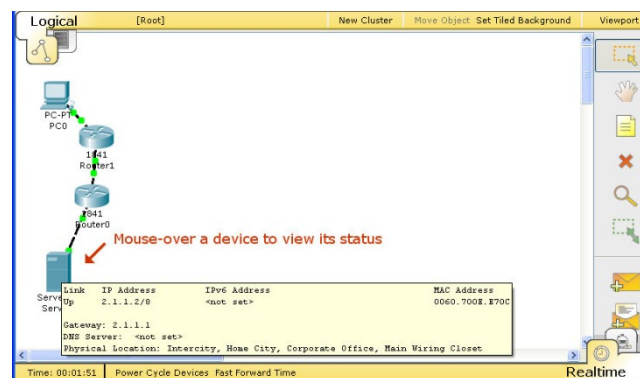


Рис. 5. Функционирование сети в режиме реального времени.

## Режим реального времени

В режиме реального времени, сеть всегда работает независимо от действий пользователя. Конфигурирование сети осуществляется в реальном времени. При просмотре статистики сети, они отображаются в режиме реального времени, как показано на панели инструментов. В дополнение к использованию Cisco IOS для настройки и диагностики сети, вы можете использовать Add Simple PDU и User Created PDU List для наглядной отправки пакета.

## Режим симуляции

В режиме симуляции, вы можете смотреть свои сети работать в более медленном темпе, исследуя пути, по которым пересылаются пакеты. При переключении в режим моделирования, появится специальная панель. Вы можете графически просматривать распространение пакетов по сети, нажав на кнопку Add Simple PDU. Имеется возможность контроля скорости моделирования с использованием кнопки Speed Slider. Также можно просматривать предыдущие события, нажав на кнопку Назад.

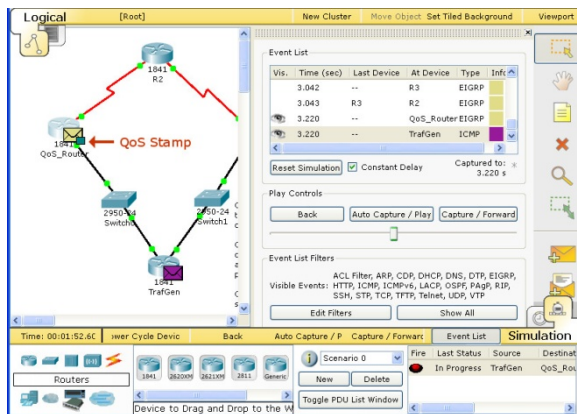


Рис.6. Режим моделирования

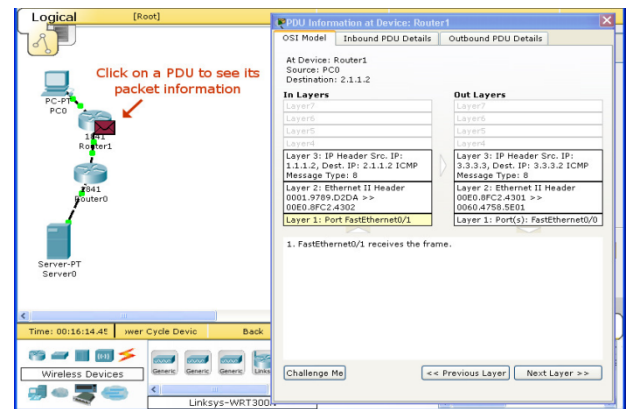


Рис.7. Информация о пакете

Во время моделирования можно кликнуть на пересылаемом пакете и получить о нем подробную информацию.

## Физическое пространство

Целью физической рабочей области является обзор физических аспектов логической топологии сети. Это дает ощущение масштаба и размещения (как ваша сеть может выглядеть в реальной среде).

Физическая рабочая область разделена на четыре слоя, чтобы отразить физический масштаб четыре среды: междугородная, город, дом, и коммутационного узла. Междугородный является крупнейшим окружающей среды. Он может содержать много городов. Каждый город может содержать множество зданий. Наконец, каждое здание может содержать множество шкафов проводки. Распределительный шкаф обеспечивает вид, который отличается от трех других видов. Здесь вы можете увидеть устройства, которые были созданы в логических Workspace; позиционируется в области сетевых технологий стойки и на столах. Три других слоя обеспечивают просмотр миниатюр их

макетов. Это по умолчанию расположение в физической рабочей области, но устройства в шкафу могут быть перемещены в любой из слоев.

