**Лабораторные работы**

по дисциплине**Компьютерные сети**

**1.Тематика лабораторных работ**

Лабораторная работа №1 «Изучение работы протоколов ARP и ICMP»

Лабораторная работа №2 «Виртуальные локальные сети VLAN»

Лабораторная работа №3 «Настройка сетевых сервисов»

**2. Методические рекомендации по выполнению лабораторных работ**

Процесс выполнения лабораторной работы документируется с помощью текстового редактора и сопровождается *читаемыми рисунками*, полученные сведения служат основой для формирования отчета о выполнении лабораторной работы.

Защита отчета о выполнении лабораторной работы сопровождается демонстрацией полученных результатов, теоретических знаний и ответов на дополнительные вопросы преподавателя по теме занятия.

Работа должна содержать:

* Титульный лист
* Цель работы
* Ход работы
* Вывод

При наборе текста отчетов используется шрифт *Times New Roman*, размер шрифта – 14, выравнивание текста – по ширине, междустрочный интервал – 1,15. Поля отчета имеют следующие размеры: верхнее, нижнее – 2 см, левое – 2-3 см, правое – 1,5 см. Текст отчетов распечатывается на принтере с одной стороны листа.

Подписи к рисункам *обязательны* и должны иметь вид, как показано на рисунке 1.

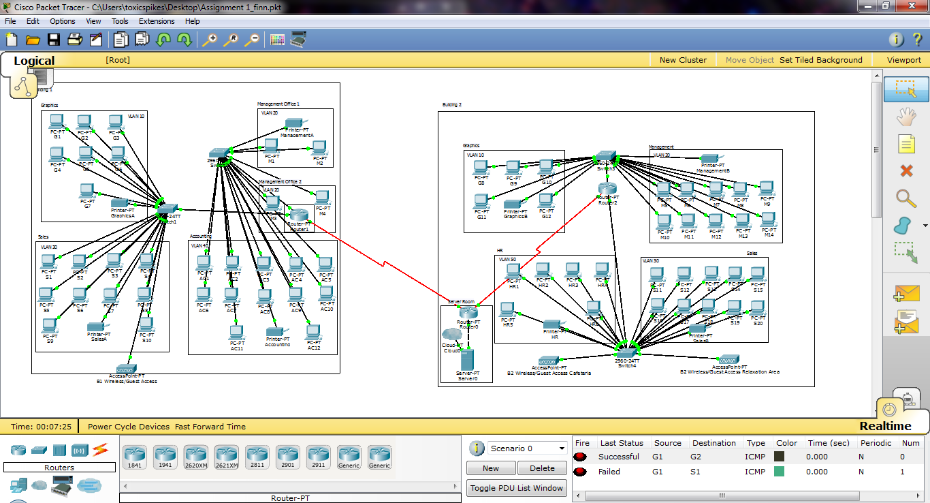


Рис. 1 – Пример нечитаемого рисунка

**3. Методические указания к выполнению лабораторных работ**

**Вводная теоретическая часть к лабораторным работам**

**Эмулятор компьютерных сетей Cisco Packet Tracer**

**Cisco Packet Tracer (CPT)** – это пакет программ для эмуляции работы компьютерных сетей, разработанный фирмой **CiscoSystems**. Пакет программ позволяет создавать визуальные модели сети, производить настройку элементов этой сети при помощи графического интерфейса и команд **cisco IOS**. Пакет позволяет эмулировать работу конкретных сетевых и пользовательских устройств: коммутаторов **Cisco** серии **2950**, **2960**, **3650**, маршрутизаторов **1800**, **2600**, **2800**, серверов **DHCP**, **HTTP**, **TFTP**, **FTP**, рабочих станций, предоставляет возможности устанавливать различные модули расширения в компьютеры, коммутаторы и маршрутизаторы. Пакет программ позволяет создавать макеты компьютерных сетей довольно сложных топологий, проверять работоспособность и проводить исследования сетей.

***Ссылка для скачивания CPT:*** [***https://disk.yandex.ru/d/7f-qgOPk2iK6pA***](https://disk.yandex.ru/d/7f-qgOPk2iK6pA) ***(для полного доступа к программе перед запуском отключить Интернет-соединение).***

**Графический интерфейс Cisco Packet Tracer**

Основная работа выполняется в главном окне программы, представляющей графический интерфейс (Рис 1).

Наименования и функции для основных полей главного окна **Cisco Packet Tracer**, соответствующие нумерации Рис.1, приведены ниже.

**1.** ***Главное меню*** содержит стандартные для многих программ пункты: Файл, Правка, Настройки, Вид, Инструменты, Расширения, Помощь. Особого внимания заслуживает пункт «Расширения», содержащий мастер проектов, многопользовательский режим и ряд других дополнительных возможностей.

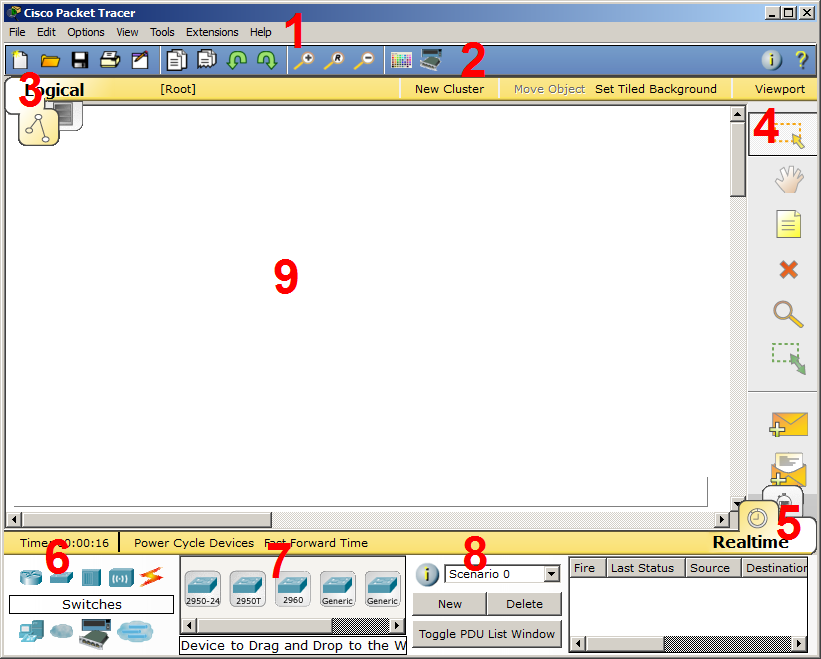


Рис. 1

**2.** ***Панель инструментов***, часть которых дублирует пункты главного меню.

**3**. ***Переключатель*** логической и физической организации рабочего пространства.

**4.** ***Панель инструментов***, содержащая средства выделения, удаления, перемещения, масштабирования объектов, а также формирования и передачи пакетов данных (PDU) между устройствами.

**5**. ***Переключатель*** режима реального времени (***Realtime***) и режима имитации (***Simulation Mode***).

**6**. ***Панель*** выбора группы коммуникационных устройств, оконечных станций и линий связи.

**7.** ***Панель***, содержащая конкретные типы коммуникационных устройств (маршрутизаторов, коммутаторов, концентраторов), оконечных устройств и линий связи. Содержимое этой панели зависит от выбранной группы устройств в предыдущем пункте. Используя символические обозначения конкретных устройств, можно собрать логическую схему сети, перенося символы методом *Drag* *and Drop* в рабочее пространство.

**8**. Панель создания пользовательских сценариев.

**9**. Рабочее пространство.

**Оборудование и линии связи в Cisco Packet Tracer**

***Маршрутизаторы***

http://litl-admin.ru/wp-content/uploads/2012/02/3df0abae8c2db6526774005880df699e.png

Маршрутизаторы (роутеры) используется для поиска оптимального маршрута передачи данных на основании специальных алгоритмов маршрутизации, например выбор маршрута (пути) с наименьшим числом транзитных узлов. Работают на сетевом уровне модели OSI.

***Концентраторы***

http://litl-admin.ru/wp-content/uploads/2012/02/b9cd4e1bbf9593a5be7dd8a95d690cad.png

Концентратор повторяет пакет, принятый на одном из его портов на всех остальных портах.

***Коммутаторы***

http://litl-admin.ru/wp-content/uploads/2012/02/b4cba0912611a18713680e52ce219359.png

Коммутаторы – это устройства, работающие на канальном уровне модели OSI и предназначенные для объединения нескольких узлов в пределах одного или нескольких сегментах сети. Передаёт пакеты коммутатор на основании внутренней таблицы – таблицы коммутации, следовательно, трафик идёт только на тот MAC-адрес, которому он предназначается, а не повторяется на всех портах (как на концентраторе).

***Беспроводные устройства***

http://litl-admin.ru/wp-content/uploads/2012/02/afba2322e689f5547831388e738b04ed.png

Беспроводные технологии Wi-Fi и сети на их основе. Включает в себя точки доступа.

***Линии связи***

http://litl-admin.ru/wp-content/uploads/2012/02/16310d3508ad04fd4217980a5bb47803.png

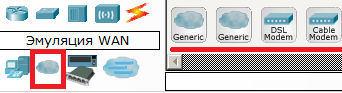
С помощью этих компонентов узлы соединяются в единую схему. Здесь есть автоматическое определение типа кабеля, консольный кабель, витая пара и оптоволокно.

***Конечные устройства***



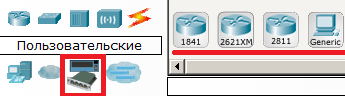
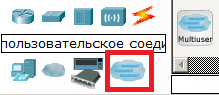
Здесь присутствуют конечные узлы: компьютеры, сервера, принтеры, телефоны и т.д.

***Эмуляция Интернета***



Эмуляция глобальной сети: модем DSL, “облако” и т.д.

***Пользовательские устройства и облако для многопользовательской работы***

**Режим симуляции Cicso Packet Tracer**

Режим симуляции (имитации) позволяет проследить, как проходят пакеты по сети.

Чтобы перейти в режим симуляции необходимо либо кликнуть на иконку симуляции в правом нижнем углу рабочего пространства, либо с помощью клавиш **Shift+S**. Появится окно событий (список событий), кнопка сброса (сбросить симуляцию - очищает список событий), управление воспроизведением (Авто захват/воспризведение и Захват/Вперед [2]) и фильтр протоколов) (рис. 2). Здесь предложено много протоколов, но на первой стадии изучения можно отфильтровать только протокол **ICMP**.

## Задачей протокола межсетевых управляющих сообщений ICMP ( Internet Control Message Protocol) является контроль правильности работы сети. Утилита ping, служащая для проверки возможности доставки IP-пакетов, использует ICMP-сообщения эхо-запрос (определяет, доступен ли в сети IP-узел: компьютер или маршрутизатор) и эхо-ответ (отвечает на эхо-запрос ICMP)

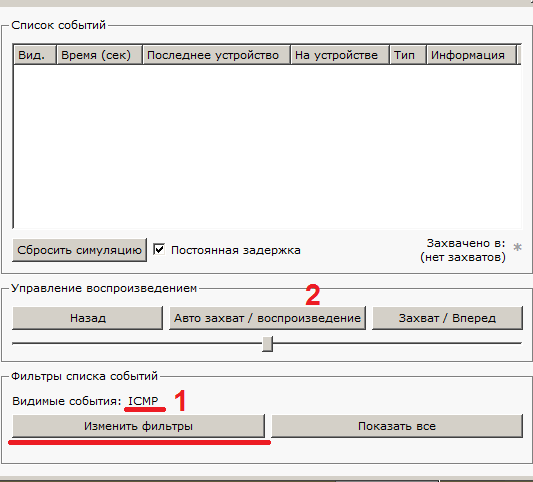


Рис. 2

Рассмотрим типовую конфигурацию **Cisco Packet Tracer** (рис.3).Здесь 4 узла, сервер, принтер и два концентратора (хаба).

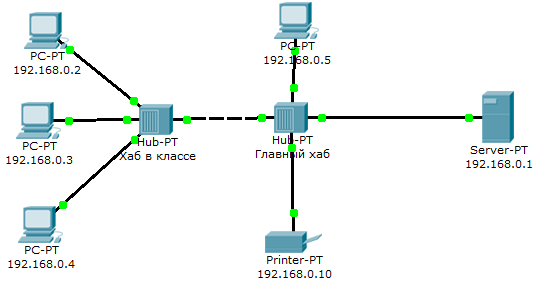


Рис. 3

Для того чтобы проверить доступность, например, узла PC5 (192.168.0.5) с узла PC4 (192.168.0.4), необходимо в окне управления PC4 во вкладке ″Desktop″ (рабочий стол) выбрать приложение “*Command promt*” (командная строка) и выполнить команду ***ping 192.168.0.5*** (рис. 4).