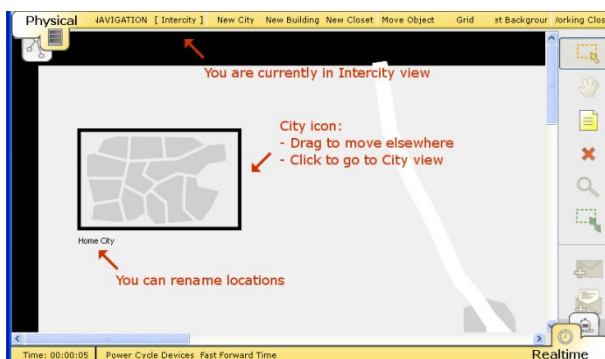


Рис.8. Междугородний масштаб

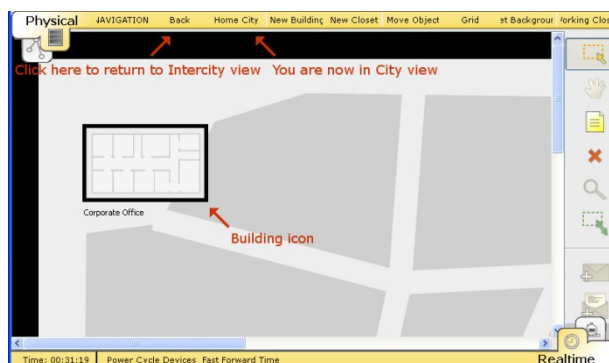


Рис.9. Масштаб здания

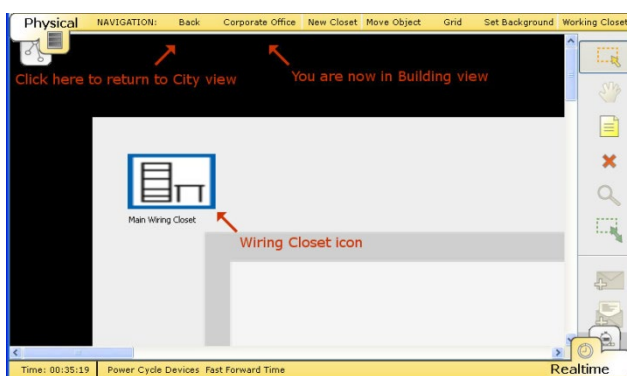


Рис.10. Коммутационный шкаф

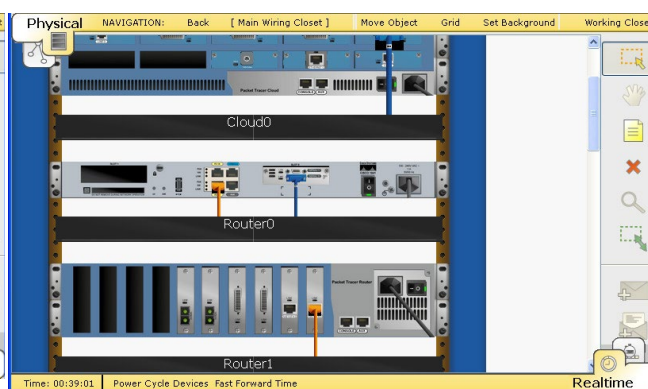


Рис.11. Устройства, размещенные в коммутационном шкафу

Содержание отчета



Обмен информацией между модулями происходит на основе определенных соглашений, которые называются интерфейсом. При передаче сообщения модуль верхнего уровня решает свою часть задачи, а результат, понятный только ему, оформляет в виде дополнительного поля к исходному сообщению (заголовка) и передает измененное сообщение на дообслуживание в нижележащий уровень. Этот процесс называется инкапсуляцией.

### **Стек протоколов Интернета**

Стек протоколов сети Интернет был разработан до модели OSI. Поэтому уровни в стеке протоколов Интернета не соответствуют аналогичным уровням в модели OSI. Стек протоколов Интернета состоит из пяти уровней: физического, звена передачи данных, сети, транспортного и прикладного. Первые четыре уровня обеспечивают физические стандарты, сетевой интерфейс, межсетевое взаимодействие и транспортные функции, которые соответствуют первым четырем уровням модели OSI. Три самых верхних уровня в модели OSI представлены в стеке протоколов Интернета единственным уровнем, называемым прикладным уровнем рис. 2.

Класс сети	Наименьший идентификатор сети	Наибольший идентификатор сети	Количество сетей
Класс А	1.0.0.0	126.0.0.0	126
Класс В	128.0.0.0	191.255.0.0	16384
Класс С	192.0.0.0	223.255.255.0	2097152

### Сетевое оборудование

**Мост (bridge)**, как и репитер, может соединять сегменты или локальные сети рабочих групп. Однако, в отличие от репитера, мост также служит для разбиения сети, что помогает изолировать трафик или отдельные проблемы. Например, если трафик одного-двух компьютеров или одного отдела “затопляет” сеть пакетами, уменьшая ее производительность в целом, мост изолирует эти компьютеры или этот отдел. Мосты обычно решают следующие задачи. Увеличивают размер сети. Увеличивают максимальное количество компьютеров в сети.



В среде, объединяющей несколько сетевых сегментов с различными протоколами и архитектурами, мосты не всегда гарантируют быструю связь между всеми сегментами. Для такой сложной сети необходимо устройство, которое не только знает адрес каждого сегмента, но и определяет наилучший маршрут для передачи данных и фильтрует широковещательные сообщения. Такое устройство называется **маршрутизатором**.

Маршрутизаторы (routers) работают на Сетевом уровне модели OSI. Это значит, что они могут переадресовывать и маршрутизировать пакеты через множество сетей, обмениваясь информацией (которая зависит от протокола) между отдельными сетями. Маршрутизаторы считывают в пакете адресную информацию сложной сети и, поскольку они функционируют на более высоком по сравнению с мостами уровне модели OSI, имеют доступ к дополнительным данным. Маршрутизаторы могут выполнять следующие функции мостов: фильтровать и изолировать трафик; соединять сегменты сети.

Устройства **Switch** - коммутатор. Это оборудование относится к активному сетевому оборудованию, и служит для обработки пакетов в сети. Свитч (то же самое, что переключатель, мост, switch, bridge) - устройство, служащее для разделения сети на отдельные сегменты, которые могут содержать хабы и сетевые карты. Свитчи являются устройствами 2-го уровня, т.е. содержат в себе порты - устройства 1-го уровня для работы с сигналами, но помимо того, работают с содержимым сетевых пакетов - читают поле физического адреса назначения (MAC) пакета, пришедшего на один из портов, и в зависимости от его значения и таблицы MAC-адрес - порт "ретранслируют" пакет на другой порт (или не ретранслируют).

## ВЫПОЛНЕНИЕ

Ниже на рис.3 приведена спроектированная сеть, которая включает в себя следующее оборудование:

- Маршрутизаторы;
- Коммутаторы;
- ПК;
- IP-телефоны;
- Сервер.

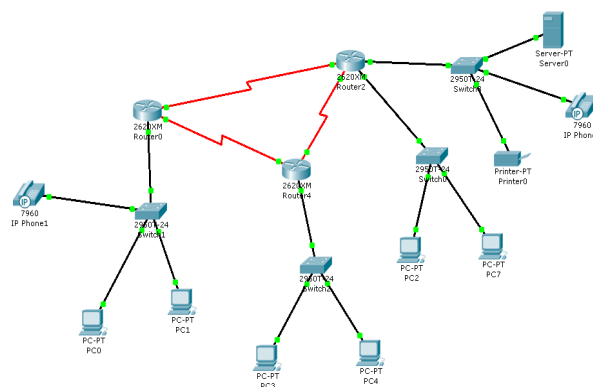


Рис. 3. Результат построения сети

Маршрутизаторы соединяются между собой при помощи DCE – кабеля. В данной сети маршрутизаторы используют RIP – протокол для осуществления передачи данных между различными подсетями(рис. 4).