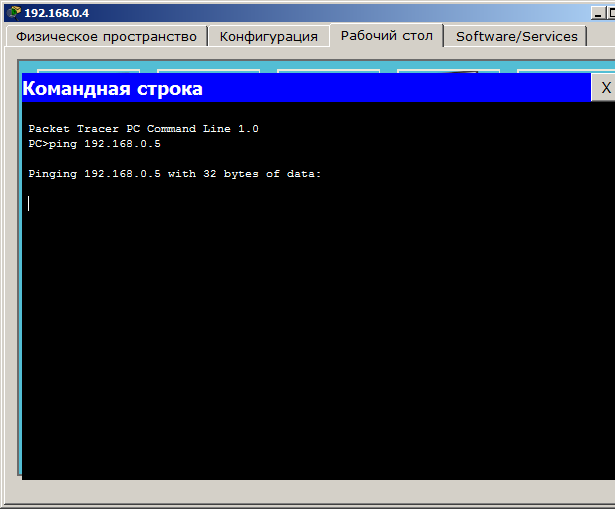


Рис. 4

Можно будет видеть, как на узле 192.168.0.4 образовался пакет (конвертик), который ждёт (иконка паузы на нём, рис. 5). Стронуть пакет с места можно нажав кнопку “Вперёд” в окне симуляции.

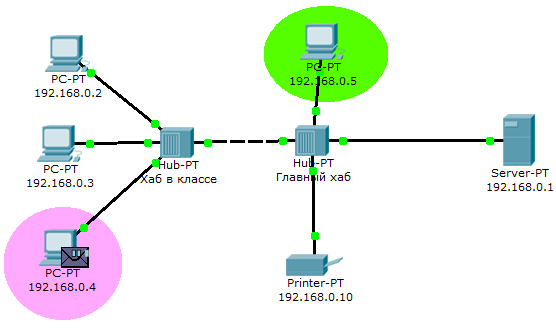


Рис. 5

Так же в окне симуляции можно увидеть (рис. 6) этот пакет, его тип (ICMP) и источник (192.168.0.4).

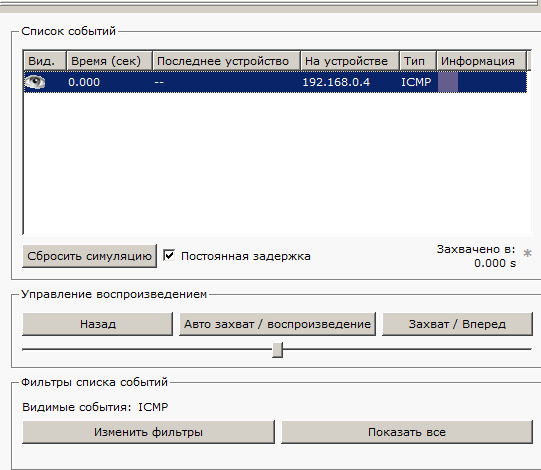


Рис. 6

Если кликнуть на пакет, то можно получить подробную информацию (рис. 7). Согласно модели OSI видно, что пакет возник на 3-ем (сетевом) уровне на исходящем направлении, который пойдёт до второго уровня, затем до первого, на физическую среду и передастся на следующий узел.

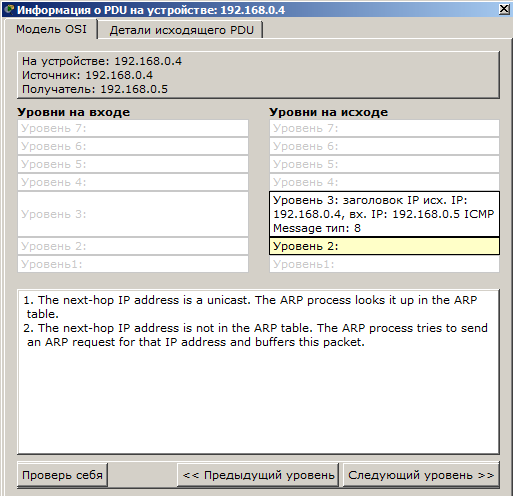


Рис. 7

На вкладке ***Детали исходящего* PDU** можно посмотреть структуру пакета (рис. 8).

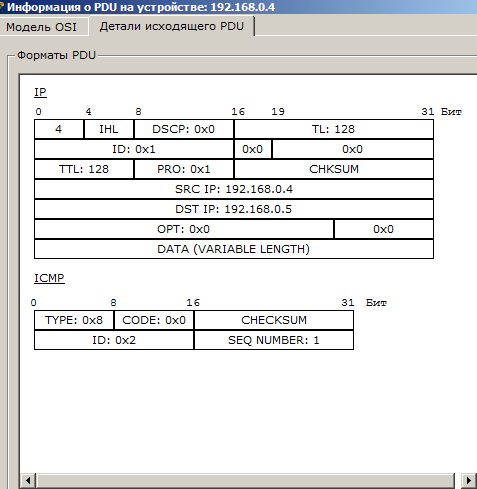


Рис. 8

При нажатии кнопки “Вперёд” пакет поступает на концентратор (рис. 9).

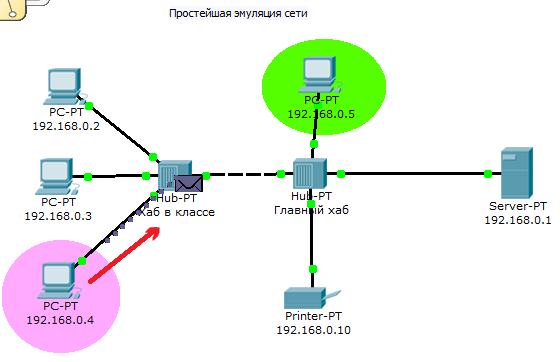


Рис. 9

Концентратор повторяет пакет на всех остальных портах (рис. 10).

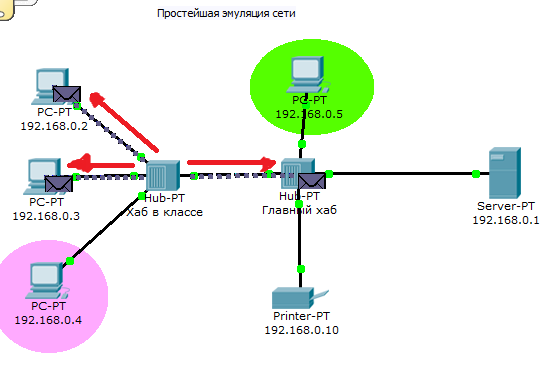


Рис. 10

Те узлы, для которых пакеты не предназначены, игнорируют эти пакеты (рис. 11).

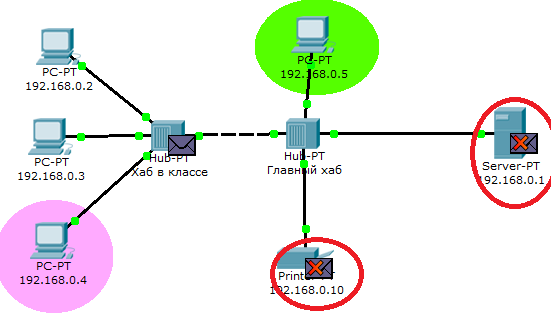


Рис. 11

Когда пакет вернётся обратно к источнику, появятся строчки: ***Reply from 192.168.0.5: bytes=32 time=16ms TTL=128.***

**Лабораторная работа № 1**

**«****Изучение работы протоколов ARP и ICMP»**

**Цель работы:** изучить режим симуляции Cisco Packet Tracer, протоколы ARP и ICMP на примере программ ping и tracert.

**Программа работы:**

1. Построение топологии сети, настройка конечных узлов.
2. Настройка маршрутизатора
3. Проверка работы сети в режиме симуляции.
4. Посылка ping-запроса внутри сети.
5. Посылка ping-запроса во внешнюю сеть.
6. Посылка ping-запроса на несуществующий IP-адрес узла.

**Теоретические сведения:**

**Протокол ARP**

Для определения физического адреса по IP-адресу используется протокол разрешения адреса Address Resolution Protocol (ARP).

Протокол ARP позволяет динамически определить МАС-адрес по IP-адресу. МАС-адрес – это уникальный серийный номер, присваиваемый каждому сетевому устройству для идентификации его в сети, так же называется физическим или аппаратным адресом. В Ethernet сетях, использующих стек TCP/IP, сетевой интерфейс имеет физический адрес длиной в 48 бит. Кадры, которыми обмениваются на канальном уровне, должны содержать аппаратный адрес сетевого интерфейса. Однако стек TCP/IP использует собственную схему адресации: 32-битные IP-адреса. Значение IP-адреса приемника недостаточно, чтобы отправить датаграмму этому хосту. Чтобы послать данные, драйвер Ethernet должен знать МАС-адрес интерфейса назначения. В задачу протокола ARP входит обеспечение динамического соответствия между 32-битными IP-адресами и 48-битными МАС-адресами, используемыми различными сетевыми технологиями. Протокол ARP работает в пределах одной подсети и автоматически запускается, когда возникает необходимость преобразования IP-адреса в аппаратный адрес.

В ходе обычной работы сетевая программа отправляет прикладное сообщение, пользуясь транспортными услугами протокола TCP. Модуль TCP посылает соответствующее транспортное сообщение через модуль IP. В результате составляется IP-пакет, который должен быть передан драйверу Ethernet. IP-адрес места назначения известен прикладной программе, модулю TCP и модулю IP. Необходимо на его основе найти Ethernet-адрес места назначения. Для определения искомого Ethernet-адреса используется ARP-таблица.

ARP-таблица хранится в памяти и содержит строки для каждого узла сети. В двух столбцах содержатся IP- и Ethernet-адреса. Если требуется преобразовать IP-адрес в Ethernet-адрес, то ищется запись с соответствующим IP-адресом. Ниже приведен пример упрощенной ARP-таблицы.

|  |  |
| --- | --- |
| **IP-адрес** | **Ethernet-адрес** |
| 192.168.3.1. 192.168.3.2.  192.168.3.3. | 08:00:39:00:2F:C3  08:00:5A:21:A7:22  08:00:10:99:AC:54 |

Рис.1. Пример ARP-таблицы

Принято все байты 4-байтного IP-адреса записывать десятичными числами, разделенными точками. При записи 6-байтного Ethernet-адреса каждый байт указывается в 16-ричной системе и отделяется двоеточием.

ARP-таблица необходима, потому что IP-адреса и Ethernet-адреса выбираются независимо, и нет какого-либо алгоритма для преобразования одного в другой. IP-адрес выбирает менеджер сети с учетом положения машины в сети Internet. Если машину перемещают в другую часть сети Internet, то ее IP-адрес должен быть изменен. Ethernet-адрес выбирает производитель сетевого интерфейсного оборудования из выделенного для него по лицензии адресного пространства. Когда у машины заменяется плата сетевого адаптера, то меняется и ее Ethernet-адрес.

Каждая запись в ARP-таблице имеет свое время жизни. При добавлении записи для нее активируется таймер.

ARP-таблица заполняется автоматически модулем ARP узла по мере необходимости. Когда с помощью существующей ARP-таблицы не удается преобразовать IP-адрес, то узел, которому нужно выполнить отображение IP-адреса на физический адрес, формирует ARP-запрос, указывая в нем известный IP-адрес получателя, вкладывает его в кадр протокола канального уровня и рассылает запрос ***широковещательно***. Все узлы локальной сети получают ARP запрос и сравнивают указанный там IP-адрес с собственным. В случае их совпадения узел формирует ARP-ответ, в котором указывает свой IP-адрес и свой физический адрес и отправляет его уже направленно, так как в ARP запросе отправитель указывает свой локальный адрес.

ARP-запросы и ответы используют один и тот же формат пакета. На рис. 2 показан формат пакета запросов и ответов.

|  |  |  |  |
| --- | --- | --- | --- |
| 7 | 15 | 23 | 31 |
| Тип оборудования | | Протоколсетевого уровня | |
| HA-Len | PA-Len | Код операции | |
| Аппаратный адрес отправителя (октеты 0,1,2,3) | | | |
| Адрес отправителя (октеты 4,5) | | IP-адрес отправителя (октеты 0,1) | |
| IP-адрес отправителя (октеты 2,3) | | Аппаратный адрес получателя (0,1) | |
| Аппаратный адрес получателя (октеты 2,5) | | | |
| IP-адрес получателя | | | |

Рис. 2.Формат пакета ARP

Здесь: HA-Len - длина аппаратного адреса; PA-Len - длина сетевого адреса (длина в байтах, например, для IP-адреса *PA-Len=4*); поле *тип оборудования* - это тип интерфейса, для которого отправитель ищет адрес; код содержит 1 для Ethernet; поле *код операции* определяет, является ли данный пакет ARP-запросом (код = 1), ARP-откликом (2), RARP-запросом (3), или RARP-откликом (4). Это поле необходимо, как поле тип кадра в Ethernet пакетах, они идентичны для ARP-запроса и отклика.

Узел, отправляющий ARP-запрос, заполняет в пакете все поля, кроме поля искомого физического адреса. Значение этого поля заполняется узлом, опознавшим свой IP-адрес.

Каждый сетевой адаптер принимает широковещательные передачи. Все драйверы Ethernet проверяют поле типа в принятом Ethernet-кадре и передают ARP-пакеты модулю ARP. ARP-запрос можно интерпретировать так: "Если ваш IP-адрес совпадает с указанным, то сообщите мне ваш Ethernet-адрес". Пакет ARP-запроса выглядит примерно так:

|  |  |
| --- | --- |
| IP-адрес отправителя  Ethernet-адрес отправителя | 192.168.3.1.  08:00:39:00:2F:C3 |
| Искомый IP-адрес  Искомый Ethernet-адрес | 192.168.3.4.  <пусто> |

Рис. 3. Пример ARP-запроса

Каждый модуль ARP проверяет поле искомого IP-адреса в полученном ARP-пакете и, если адрес совпадает с его собственным IP-адресом, то посылает ответ по Ethernet-адресу отправителя запроса. ARP-ответ можно интерпретировать так: "Да, это мой IP-адрес, ему соответствует такой-то Ethernet-адрес". Пакет с ARP-ответом выглядит примерно так:

|  |  |
| --- | --- |
| IP-адрес отправителя  Ethernet-адрес отправителя | 192.168.3.4.  08:00:28:00:38:A9 |
| Искомый IP-адрес  Искомый Ethernet-адрес | 192.168.3.1.  08:00:39:00:2F:C3 |

Рис. 4. Пример ARP-ответа

Этот ответ получает узел, сделавший ARP-запрос. Драйвер этого узла проверяет поле типа в Ethernet-кадре и передает ARP-пакет модулю ARP. Модуль ARP анализирует ARP-пакет и добавляет запись в свою ARP-таблицу.

Обновленная таблица выглядит следующим образом:

|  |  |
| --- | --- |
| **IP-адрес** | **Ethernet-адрес** |
| 192.168.3.1. 192.168.3.2.  192.168.3.3.  192.168.3.4. | 08:00:39:00:2F:C3  08:00:5A:21:A7:22  08:00:10:99:AC:54  08:00:28:00:38:A9 |

Рис. 5. ARP-таблица после обработки ответа

**Протокол ICMP**

Протокол ICMP предназначен для передачи управляющих и диагностических сообщений. С его помощью передаются сообщения об ошибках, а также о возникновении ситуаций, требующих повышенного внимания. Протокол относится к сетевому уровню модели TCP/IP. Сообщения ICMP генерируются и обрабатываются протоколами сетевого (IP) и более высоких уровней (TCP или UDP). При появлении некоторых ICMP-сообщений генерируются сообщения об ошибках, которые передаются пользовательским процессам. ICMP-сообщения передаются внутри IP-дейтаграмм (рис. 6).



Рис. 6. Инкапсуляция ICMP-сообщений в IP-дейтаграммы

Существует несколько типов сообщений ICMP. Каждый тип сообщения имеет свой формат, при этом все они начинаются с общих трех полей: 8-битного целого числа, обозначающего тип сообщения (TYPE), 8-битного поля кода (CODE), который конкретизирует назначение сообщения, и 16-битного поля контрольной суммы (CHECKSUM). Поле контрольной суммы охватывает ICMP-сообщение целиком.

**Программа ping**

Программа ping была разработана для проверки доступности удаленного узла. Программа посылает ICMP-эхо-запрос на узел и ожидает возврата ICMP-эхо-отклика. Программа рing является обычно первым диагностическим средством, с помощью которого начинается идентификация какой-либо проблемы в сетях. Помимо доступности, с помощью ping можно оценить время возврата пакета от узла, что дает представление о том, "насколько далеко" находится узел. Кроме этого, Ping имеет опции записи маршрута и временной метки. Сообщения эхо-запроса и эхо-отклика имеют один формат (рис 7).

|  |  |  |
| --- | --- | --- |
| Тип (8 или 0) | Код (0) | Контрольная сумма |
| Идентификатор | | Последовательный номер |
| Данные | | |

Рис. 7. Формат эхо-запроса и отклика ICMP

* Тип – тип пакета: 8 – запрос эха, 0 – ответ на запрос эха.
* Код – расшифровка назначения пакета внутри типа (в данном случае 0)
* Контрольная сумма вычисляется для всего пакета.
* Идентификатор – обычно это идентификатор процесса.
* Последовательный номер – номер пакета в потоке (увеличивается на 1 при посылке каждого пакета).

Поля ***идентификатор*** и ***последовательный номер*** служат для того, чтобы отправитель мог связать в пары запросы и отклики. Поле *данные* служит для записи информации, возвращаемой отправителю. При выполнении процедуры ping эхо-запрос с временной меткой в поле данные посылается адресату. Если адресат активен, он принимает IP-пакет, меняет адрес отправителя и получателя местами и посылает его обратно. ЭВМ-отправитель, восприняв этот отклик, может сравнить временную метку, записанную им в пакет, с текущим показанием внутренних часов и определить время распространения пакета (RTT - round trip time). Размер поля *данные* не регламентирован и определяется предельным размером IP-пакета.

В поле идентификатора ICMP сообщения устанавливается идентификатор процесса, отправляющего запрос. Это позволяет программе ping идентифицировать вернувшийся ответ, если на одном и том же хосте в одно и то же время запущено несколько программ ping.

Номер последовательности начинается с 0 и инкрементируется каждый раз, когда посылается следующий эхо-запрос. Вывод программы показан на рис. 8. Первая строка вывода содержит IP-адрес хоста назначения, даже если было указано имя. Поэтому программа рing часто используется для определения IP-адреса удаленного узла.

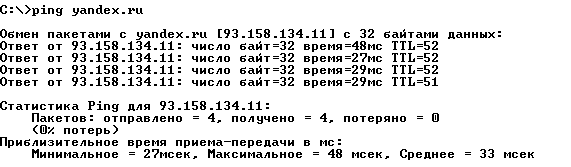


Рис. 8. Вывод программы **ping**

**Программа tracert**

Программа tracert позволяет посмотреть маршрут, по которому двигаются IP-датаграммы от одного хоста к другому.

Программа tracert не требует никаких специальных серверных приложений. В ее работе используются стандартные функции протоколов ICMP и IP. Для понимания работы программы следует вспомнить порядок обработки поля *время жизни* (TTL) в заголовке IP-датаграммы.

Каждый маршрутизатор, обрабатывающий датаграмму, уменьшает значение поля TTL в ее заголовке на единицу. При получении дейтаграммы с TTL равным 1, маршрутизатор уничтожает ее и посылает хосту, который ее отправил, ICMP-сообщение "время истекло". При этом дейтаграмма, содержащая это ICMP-сообщение, имеет в качестве адреса источника IP-адрес маршрутизатора.

Это и используется в программе tracert. На хост назначения отправляется IP-датаграмма, в которой поле TTL установлено в единицу. Первый маршрутизатор на пути датаграммы, уничтожает ее (так как TTL равно 1) и отправляет ICMP-сообщение об истечении времени. Таким образом, определяется первый маршрутизатор в маршруте. Затем tracert отправляет датаграмму с полем TTL равным 2, что позволяет получить IP-адрес второго маршрутизатора. Аналогичные действия продолжаются до тех пор, пока датаграмма не достигнет хоста назначения. При получении ответа от этого узла процесс трассировки считается завершённым.

Пример вывода программы показан на рис. 9.

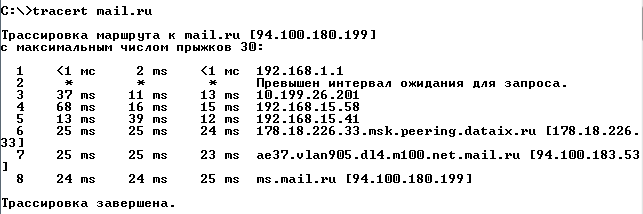


Рис.9 Вывод программы **tracert**

Первая строка, без номера, содержит имя и IP адрес пункта назначения и указывает на то, что величина TTL не может быть больше 30.

Следующие строки вывода начинаются с распечатки значения TTL (1, 2, 3 и т.д.) и содержат имя (IP-адрес) хоста или маршрутизатора и время возврата ICMP-сообщения.

Для каждого значения TTL отправляется 3 датаграммы. Для каждого возвращенного ICMP-сообщения рассчитывается и печатается время возврата.

Если ответ на датаграмму не получен в течение пяти секунд, печатается звездочка, после чего отправляется следующая дейтаграмма.

**Работа протоколов ARP и ICMP в Cisco Packet Tracer**

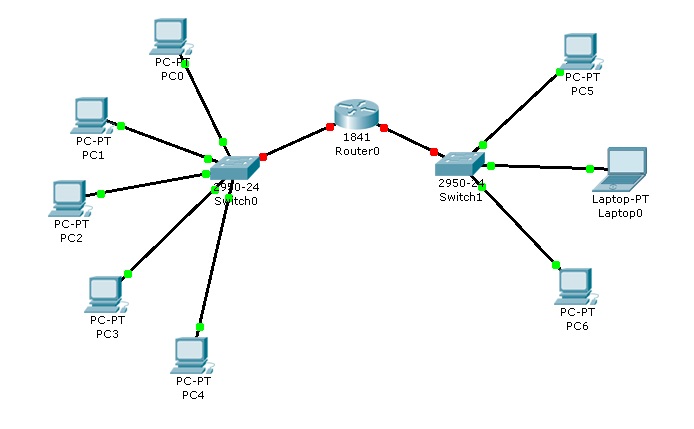


Рис. 10. Тестовая топология сети

Маршрутизатор Router0 имеет два интерфейса и соединяет две подсети. Произведем настройку конечных узлов.

1. **Настройка конечных узлов**

На устройствах PC0-PC4 установим заданные IP-адреса и маску подсети (табл. 1). IP-адрес шлюза для всех узлов – 192.168.3.1. IP-адрес DNS-сервера указывать необязательно, т.к. в данной работе он использоваться не будет.

Таблица 1

|  |  |  |
| --- | --- | --- |
| Хост | IP-адрес | Маска подсети |
| PC0 | 192.168.3.3 | 255.255.255.0 |
| PC1 | 192.168.3.4 | 255.255.255.0 |
| PC2 | 192.168.3.5 | 255.255.255.0 |
| PC3 | 192.168.3.6 | 255.255.255.0 |
| PC4 | 192.168.3.7 | 255.255.255.0 |

На устройствах PC5, Laptop0, PC6 установим заданные IP-адреса и маску подсети (табл. 2). IP-адрес шлюза для всех узлов – 192.168.5.1. IP-адрес DNS-сервера указывать необязательно.

Таблица 2

|  |  |  |
| --- | --- | --- |
| Хост | IP-адрес | Маска подсети |
| PC5 | 192.168.5.3 | 255.255.255.0 |
| Laptop0 | 192.168.5.4 | 255.255.255.0 |
| PC6 | 192.168.5.5 | 255.255.255.0 |

Каждый узел переименуем его же IP-адресом, получится следующее (рис. 11):

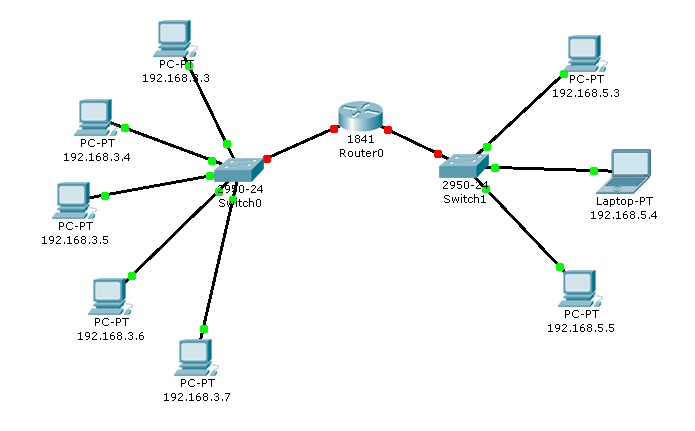


Рис. 11. Вид рабочей области

1. **Настройка маршрутизатора**

При настройке конечных узлов уже упоминалось о том, что маршрутизатор в данной топологии сети имеет два интерфейса. Произведем настройку интерфейса FastEthernet0/0:

1. Один клик по устройству (маршрутизатору);
2. Выбираем вкладку “Config”;
3. Находим интерфейс FastEthernet0/0, задаем нужный IP-адрес и маску подсети (рис. 12).

***Важно: интерфейс маршрутизатора, по умолчанию, отключен; необходимо его включить, кликнув мышкой рядом с “On”.***

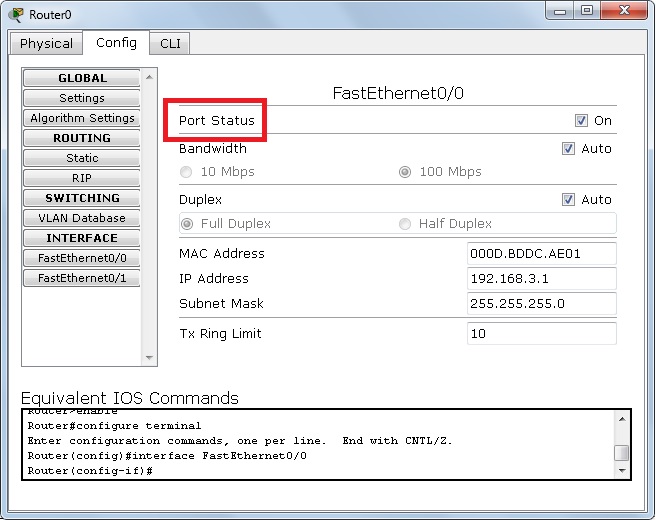


Рис. 12. Настройка интерфейса маршрутизатора

Аналогично производим настройку интерфейса FastEthernet0/1.

Сделать надписи к интерфейсам маршрутизатора, можно с помощью инструмента Place Note на панели Common Tools . Необходимо кликнуть на инструмент, затем сделать клик в нужном месте на рабочей области.