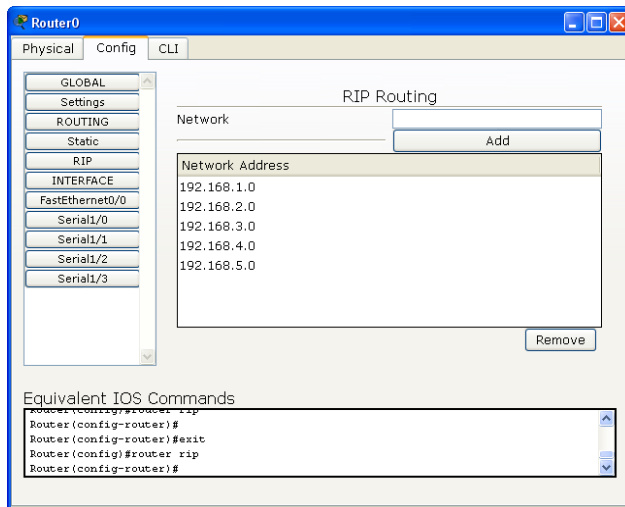


Рис. 4. Список подсетей

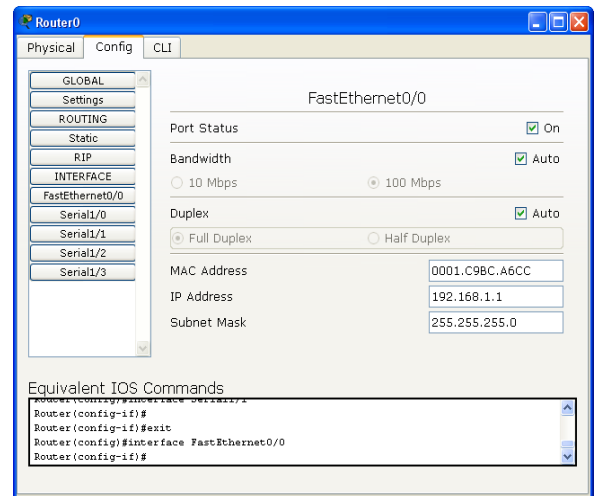


Рис. 5. Назначение IP-адреса маршрутизатору

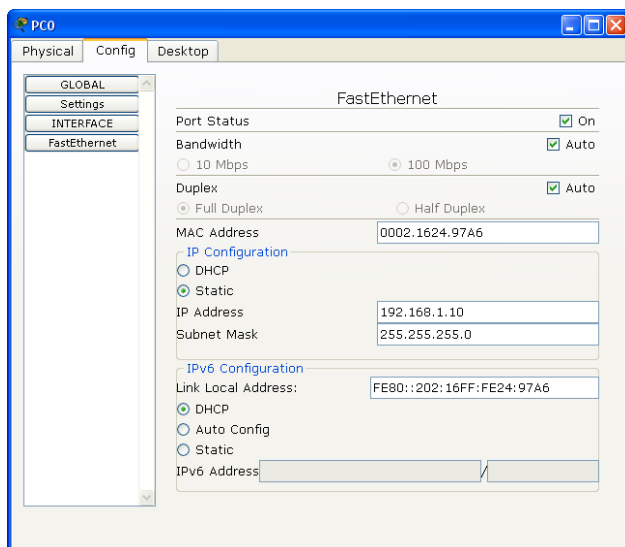
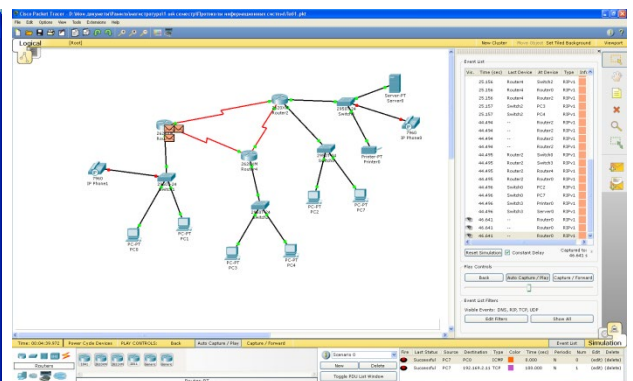


Рис.6. Назначение IP-адреса ПК



При этом, IP-адреса назначаются статически для маршрутизаторов и оконечного оборудования(рис. 5, 6).

## Лабораторная работа №3

### «Анализ протоколов уровня приложения и транспорта»

**Цель работы.** Провести анализ работы протоколов уровня приложений и транспорта с использованием программного сетевого эмулятора Packet Tracer Cisco Systems.

#### Краткие теоретические сведения

##### Модель TCP/IP

Семейство протоколов TCP/IP основано на четырехуровневой эталонной модели. Все протоколы, входящие в семейство протоколов TCP/IP, расположены на трех верхних уровнях этой модели.

Каждый уровень модели TCP/IP соответствует одному или нескольким уровням семиуровневой эталонной модели OSI (Open Systems Interconnection — взаимодействие открытых систем), предложенной ISO — международной организацией по стандартам (International Standards Organization).

Типы служб и протоколов, используемых на каждом уровне модели TCP/IP, более подробно описаны в следующей таблице.

Уровень	Описание	Протоколы
Приложение	Определяет прикладные протоколы TCP/IP и интерфейс программ со службами транспортного уровня, необходимый для использования сети.	HTTP, Telnet, FTP, TFTP, SNMP, DNS, SMTP, X Windows, другие прикладные протоколы
Транспортный	Обеспечивает управление сеансами связи между компьютерами. Определяет уровень служб и состояние подключения, используемые при транспортировке данных.	TCP, UDP
Интернет	Упаковывает данные в IP-датаграммы, содержащие информацию об адресах источника и приемника, которая используется для перенаправления датаграмм от узла к узлу и по сетям. Выполняет маршрутизацию IP-датаграмм.	IP, ICMP, ARP, RARP
Сетевого интерфейса	Определяет средства и принципы физической передачи данных по сети, включая преобразование битов данных в электрические или другие сигналы аппаратными устройствами, непосредственно подключенными к среде передачи, такой как коаксиальный кабель, оптоволокно или витая пара.	Ethernet, Token Ring, FDDI, X.25, Frame Relay, RS-232, v.35

#### Протоколы уровня приложений

*SMTP (Simple Mail Transfer Protocol)* – простой почтовый протокол. Он поддерживает передачу почтовых электронных сообщений по сети Интернет. Протокол называется простым, потому что обеспечивает передачу информации пользователям, готовым к немедленной доставке. Передача осуществляется в режиме 7-битовых слов. Он требует наличия программ перехода от принятого в большинстве программ формата с 8-разрядными словами к формату с 7-разрядными словами.

Система поддерживает:

- посылку одиночных сообщений одному или более получателям;

- посылку сообщений, включающих в себя текст, голосовые сообщения, видео или графические материалы.