1. **Режим симуляции Cisco Packet Tracer**

Убедитесь, что вы находитесь в режиме симуляции. Для этого кликните на иконку симуляции в правом нижнем углу рабочей области симулятора. Описание: 8.png. Откроется окно событий, в котором вы увидите список событий, управляющие кнопки, заданные фильтры (рис. 13). По умолчанию, фильтруются, т.е. будут отображаться, пакеты всех возможных протоколов, необходимо поправить и ограничить этот список до исследуемых протоколов.

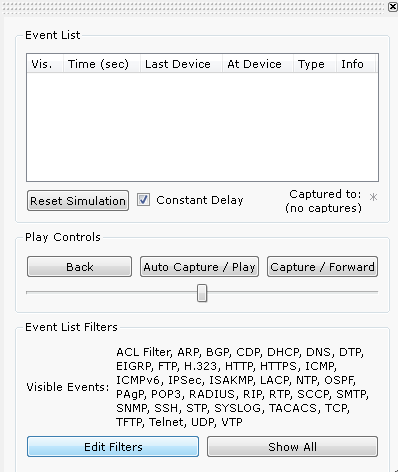


Рис. 13 Окно событий режима симуляции

Управляющие кнопки: Back – назад; Auto Capture/Play – автоматический захват пакетов от источника до приемника и обратно; Capture/Forward – захват пакетов только от одного устройства до другого

Нас интересуют пакеты двух типов: **ARP** и **ICMP**. Следовательно, нужно поставить фильтр только на сообщения заданного типа (рис. 14):

1. Нажимаем на кнопку “Edit Filters”
2. Снимаем метку с “Show All/None”
3. Выбираем ARP и ICMP

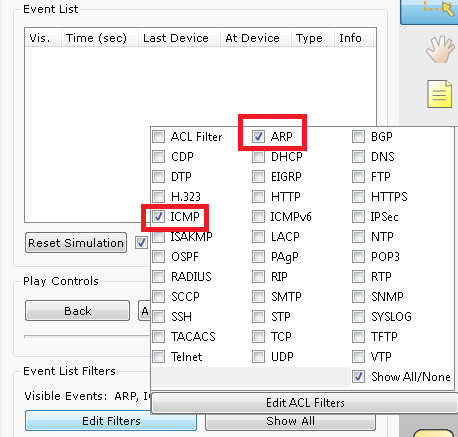


Рис. 14 Добавление фильтров на протоколы **ARP** и **ICMP**

1. **Проверка работы сети в режиме симуляции**

Отправим ping-запрос с конечного узла c IP-адресом 192.168.3.3 на хост с IP-адресом 192.168.3.5.

**Важно: оба узла находятся в пределах одного сегмента сети**

1. Один клик по выбранному устройству.
2. Выбираем вкладку Desktop, в которой содержатся симуляторы некоторых программ, доступных на компьютере.
3. Выбираем “Command Prompt”, программу, имитирующую командную строку компьютера.
4. С помощью утилиты ping отправляем ping-запрос (рис. 15). (Не забудьте нажать Enter).

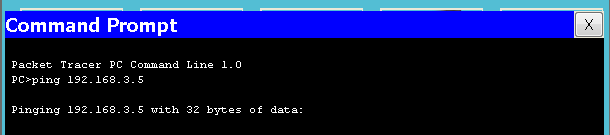


Рис. 15. Командная строка узла 192.168.3.3

На устройстве-источнике формируются два пакета протокола ARP и ICMP. ARP-запрос возникает всегда, когда хост пытается связаться с другим хостом, но не имеет информации об его МАС-адресе.

Узел 192.168.3.3 построил ARP-запрос (рис. 16) и посылает его широковещательным сообщением всем хостам подсети. Помимо IP-адреса назначения, запрос содержит IP-адрес и МАС-адрес отправителя, чтобы приемная сторона могла ответить.

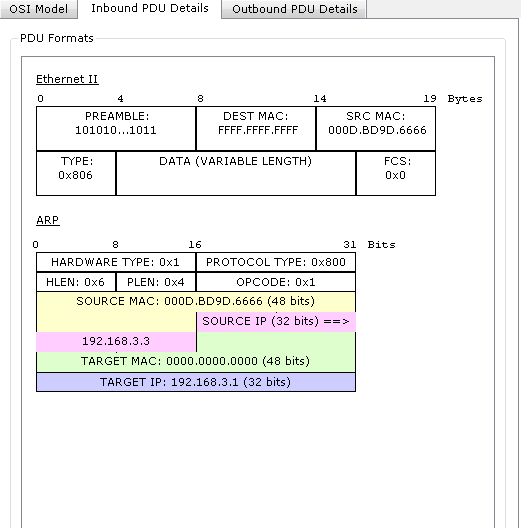


Рис. 16. Формат пакета ARP-запроса

Каждый хост в подсети получает запрос и проверяет на соответствие свой IP-адрес. Если он не совпадает с указанным адресом в запросе, то запрос игнорируется (рис. 17).

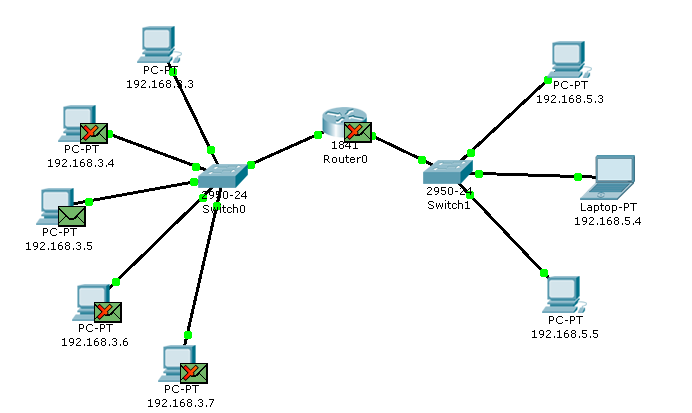


Рис. 17. Вид рабочей области

При просмотре прохождения пакетов убедитесь, что на ARP-запрос ответит только хост 192.168.3.5. Посмотрите содержимое пакета ARP-ответа, пришедшего на хост 192.168.3.3 (рис. 14). Узел 192.168.3.5. послал ARP-ответ непосредственно отправителю, используя его МАС-адрес, с указанием собственного МАС-адреса в поле “Target МАС”.

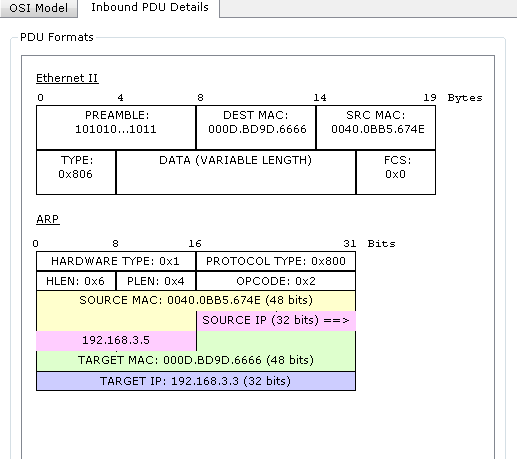


Рис. 18. Формат пакета ARP-ответа

Далее отправляется ICMP-сообщение ping-запроса. Посмотрите содержимое пакета, сделав клик по пакету (конверту) (рис. 19).

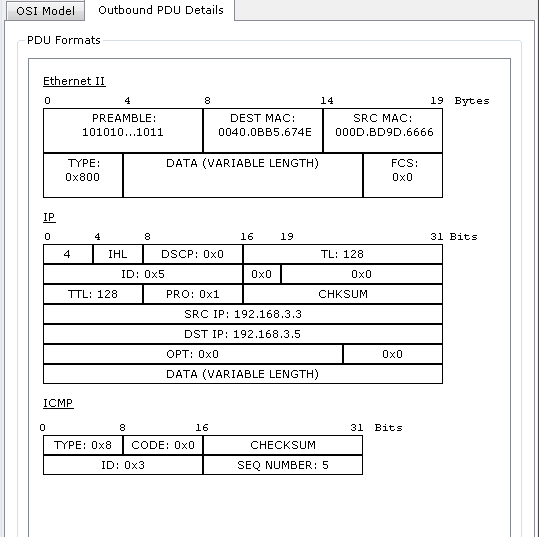


Рис. 19. Формат пакета ICMP-эхо-запроса

Физические адреса узлов известны. IP-адрес источника – 192.168.3.3. IP-адрес назначения – 192.168.3.5. Тип ICMP-сообщения – 8 (эхо-запрос).

Запрос производится на хост 192.168.3.5 через коммутатор (рис. 20).

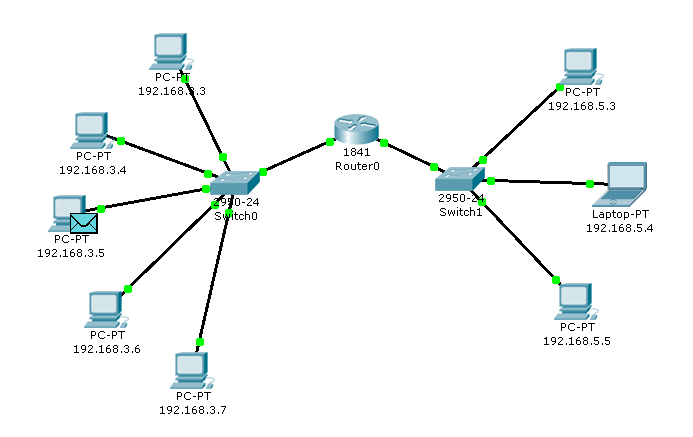


Рис. 20. Вид рабочей области

Посмотрите содержимое пакета ping-ответа, пришедшего на хост 192.168.3.3 (рис. 21).

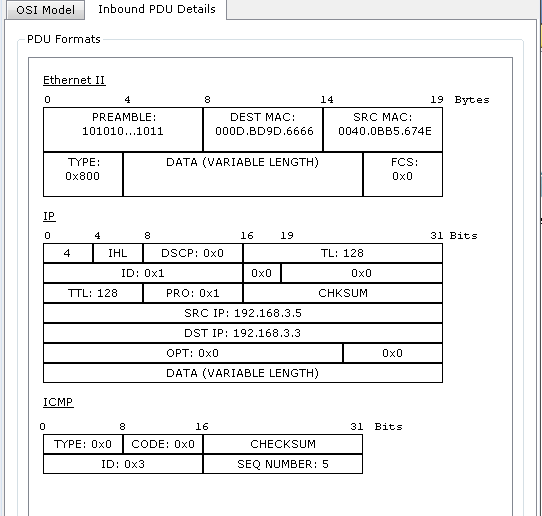


Рис.21. Формат пакета ICMP-эхо-ответа

IP-адрес источника – 192.168.3.5. IP-адрес назначения – 192.168.3.3. Тип ICMP-сообщения – 0 (эхо-ответ).

Посмотрите ping-ответ в командной строке хоста 192.168.3.3 (рис. 22).

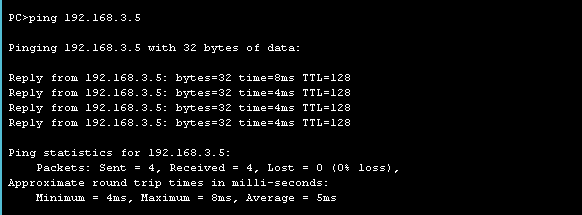


Рис. 22. Вывод программы ping

В окне событий так же указаны маршруты запроса ARP и ICMP: через какие устройства прошли пакеты (рис. 23).

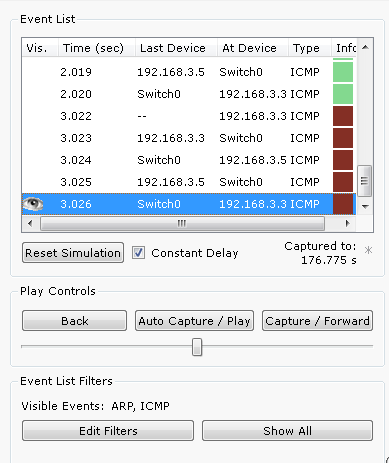


Рис. 23. Окно событий режима симуляции

Удалить сценарий симуляции можно с помощью кнопки “Reset Simulation” или воспользоваться кнопкой “Delete” в области User Created Packet Window.

Теперь ARP-таблицы хостов 192.168.3.3 и 192.168.3.5 не пусты, в них содержится одна запись. Чтобы просмотреть содержимое ARP-таблицы, нужно выполнить команду “arp –a” в командной строке.

Содержимое ARP-таблицы узла 192.168.3.3 (рис. 24):

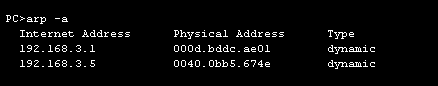


Рис. 24. ARP-таблица узла 192.168.3.3 в командной строке

Можно воспользоваться другим способом: нажать на кнопку «Inspect» , нажать на выбранное устройство, выбрать «ARP table» и просмотреть записи ARP-таблицы узла (рис.25).

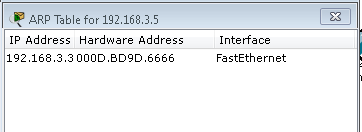


Рис. 25. ARP-таблица узла 192.168.3.5, показанная с помощью инструмента «Inspect»

Если снова задать ping-запрос на хост 192.168.3.5, то сразу будет сформирован только один пакет ICMP-сообщения, т.к. в ARP-таблице компьютера-источника уже хранится соответствующий локальный адрес.

Попробуйте отправить ping-запрос снова. Чтобы удалить все записи ARP-таблицы, следует воспользоваться командой “arp –d”.

1. **Посылка ping-запроса во внешнюю сеть**

Рассмотрим ситуацию, когда узел-источник и узел-приемник находятся в разных сетях. Отправим ping-запрос с конечного узла c IP-адресом 192.168.3.4 на хост с IP-адресом 192.168.5.5.

***Важно: один узел пытается передать пакет другому узлу, находящемуся с ним в разных сетях.***

Протокол ARP работает в пределах сегмента сети, поэтому в данном случае он будет использоваться для определения МАС-адреса маршрутизатора. Таким образом, пакет будет передан маршрутизатору для дальнейшей ретрансляции.

Открываем “Command Promt”, имитирующую командную строку, на компьютере 192.168.3.4 и посылаем на хост 192.168.5.5 ping-запрос. На узле-источнике также формируются два пакета протокола ARP и ICMP , но в этом случае инициируется ARP-запрос маршрутизатору 192.168.3.1, который пересылает пакеты в сеть 192.168.5.0 рис. 26).

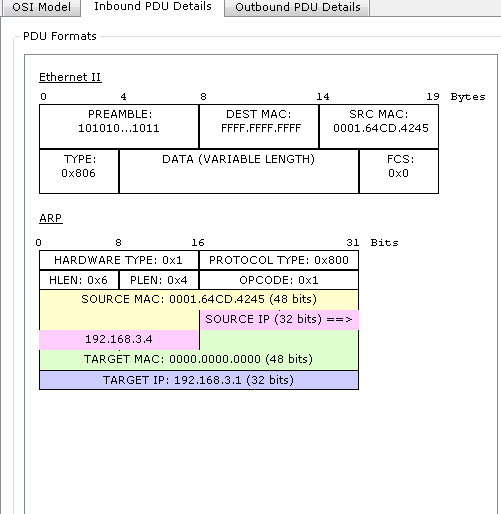


Рис. 26 Формат пакета ARP-запроса

Все узлы игнорируют пакет, кроме маршрутизатора, которому этот пакет предназначался. Маршрутизатор формирует ARP-ответ, указывая свой физический адрес, и отправляет его узлу 192.168.3.4. После получения ARP-ответа хост 192.168.3.4 посылает ICMP-сообщение ping-запроса через маршрутизатор в сеть назначения.

Посмотрите содержимое пакета, сделав клик по пакету (конверту) (рис. 27).

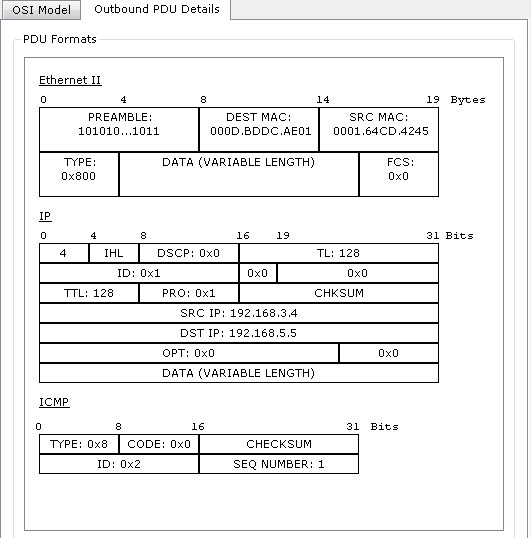


Рис. 27 Формат пакета ICMP-эхо-запроса

IP-адрес источника – 192.168.3.4. IP-адрес назначения – 192.168.5.5. Тип ICMP-сообщения – 8 (эхо-запрос).

Когда запрос приходит в сеть назначения, то маршрутизатор определяет МАС-адрес получателя, если такового нет в ARP-таблице маршрутизатора. Таким образом, снова решается задача разрешения локального адреса (рис. 28).

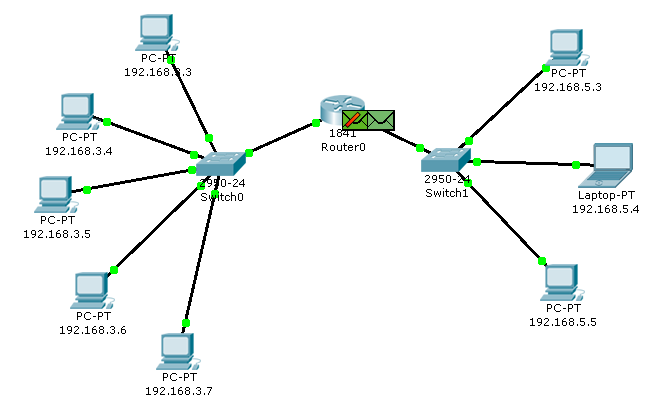


Рис. 28. Вид рабочей области

Маршрутизатор вынужден сначала узнать физический адрес получателя, прежде чем он сможет отправить ping-запрос по назначению, поэтому пакет с ping-запросом, пришедший на маршрутизатор, отклонен.

Новый ARP-запрос отправляется широковещательным сообщением от маршрутизатора, содержит его IP-адрес и МАС-адрес (рис. 25). IP-адрес назначения – узел 192.168.5.5.

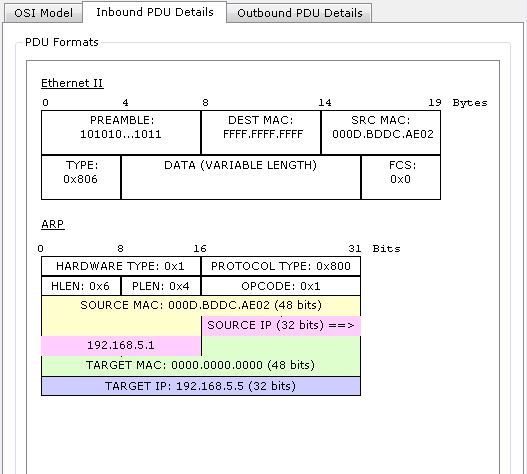


Рис. 29 Формат пакета ARP-запроса

Узлы подсети, которым пакет не предназначен, его игнорируют. Узел 192.168.5.5. формирует ARP-ответ и отправляет его маршрутизатору, указав свой МАС-адрес, о чем свидетельствует содержимое пакета (рис. 30).

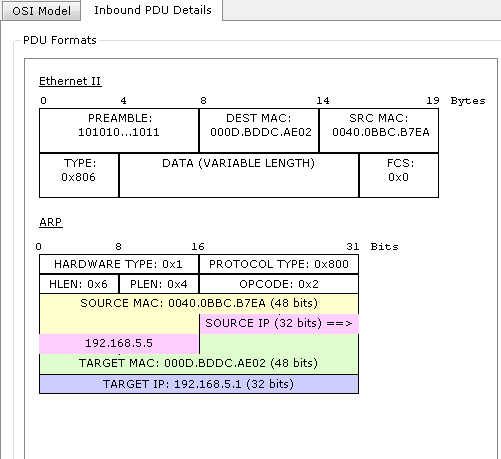


Рис. 30. Формат пакета ARP-ответа

После того, как маршрутизатор определил МАС-адрес получателя входящего ping-запроса, он посылает ICMP-ответ маршрутизатору хоста отправителя. (В данном случае это тот же маршрутизатор Router0).

Узел 192.168.3.4. снова пытается отправить ping-запрос во внешнюю сеть узлу 192.168.5.5. Его маршрут должен лежать через коммутатор Switch0, маршрутизатор Router0, коммутатор Switch1 и достигнуть узла назначения. Проследите маршрут пакета самостоятельно.

Узел 192.168.5.5 формирует ping-ответ, который отправляется обратно узлу 192.168.3.4.

Посмотрите содержимое пакета ping-ответа, пришедшего на хост 192.168.3.4 (рис. 31).

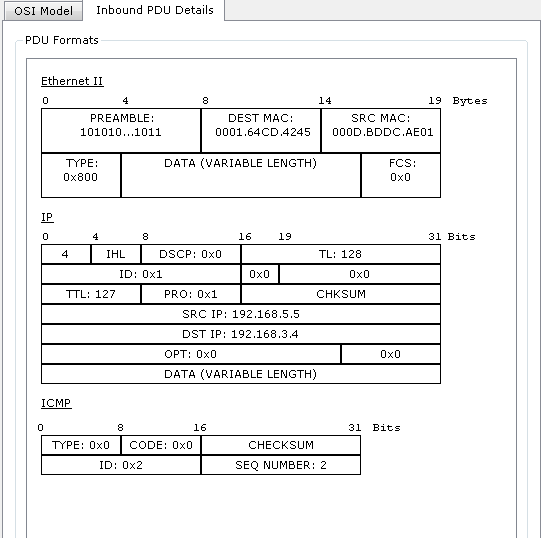


Рис. 31. Формат пакета ICMP-эхо-ответа

IP-адрес источника – 192.168.5.5. IP-адрес назначения – 192.168.3.4. Тип ICMP-сообщения – 0 (эхо-ответ).

Маршрут пакета можно посмотреть с помощью команды tracert. Выполним эту команду, например, в командной строке компьютера 192.168.3.5 (рис. 32):

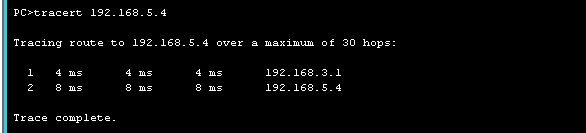


Рис. 32. Вывод программы tracert

На пути пакета до хоста 192.168.5.5 один промежуточный маршрутизатор.

1. **Посылка ping-запроса на несуществующий хост**

Отправим ping-запрос на несуществующий адрес в сеть 192.168.5.0/24.

Откроем программу “Command Promt” на узле 192.168.3.7 и попробуем отправить ping-запрос на несуществующий хост с IP-адресом 192.168.5.6.

ARP-таблица на узле-источнике не содержит соответствующей записи о МАС-адресе узла 192.168.5.6, поэтому формируется ARP-запрос.

Все узлы игнорируют пакет, кроме маршрутизатора, которому этот пакет предназначался. Узел 192.168.3.7 получает ARP-ответ с МАС-адресом маршрутизатора. Теперь, зная его аппаратный адрес, хост отправляет ping-запрос на узел 192.168.5.6.

Маршрутизатор пришедший пакет уничтожает, т.к. не может его перенаправить на указанный адрес, потому что соответствующего МАС-адреса он «не знает». В связи с этим маршрутизатор формирует ARP-запрос по адресу 192.168.5.6.

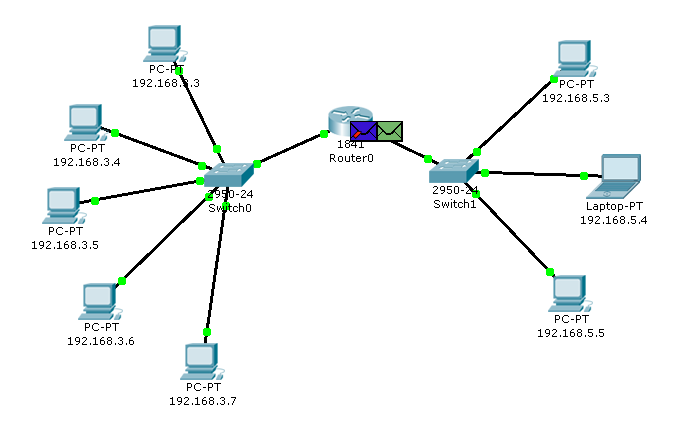


Рис. 33 Вид рабочей области

Все узлы подсети игнорируют пакет, потому что IP-адрес в запросе не соответствует их собственным. Маршрутизатор ни какого ответа ни от кого не получает.

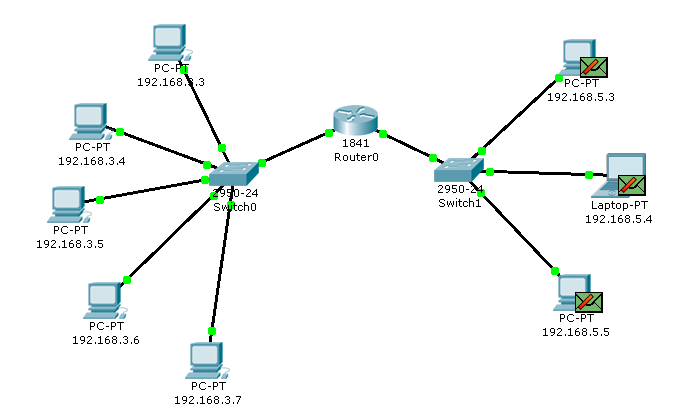
программный

Рис. 34. Вид рабочей области

Процедура прохождения пакетов повторяется в течение всего сценария симуляции: маршрутизатор по-прежнему «не знает» МАС-адрес указанного в ping-запросе IP-адреса 192.168.5.6 и продолжает рассылать ARP-запросы. Ни один из узлов подсети на эти запросы не реагирует. Не получив ответа, маршрутизатор и сам «молчит», никак не уведомляя об ошибке хост-источник ping-запроса.