*Протокол передачи файлов (FTP — File Transfer Protocol)* используется для передачи файлов от одного компьютера к другому. Обеспечивает просмотр каталогов удаленного компьютера, копирование, удаление и пересылку файлов. FTP отличается от других протоколов тем, что устанавливает два соединения между хостами. Одно используется для передачи информации, а другое — для управления передачей.

*DNS (Domain Name System)* – служба доменных имен. Она осуществляет присвоение уникальных имен всем пользователям и узлам сети Интернет и устанавливает логическую связь с их сетевыми адресами. Доменное имя представляется иерархической структурой, имеющей несколько уровней. Типовые имена доменов верхнего уровня закреплены следующим образом:

- .com – коммерческие организации;
- .gov – правительственные учреждения;
- .org – некоммерческие организации;
- .net — центры поддержки сети;
- .int – международные организации;
- .mil – военные структуры.

*SNMP (Simple Network Management Protocol)* — простой протокол управления сетью. Он обеспечивает набор фундаментальных действий по наблюдению и обслуживанию Интернета.

Протокол разработан так, чтобы он мог контролировать устройства, созданные различными изготовителями и установленные на различных физических сетях. Другими словами, SNMP освобождает задачи управления от учета физических характеристик управляемых устройств и от основной технологии организации сети.

*Сетевая файловая система (NFS — Network File System)*. Это один из многих протоколов (например, на рисунке показан еще один протокол RPC –Remote Procedure Call – вызов удаленной процедуры), который позволяет использование файлов, содержащих процедуры управления и периферии в другом компьютере.

*Тривиальный (простейший) протокол передачи файлов TFTP (Trivial File Transfer Protocol)*. Используется в простых случаях при начальной загрузке рабочих станций или загрузке маршрутизаторов, не имеющих внешней памяти.

*Протокол передачи гипертекста (HTTP — Hyper Text Transfer Protocol)* — транспортный протокол, который применяется в Интернете при обмене документами, представленными на языке описания гипертекстовых документов.

Язык разметки гипертекста (HTML — Hyper Text Markup Language). Является одним из главных языков, используемых в сети WWW.

### **Протоколы уровня транспорта**

*Протокол управления передачей TCP (Transmission Control Protocol)* является обязательным стандартом TCP/IP, который описан в документе RFC 793 «Transmission Control Protocol (TCP)» и предоставляет надежную службу доставки пакетов, ориентированную на установление соединения. Протокол TCP:

- гарантирует доставку IP-датаграмм;
- выполняет разбиение на сегменты и сборку больших блоков данных, отправляемых программами;
- обеспечивает доставку сегментов данных в нужном порядке;
- выполняет проверку целостности переданных данных с помощью контрольной суммы;

- посылает положительные подтверждения, если данные получены успешно. Используя избирательные подтверждения, можно также посылать отрицательные подтверждения для данных, которые не были получены;
- предлагает предпочтительный транспорт для программ, которым требуется надежная передача данных с установлением сеанса связи, например для баз данных «клиент-сервер» и программ электронной почты.

### Как работает TCP

TCP основан на связи «точка-точка» между двумя узлами сети. TCP получает данные от программ и обрабатывает их как поток байтов. Байты группируются в сегменты, которым TCP присваивает последовательные номера, необходимые для правильной сборки сегментов на узле-приемнике.

Чтобы два узла TCP могли обмениваться данными, им нужно сначала установить сеанс связи друг с другом. Сеанс TCP инициализируется с помощью процесса, называемого трехэтапным установлением связи. В этом процессе синхронизируются номера последовательности и передается управляющая информация, необходимая для установления виртуального соединения между узлами.

По завершении процесса трехэтапного установления связи начинается пересылка и подтверждение пакетов в последовательном порядке между этими узлами. Аналогичный процесс используется TCP перед прекращением соединения для того, чтобы убедиться, что оба узла закончили передачу и прием данных.

### Протокол UDP

Протокол датаграмм пользователя UDP (User Datagram Protocol) является стандартом TCP/IP, описанным в документе RFC 768 «User Datagram Protocol (UDP)». UDP используется некоторыми программами вместо TCP для быстрой, простой, но ненадежной передачи данных между узлами TCP/IP.

UDP обеспечивает службу датаграмм, не ориентированную на установление соединения, что означает, что UDP не гарантирует ни доставку, ни правильность порядка доставки датаграмм. Узел-источник, которому требуется надежная связь, должен использовать либо протокол TCP, либо программу, которая сама обеспечивает подтверждения и следит за правильностью порядка датаграмм.