Посмотрим ответ на ping-запрос в командной строке узла-источника 192.168.3.7: «превышено время ожидания» (рис. 35).

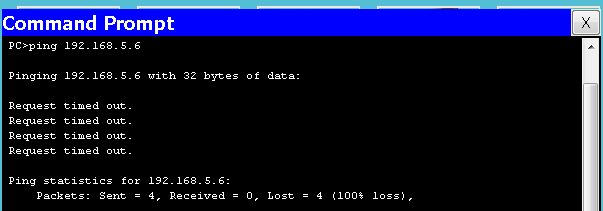


Рис. 35 Вывод программы ping

Попробуем отправить ping-запрос, содержащий IP-адрес узла, в сеть, на которую нет маршрута.

Откроем программу “Command Promt” на узле 192.168.3.6 и попробуем отправить ping-запрос на несуществующий хост с IP-адресом 192.168.6.6.

Так как ARP-таблица узла-источника соответствующей записи не имеет, формируется ARP-запрос на заданный узел с IP-адресом 192.168.6.6.

Все узлы игнорируют пакет, кроме маршрутизатора, которому этот пакет предназначался.

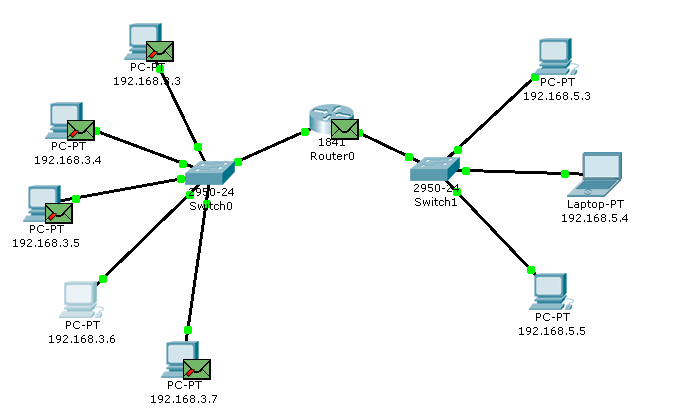


Рис. 36. Вид рабочей области

Узел 192.168.3.6 получает ARP-ответ с МАС-адресом маршрутизатора. Теперь, зная его аппаратный адрес, хост отправляет ping-запрос.

Когда ping-запрос попадает на маршрутизатор, тот не может его перенаправить не на какой из своих интерфейсов, т.к. IP-адреса его интерфейсов не совпадают с тем адресом, который указан в ping-запросе. Соответственно, этот пакет уничтожается и формируется новое ICMP-сообщение (рис. 37).

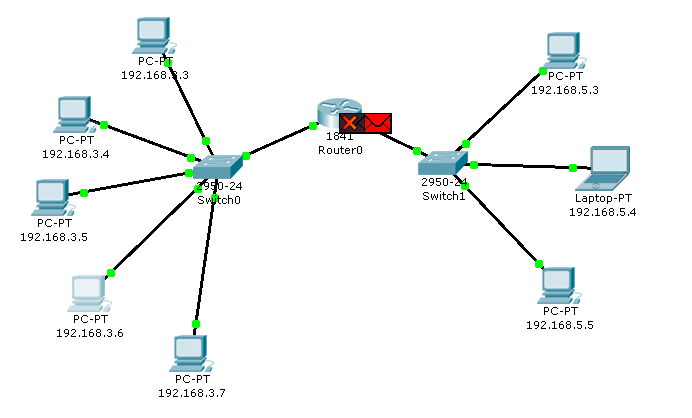


Рис. 37. Вид рабочей области

Посмотрим содержимое пакета, сформированного маршрутизатором (рис. 38).

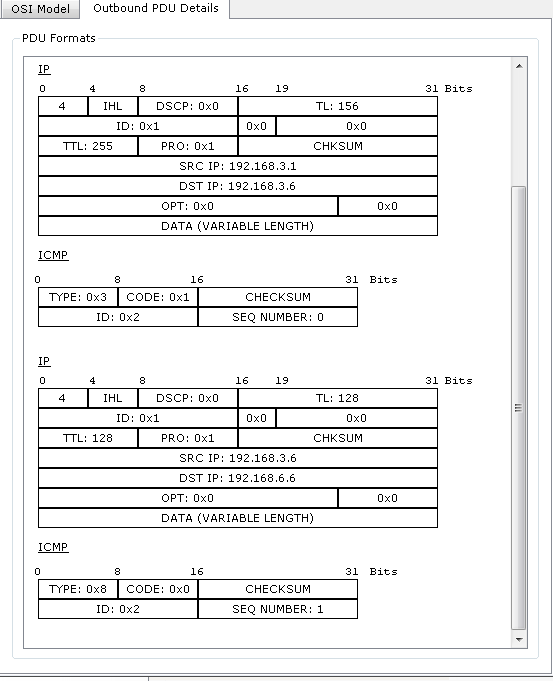


Рис. 38. Формат пакета ICMP «хост недостижим»

IP-адрес источника – 192.168.3.1. IP-адрес назначения – 192.168.3.6. Тип ICMP-сообщения – 3 с кодом 1, что означает «хост недостижим». Этот пакет приходит на узел 192.168.3.6.

Результат ping-запроса в командной строке узла 192.168.3.6: «хост назначения недостижим» (рис. 39).

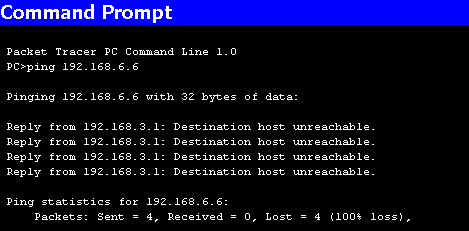


Рис. 39. Вывод программы ping

Таким образом, маршрутизатор «ответил» на ping-запрос, для которого у него не было соответствующего маршрута, новым ICMP-сообщением «хост недостижим».

**Отчет по лабораторной работе должен содержать:**

1. Топологию исследуемой сети с указанием IP-адресов составляющих ее узлов.
2. Примеры отработки протокола ARP с приведением конкретных используемых пакетов и кадров.
3. Примеры выполнения команд ping и tracert с приведением конкретных используемых пакетов и кадров.

**Лабораторная работа № 2**

**«Виртуальные локальные сети VLAN»**

**Теоретические сведения**

**VLAN –** это виртуальные сети, которые существуют на втором уровне модели OSI. То есть, VLAN можно настроить на коммутаторе второго уровня. Если смотреть на VLAN, абстрагируясь от понятия «виртуальные сети», то можно сказать, что VLAN – это просто метка в кадре, который передается по сети. Метка содержит номер VLAN (его называют VLAN ID или VID), – на который отводится 12 бит, то есть, вилан может нумероваться от 0 до 4095. Первый и последний номера зарезервированы, их использовать нельзя. Обычно, рабочие станции о VLAN ничего не знают (если не конфигурировать VLAN на карточках специально). О них думают коммутаторы. На портах коммутаторов указывается в каком VLAN они находятся. В зависимости от этого весь трафик, который выходит через порт помечается меткой, то есть VLAN. Таким образом каждый порт имеет PVID (*port vlan identifier*).Этот трафик может в дальнейшем проходить через другие порты коммутатора(ов), которые находятся в этом VLAN и не пройдут через все остальные порты. В итоге, создается изолированная среда (подсеть), которая без дополнительного устройства (маршрутизатора) не может взаимодействовать с другими подсетями.

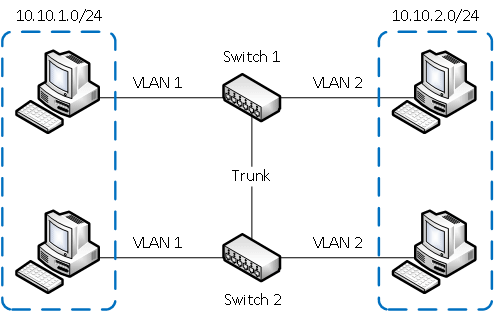
**Зачем нужны VLAN?**

* Возможность построения сети, логическая структура которой не зависит от физической. То есть, топология сети на канальном уровне строится независимо от географического расположения составляющих компонентов сети.
* Возможность разбиения одного широковещательного домена на несколько широковещательных доменов. То есть, широковещательный трафик одного домена не проходит в другой домен и наоборот. При этом уменьшается нагрузка на сетевые устройства.
* Возможность обезопасить сеть от несанкционированного доступа. То есть, на канальном уровне кадры с других VLAN будут отсекаться портом коммутатора независимо от того, с каким исходным IP-адресом инкапсулирован пакет в данный кадр.
* Возможность применять политики на группу устройств, которые находятся в одном VLAN.
* Возможность использовать виртуальные интерфейсы для маршрутизации.

**Примеры использования VLAN**

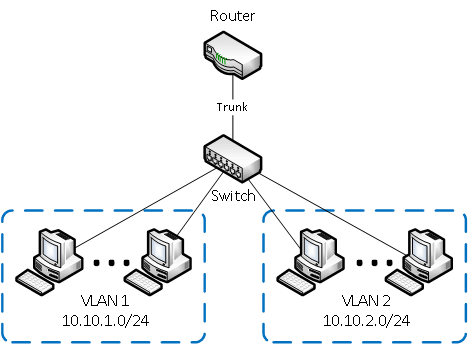
* *Объединение в единую сеть компьютеров, подключенных к разным коммутаторам*.

Допустим, у вас есть компьютеры, которые подключены к разным свитчам, но их нужно объединить в одну сеть. Одни компьютеры мы объединим в виртуальную локальную сеть *VLAN 1*, а другие – в сеть *VLAN 2*. Благодаря функции VLAN компьютеры в каждой виртуальной сети будут работать, словно подключены к одному и тому же свитчу. Компьютеры из разных виртуальных сетей *VLAN 1* и *VLAN 2* будут невидимы друг для друга.



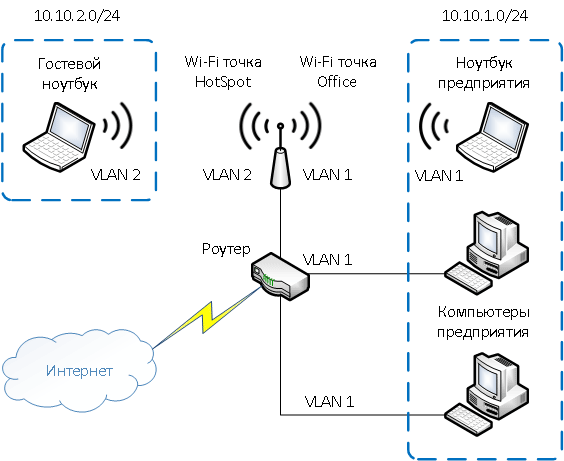
* *Разделение в разные подсети компьютеров, подключенных к одному коммутатору.*

На рисунке компьютеры физически подключены к одному свитчу, но разделены в разные виртуальные сети *VLAN 1* и *VLAN 2*. Компьютеры из разных виртуальных подсетей будут невидимы друг для друга.



* *Разделение гостевой Wi-Fi сети и Wi-Fi сети предприятия.*

На рисунке к роутеру подключена физически одна Wi-Fi точка доступа. На точке созданы две виртуальные Wi-Fi точки с названиями *HotSpot* и *Office*. К *HotSpot* будут подключаться по Wi-Fi гостевые ноутбуки для доступа к интернету, а к *Office* – ноутбуки предприятия. В целях безопасности необходимо, чтобы гостевые ноутбуки не имели доступ к сети предприятия. Для этого компьютеры предприятия и виртуальная Wi-Fi точка *Office* объединены в виртуальную локальную сеть *VLAN 1*, а гостевые ноутбуки будут находиться в виртуальной сети*VLAN 2*. Гостевые ноутбуки из сети *VLAN 2* не будут иметь доступ к сети предприятия*VLAN 1*.



**Достоинства использования VLAN**

* Гибкое разделение устройств на группы
* Как правило, одному VLAN соответствует одна подсеть. Компьютеры, находящиеся в разных VLAN, будут изолированы друг от друга. Также можно объединить в одну виртуальную сеть компьютеры, подключенные к разным коммутаторам.
* Уменьшение широковещательного трафика в сети
* Каждый VLAN представляет отдельный широковещательный домен. Широковещательный трафик не будет транслироваться между разными VLAN. Если на разных коммутаторах настроить один и тот же VLAN, то порты разных коммутаторов будут образовывать один широковещательный домен.
* Увеличение безопасности и управляемости сети
* В сети, разбитой на виртуальные подсети, удобно применять политики и правила безопасности для каждого VLAN. Политика будет применена к целой подсети, а не к отдельному устройству.
* Уменьшение количества оборудования и сетевого кабеля
* Для создания новой виртуальной локальной сети не требуется покупка коммутатора и прокладка сетевого кабеля. Однако вы должны использовать более дорогие управляемые коммутаторы с поддержкой VLAN.

Тэгированные и нетэгированные порты

Когда порт должен уметь принимать или отдавать трафик из разных VLAN, то он должен находиться в тэгированном или транковом состоянии. Понятия транкового порта и тэгированного порта одинаковые. Транковый или тэгированный порт может передавать как отдельно указанные VLAN, так и все VLAN по умолчанию, если не указано другое. Если порт нетэгирован, то он может передавать только один VLAN (родной). Если на порту не указано в каком он VLAN, то подразумевается, что он в нетэгированном состоянии в первом VLAN (VID 1).