UDP	TCP
Служба, не ориентированная на установление соединения; сеанс связи между узлами не устанавливается.	Служба, ориентированная на установление соединения; между узлами устанавливается сеанс связи.
UDP не гарантирует и не подтверждает доставку данных, а также не гарантирует порядок их доставки.	TCP гарантирует доставку при помощи подтверждений и контроля порядка принимаемых данных.
Программы, использующие UDP, ответственны за обеспечение надежности передачи данных.	Программам, использующим TCP, гарантируется надежность передачи данных.
UDP — быстрый протокол с небольшими накладными расходами, поддерживающий связь «точка-точка» и «точка-многие точки».	TCP медленнее, требует больших накладных расходов и поддерживает только связь «точка-точка».

## ВЫПОЛНЕНИЕ

Ниже на рис.1 приведена спроектированная сеть, которая включает в себя следующее оборудование:

- Маршрутизаторы;
- ПК;
- Сервер.



Рис. 1. Результат построения сети

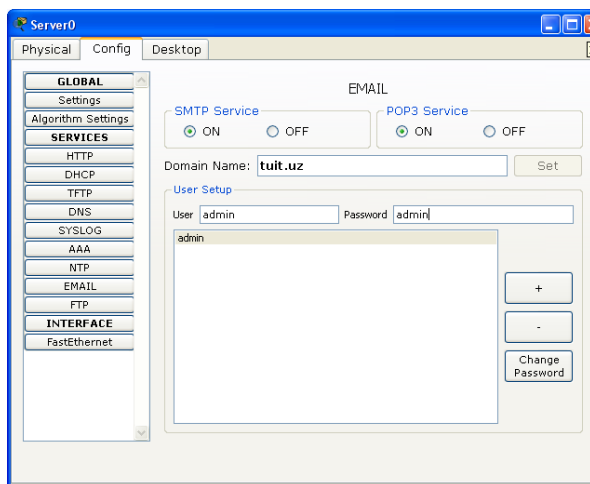


Рис.2. Настройка e-mail

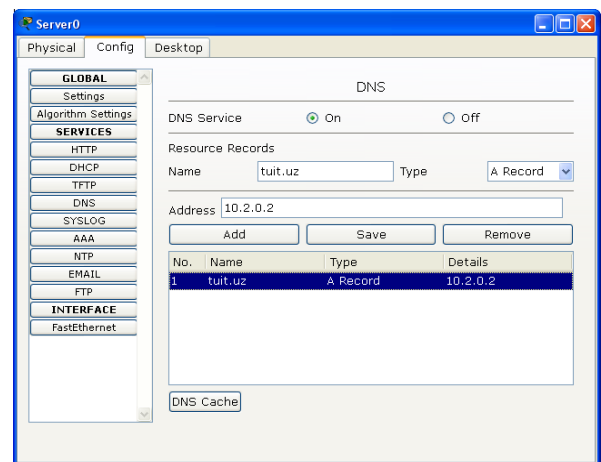


Рис.3. Настройка DNS-сервиса

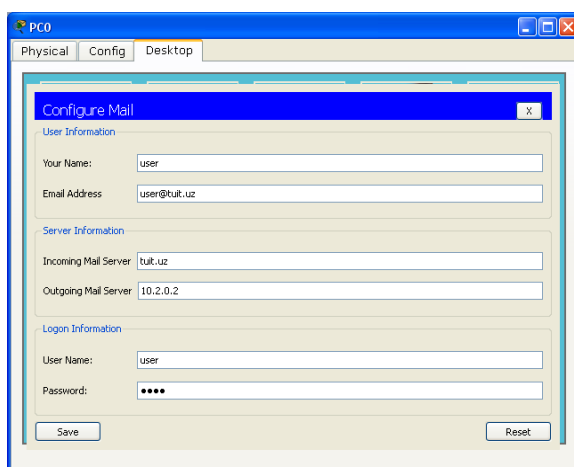


Рис.4. Настройка электронной почты

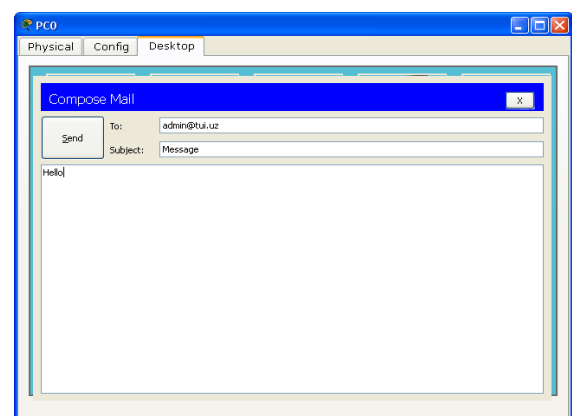


Рис.6. Передача сообщения

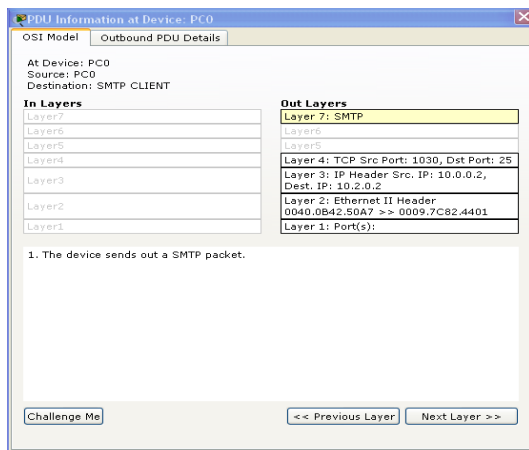


Рис.7. Передача пакетов SMTP

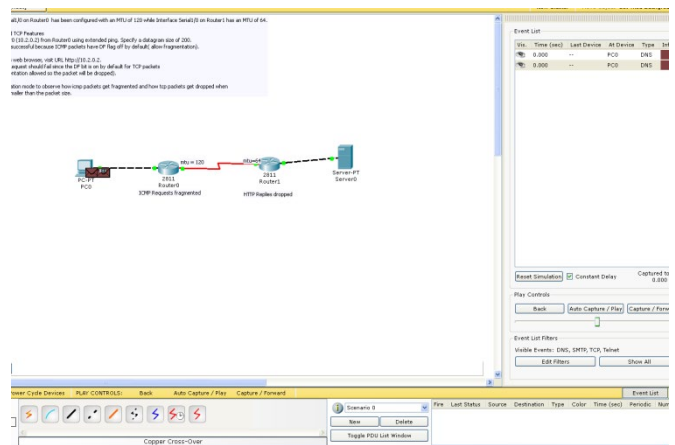


Рис.8. Работа DNS-сервиса

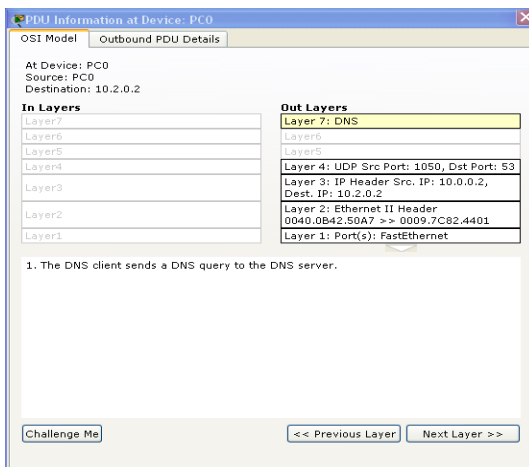