Разное оборудование настраивается по-разному в данном случае. Для одного оборудования нужно на физическом интерфейсе указать в каком состоянии находится этот интерфейс, а на другом в определенном VLAN необходимо указать какой порт как позиционируется – с тэгом или без тэга. И если необходимо, чтобы этот порт пропускал через себя несколько VLAN, то в каждом из этих VLAN нужно прописать данный порт с тэгом. Например, в коммутаторах Enterasys Networks мы должны указать в каком VLAN находится определенный порт и добавить этот порт в egress list этого VLAN для того, чтобы трафик мог проходить через этот порт. Если мы хотим чтобы через наш порт проходил трафик еще одного VLAN, то мы добавляем этот порт в egress list еще и этого VLAN. На оборудовании HP (например, коммутаторах *ProCurve*)  мы в самом VLAN указываем какие порты могут пропускать трафик этого VLAN и добавляем состояние портов – тэгирован или нетегирован. Проще всего на оборудовании Cisco Systems. На таких коммутаторах мы просто указываем какие порты какими VLAN нетэгированы (находятся в режиме *access*) и какие порты находятся в тэгированном состоянии (находятся в режиме *trunk*).

Для настройки портов в режим trunk созданы специальные протоколы. Один из таких имеет стандарт IEEE 802.1Q. Это международный стандарт, который поддерживается всеми производителями и чаще всего используется для настройки виртуальных сетей. Кроме того, разные производители могут иметь свои протоколы передачи данных. Например, Cisco создала для свого оборудования протокол ISL (*Inter Switch Lisk*).

**Межвлановская маршрутизация**

*Что такое межвлановская маршрутизация?* Это обычная маршрутизация подсетей. Разница только в том, что каждой подсети соответствует какой-то VLAN на втором уровне. Что это значит. Допустим у нас есть два VLAN: VID = 10 и VID = 20. На втором уровне эти VLAN осуществляют разбиение одной сети на две подсети. Хосты, которые находятся в этих подсетях не видят друг друга. То есть, трафик полностью изолирован. Для того, чтобы хосты могли взаимодействовать между собой, необходимо смаршрутизировать трафик этих VLAN. Для этого нам необходимо на третьем уровне каждому из VLAN присвоить интерфейс, то есть прикрепить к ним IP-адрес. Например, для VID = 10 IP address будет 10.0.10.1/24, а для VID = 20 IP address – 10.0.20.1/24. Эти адреса будет дальше выступать в роли шлюзов для выхода в другие подсети. Таким образом, мы можем трафик хостов с одного VLAN маршрутизировать в другой VLAN. Что дает нам маршрутизация VLAN по сравнению с простой маршрутизацией посетей без использования VLAN? А вот что:

* Возможность стать членом другой подсети на стороне клиента заблокирована. То есть, если хост находится в определенном VLAN, то даже, если он поменяет себе адресацию с другой подсети, он всеравно останется в том VLAN,  котором он был. Это значит, что он не получит доступа к другой подсети. А это в свою очередь обезопасит сеть от «плохих» клиентов.
* Мы можем поместить в VLAN несколько физических интерфейсов коммутатора. То есть, у нас есть возможность на коммутаторе третьего уровня сразу настроить маршрутизацию, подключив к нему клиентов сети, без использования внешнего маршрутизатора. Либо мы можем использовать внешний маршрутизатор подключенный к коммутатору второго уровня, на котором настроены VLAN, и создать столько сабинтерфейсов на порте маршрутизатора, сколько всего VLAN он должен маршрутизировать.
* Очень удобно между первым и третьим уровнями использовать второй уровень в виде VLAN. Удобно подсети помечать как VLAN с определенными интерфейсами. Удобно настроить один VLAN и поместить в него множество портов коммутатора.

**Ход выполнения работы**

***VLAN с одним коммутатором***

Для рисования ПК выбираем в конечных устройства настольный компьютер и, удерживая Ctrl, (так быстрее) нажмите 1 раз на ПК а затем рисуйте нужное кол-во ПК, щелкая мышкой (рис. 1). Этим приемом вы сможете за один раз нарисовать сразу 4 ПК.

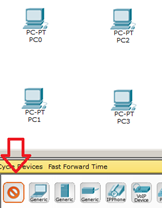
****

Рис. 1 – Выбор устройств, удерживая Ctrl

Устанавливаем коммутатор и, удерживая Ctrl, создаем подключение прямым кабелем, выбирая порты коммутатора. После инициализации портов все лампы загорятся зеленым. На схему будет две подсети (Рис. 2) .

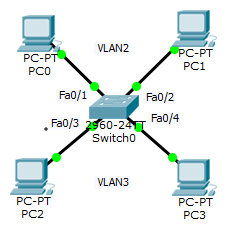


Рис. 2 – Две подсети: VLAN2 и VLAN3

Примечание

Имя VLAN1 используется по умолчанию, его лучше в нашем примере не использовать.

На коммутаторе набираем команду **en** и входим в привилегированный режим. Затем набираем команду **conf t** для входа в режим глобального конфигурирования. Если подвести курсор мыши к портам коммутатора, то вы увидите какие порты в каком сегменте задействованы. Для VLAN3 – это Fa0/3 и Fa0/4 (предположим, что это будет бухгалтерия - buh) и для VLAN2 – это Fa0/1 и Fa0/2 (предположим, что это будет склад – sklad). Сначала будем конфигурировать второй сегмент сети VLAN2 (sklad) – рис. 3.

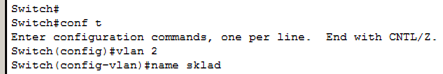


Рис. 3 – VLAN2 получает имя sklad

В виртуальной сети VLAN2 настраиваем порты коммутатора Fa0/1 и Fa0/2 как **access** порты, т.е. порты для подключения пользователей ( рис. 4).

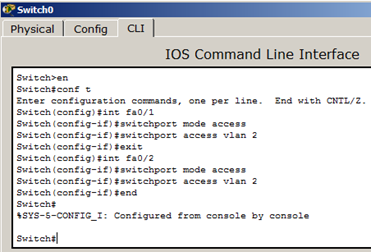


Рис. 4. Указываем порты коммутатора для подключения пользователей

Теперь командой **show vlan** можно проверить результат ( рис. 5).

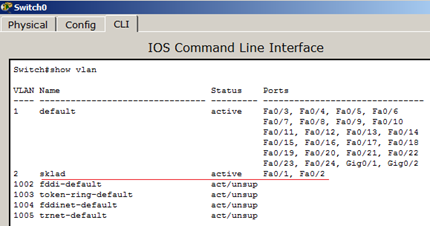


Рис. 5. Подсеть VLAN2 склад настроена

Далее работаем с VLAN3 (рис. 6).

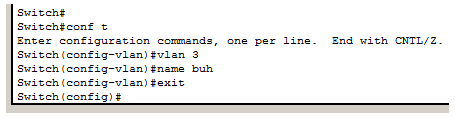
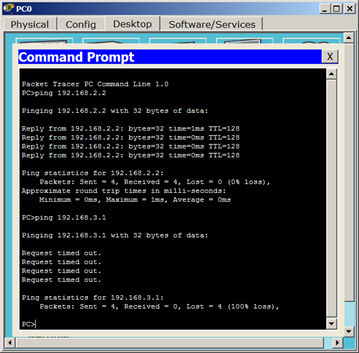


Рис. 6. VLAN3 получает имя buh

В виртуальной сети VLAN3 настраиваем порты коммутатора Fa0/3 и Fa0/4 как **access** порты, т.е. порты для подключения пользователей, затем командой **show vlan** можно проверить и убедиться, что мы создали в сети 2 сегмента на разные порты коммутатора (рис. 7).



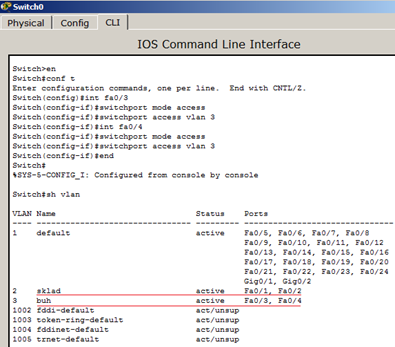


Рис. 7. Настроены VLAN2 и VLAN3

Настраиваем IP адреса компьютеров – для VLAN2 из сети 192.168.2.0, а для VLAN3 из сети 192.168.3.0 (рис. 8).

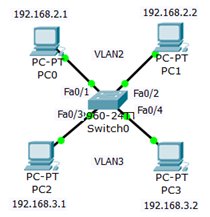


Рис. 8. Настраиваем IP адреса компьютеров

Проверяем связь ПК в пределах VLAN и отсутствие связи между VLAN2 и VLAN3 (рис. 9).

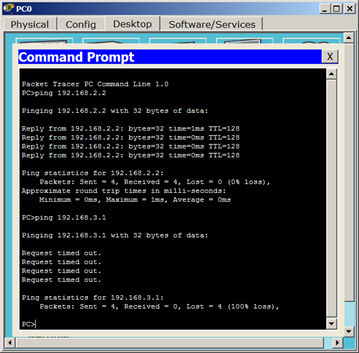


Рис. 9. Все работает, как необходимо

Итак, на компьютере ПК0 мы убедились, что компьютер в своем сегменте видит ПК, а в другом сегменте – нет.

***Настройка виртуальной сети на коммутаторе 2960***

В данной работе рассматривается настройка *VLAN* на коммутаторе фирмы Сisco в программе CPT. Мы уже делали подобную работу. Но здесь мы не только закрепим пройденное, но и узнаем ряд новых команд Cisco *IOS*.

Создайте *сеть*, *топология* которой представлена на рис. 10.

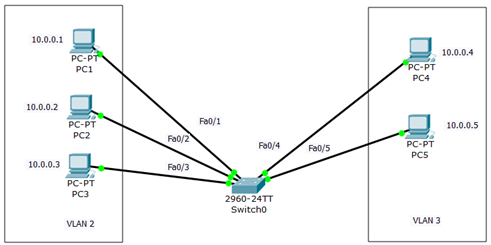


Рис. 10. Схема сети с одним коммутатором

Задача данной работы является создание 2х независимых групп компьютеров: ПК1-ПК3 должны быть доступны только друг для друга, а вторая независимая *группа* - компьютеры ПК4 и ПК5.

**Настройка коммутатора**

Первоначально сформируем VLAN2. Дважды щелкните левой кнопкой мыши по коммутатору. В открывшемся окне перейдите на вкладку **CLI**. Вы увидите окно консоли. Нажмите на клавишу **Enter** для того, чтобы приступить к вводу команд. Перейдем в привилегированный режим, выполнив команду **enable**:

**Switch>en**

По умолчанию все ПК объединены в VLAN1. Для реализации сети, которую мы запланировали, создадим на коммутаторе еще два VLAN (2 и 3). Для этого в привилегированном режиме выполните следующую команду для перехода в режим конфигурации:

**Switch#conf t**

Теперь вводим команду **VLAN 2**. Данной командой вы создадите на коммутатореVLANс номером 2. Указатель ввода Switch (config)# изменится на Switch (config-vlan)# это свидетельствует о том, что вы конфигурируете уже не весь коммутатор в целом, а только отдельныйVLAN, в данном случаеVLAN номер 2 (рис. 11).

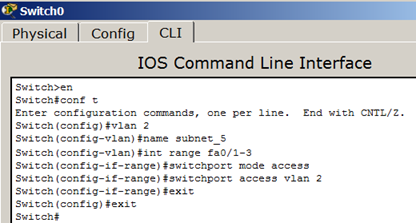


Рис. 11. Листинг команд для формирования VLAN2

**Примечание**

Командой VLAN2, мы создаем на коммутаторе новый VLAN с номером 2. Команда **name subnet\_5** присваивает имя subnet\_5 виртуальной сети номер 2. Выполняя команду **interface range fast Ethernet 0/1-3** мы переходим к конфигурированию интерфейсов fastEthernet 0/1, fastEthernet 0/2 и fastEthernet 0/3 коммутатора. Слово range в данной команде, указывает на то, что мы будем конфигурировать не один порт, а диапазон портов. Команда **switch port mode access** конфигурирует выбранный порт коммутатора, как порт доступа (access порт). Команда **switch port access vlan 2** указывает, что данный порт является портом доступа для VLAN номер 2.

Выйдите из режима конфигурирования, дважды набрав команду **exit** и просмотрите результат конфигурирования (рис. 12), выполнив команду **sh vl br**. Как видим, на коммутаторе появился VLAN с номером 2 и именем subnet\_5, портами доступа которого являются fastEthernet 0/1, fastEthernet 0/2 и fastEthernet 0/3.

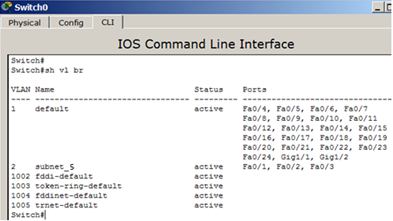


Рис. 12. Просмотр информации о VLAN на коммутаторе

**Примечание**

Команда **shvlbr** выводит информацию о существующих на коммутаторе VLAN-ах. В результате выполнения команды на экране появится: **номера VLAN**(первый столбец), **название VLAN** (второй столбец), **состояние VLAN** (работает он или нет) – третий столбец, **порты**, принадлежащие к данному VLAN (четвертый столбец).