Рис.9. Выполнение DNS-запроса

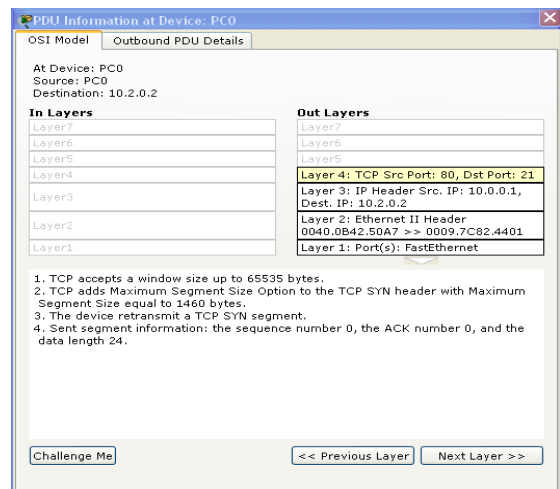


Рис.10. Передача TCP-пакетов

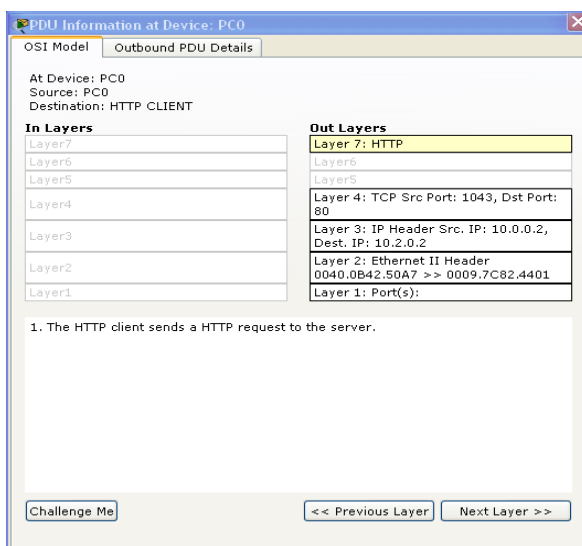


Рис.11. Выполнение HTTP-запроса

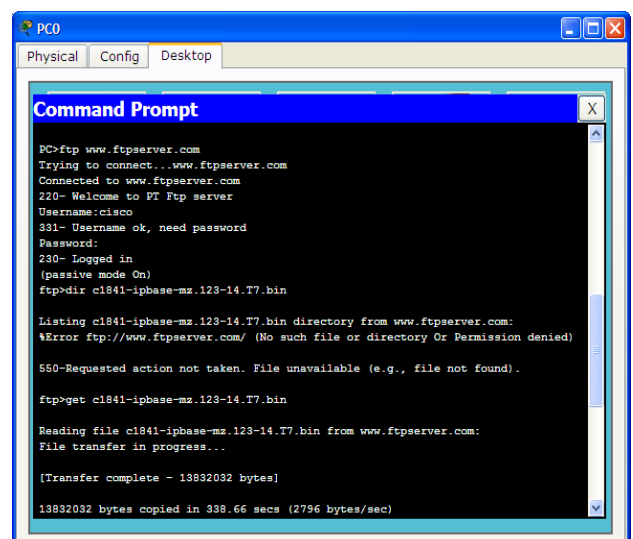


Рис.12. Работа с FTP-сервером

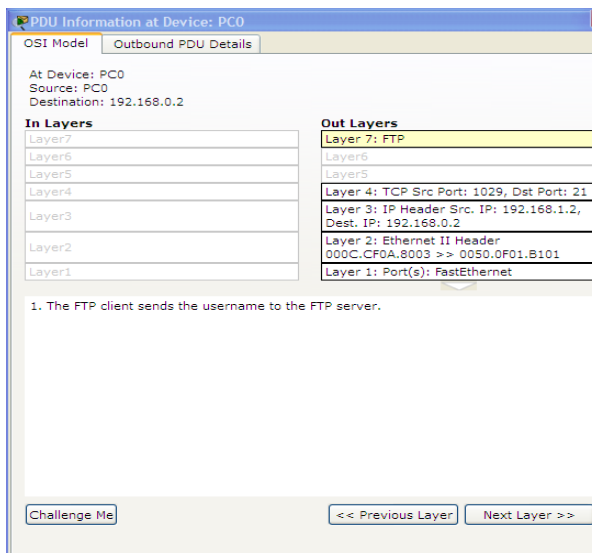


Рис.13. Авторизация пользователя на FTP-сервера

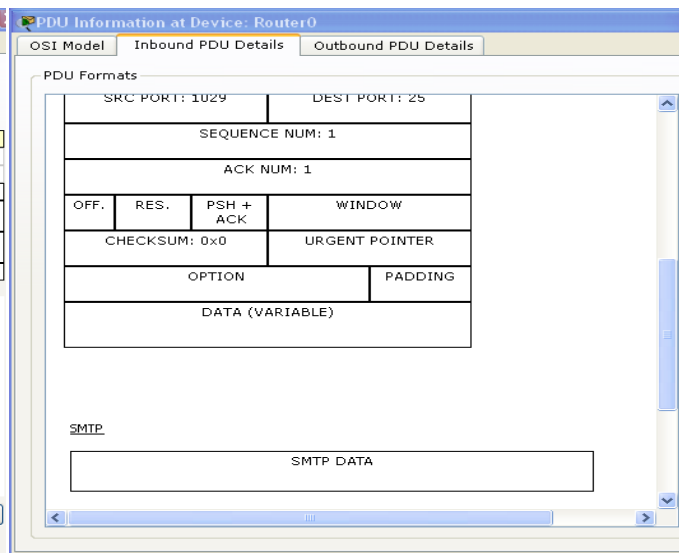


Рис.14. Структура пакетов

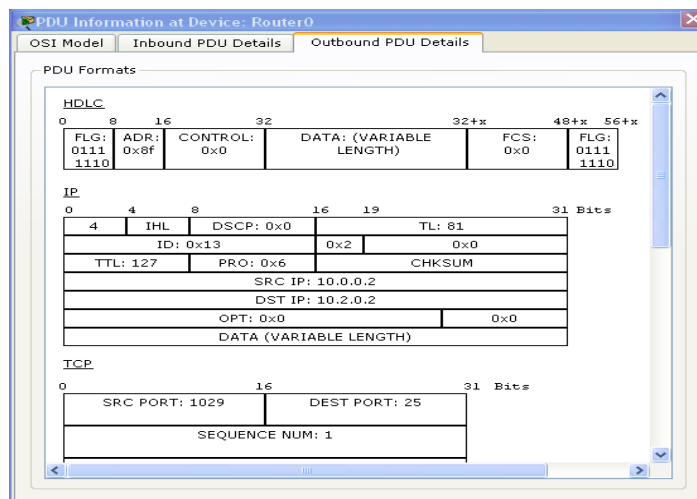


Рис.15. Структура пакетов

## Содержание отчета

1. Краткое теоретическое сведение
2. Анализ протоколов уровня приложения и транспорта.
3. Сравнение протоколов TCP и UDP.
4. Смоделированная сеть передачи данных с указанием параметров настройки.