Далее аналогичным образом создадим **VLAN 3** с именем **subnet\_6** и сделаем его портами доступа интерфейсы fastEthernet 0/4 и fastEthernet 0/5. Результат показан на рис. 13.

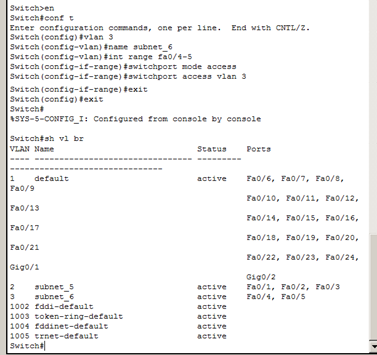


Рис. 13. Результат – настройка на коммутатореVLAN2 и VLAN3

**Проверка результатов работы**

Сеть настроена и нужно ее протестировать. Результат положителен, если в пределах своей VLAN компьютеры доступны, а компьютеры из разных VLAN не доступны (рис. 14). У нас все пять компьютеров находя в одной сети 10.0.0.0/8, но они находятся в разных виртуальных локальных сетях.

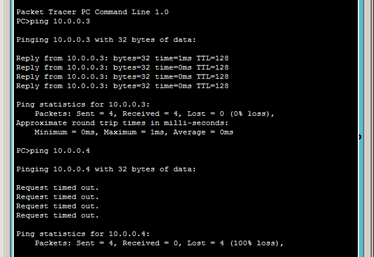


Рис. 14. Пинг с PC1 на PC3 и PC4

***VLAN с двумя коммутаторами. Разделяемый общий канал (транк)***

На практике часто возникает задача разделения устройств, подключенных к одному или нескольким коммутаторам на несколько непересекающихся локальных сетей. В случае, если используется только один *коммутатор*, то эта задача решается путем конфигурирования портов коммутатора, указав каждому порту к какой локальной сети он относится. Если же используется несколько коммутаторов (рис. 15), то необходимо между коммутаторами помимо данных передавать информацию к какой локальной сети относится *кадр*. Для этого был разработан стандарт 802.1Q.

.

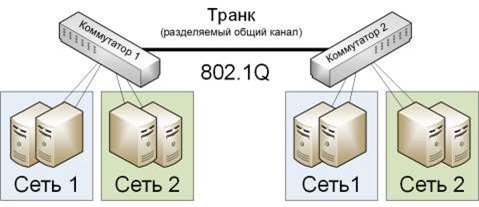


Рис. 15. Виртуальные локальные сети (VLAN) с использованием двух коммутаторов

От теории перейдем к практике и произведем дублирование нашей сети (той, которая была показана ранее на рис. 10). Для этого выделим всю *сеть* инструментом **Select** (Выделить), и, удерживая клавишу **Ctrl**, перетащим на новое *место* в рабочей области программы. Так мы произведем *копирование* (рис. 16).

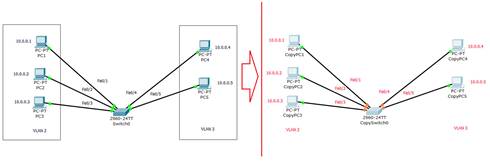


Рис. 16. Дублируем сеть с одним коммутатором

Соединим коммутаторы перекрестным кабелем (кроссом) через самые производительные порты – *Gigabit* *Ethernet* (рис.17).

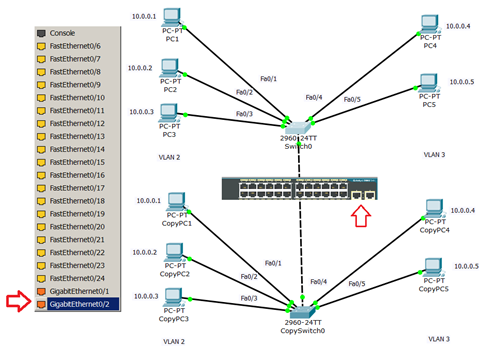


Рис. 17. Соединяем коммутаторы через Gigabit Ethernet порты

Теперь поправим настройки на дубликате исходной сети (рис. 18).

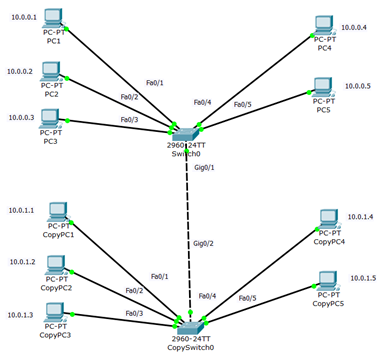


Рис. 18. Настраиваем сеть-дубликат

Укажем новый вариант подсетей VLAN2 и VLAN3, а также выделим **trunk (транк)** *связь* коммутаторов (рис. 19).

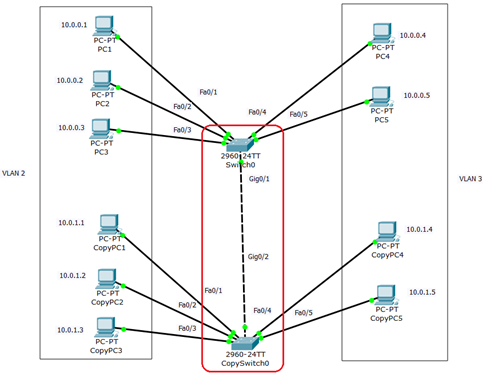


Рис. 19. В сети обозначаем подсети VLAN2 и VLAN3

**Настраиваем транк порт Gig0/1**

При настройке Gig0/1 на коммутаторе Switch0 мы меняем состояние порта и указываем vlan 2 и 3 для работы с ним (рис.20).

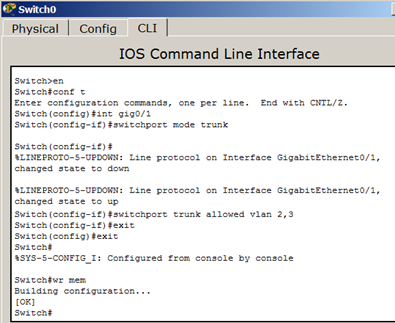


Рис. 20. Настраиваем транк порт Gig0/1 на коммутаторе Switch0

**Настраиваем транк порт Gig0/2**

Транк порт Gig0/2 на коммутаторе CopySwitch0 настраиваем аналогично (рис. 21).

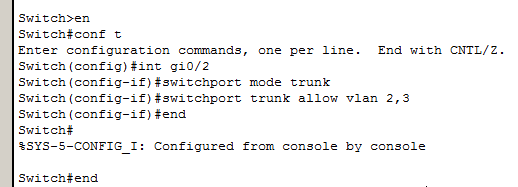


Рис. 21. Настраиваем trunk порт Gig0/2 на коммутаторе CopySwitch0

**Диагностика результатов работы**

Проверяем пинг с PC1 в разные vlan (рис. 22).Все отлично: в пределах своей vlan ПК доступны, а между ПК разных vlan связи нет.

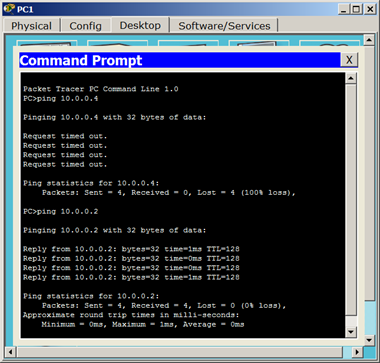


Рис. 22. Пинг с PC1 в разные vlan

**Лабораторная работа № 3**

**«****Настройка сетевых сервисов»**

Эмулятор Cisco Packet Tracer позволяет проводить настройку таких сетевых сервисов, как: DHCP, HTTP, DNS в составе сервера сети.

* **HTTP** – позволяет строить примитивные веб-страницы и проверять прохождение пакетов на 80-ый порт сервера.
* **DHCP** – позволяет организовывать пулы сетевых настроек для автоматического конфигурирования сетевых интерфейсов.
* **DNS** – позволяет организовать примитивную службу разрешения доменных имён.

Рассмотрим указанные сетевые службы.

**Служба DHCP**

Название протокола DHCP (Dynamic Host Configuration Protocol) дословно расшифровывается как «Протокол динамической конфигурации узла». Данный сетевой протокол работает на прикладном уровне модели OSI и позволяет компьютерам сети автоматически получать IP-адрес и другие параметры, необходимые для работы в сети TCP/IP.

Наиболее часто используемые опараметры:

* IP-адрес маршрутизатора по умолчанию;
* маска подсети;
* адреса серверов DNS.

IP-адреса могут назначаться администратором сети вручную. Это представляет для администратора утомительную процедуру. Протокол *Dynamic Host Configuration Protocol (DHCP)* был разработан для того, чтобы освободить администратора от этих проблем.

При автоматическом статическом способе DHCP-сервер присваивает IP-адрес (и, возможно, другие параметры конфигурации клиента) из пула наличных IP-адресов без вмешательства оператора. Границы пула назначаемых адресов задает администратор при конфигурировании DHCP-сервера. Между идентификатором клиента и его IP-адресом по-прежнему, как и при ручном назначении, существует постоянное соответствие. Оно устанавливается в момент первичного назначения сервером DHCP IP-адреса клиенту. При всех последующих запросах сервер возвращает тот же самый IP-адрес.

При динамическом распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время, что дает возможность впоследствии повторно использовать IP-адреса другими компьютерами. Динамическое разделение адресов позволяет строить IP-сеть, количество узлов в которой намного превышает количество имеющихся в распоряжении администратора IP-адресов.

Все устройства в сети при работе с протоколом DHCP можно разделить на два вида: DHCP сервера и DHCP клиенты. DHCP клиенты пытаются получить настройки, а DHCP сервера выдают их.

Рассмотрим, как работает данный протокол, на примере следующей топологии сети.

|  |
| --- |
|  |

Рис. 1

Имеется некоторая сеть (рис. 1), в которой существует DHCP сервер. Все компьютеры и DHCP сервер связываются друг с другом через коммутатор. К данной сети подключают еще один новый компьютер. Зная, что в сети существует DHCP сервер, в его настройках указывают получать IP адрес автоматически. После этого новый компьютер попытается получить IP адрес от DHCP сервера. Для этого он выполняет широковещательный запрос на IP адрес 255.255.255.255, а в качестве своего IP-адреса указывает 0.0.0.0 (так как у него еще нет IP адреса). В ходе данного широковещательного запроса рассылается сообщение DHCPDISCOVER. Данное сообщение содержит в себе информацию, позволяющую отличить его от других типов запросов/сообщений (то есть указывает на то, что это сообщение предназначено для DHCP сервера, для получения IP адреса), MAC адрес устройства сформировавшего запрос, а также предыдущий IP адрес устройства (если он у него был).

|  |
| --- |
|  |

Рис. 2. Процесс рассылки сообщения DHCPDISCOVER

Так как сообщение DHCPDISCOVER рассылается широковещательным способом, оно попадает не только на DHCP сервер, но и на другие устройства данного сегмента сети, но так как в сообщение DHCPDISCOVER указывается, что оно предназначено только для DHCP сервера, остальные устройства сети отвергают данное сообщение.

При получении сообщения DHCPDISCOVER DHCP сервером, он анализирует его содержание и в соответствии со своими настройками выбирает подходящую конфигурацию для запросившего компьютера и отправляет ее обратно в сообщении DHCPOFFER. Обычно сообщение DHCPOFFER отсылается только на MAC адрес компьютера, который был указан в сообщении DHCPDISCOVER, но иногда оно может рассылаться и методом широковещательной рассылки.

|  |
| --- |
|  |

Рис. 3. DHCP сервер отвечает сообщением DHCPOFFER

В случае если в сети существует несколько DHCP серверов, компьютер может получить в ответ на сообщение DHCPDISCOVER несколько сообщений DHCPOFFER от разных DHCP серверов. Из них компьютер выбирает одно, обычно полученное первым. И отвечает на него сообщением DHCPREQUEST, которое содержит в себе всю ту же информацию, что и сообщение DHCPDISCOVER + IP адрес выбранного DHCP сервера. Сообщение DHCPREQUEST рассылается широковещательным методом, для того чтобы его могли получить все DHCP сервера сети, если их несколько.

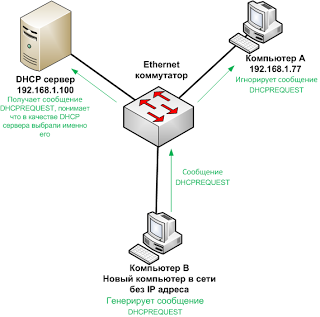


Рис. 4. Рассылка сообщения DHCPREQUEST

Все устройства сети, не являющиеся DHCP серверами игнорируют сообщение DHCPREQUEST. Сервера DHCP, IP адрес которых не содержится в сообщении DHCPREQUEST, понимают, что их не выбрали в качестве DHCP сервера. DHCP сервер, IP адрес которого указан в сообщении DHCPREQUEST, получает его и понимает, что именного его выбрали в качестве DHCP сервера для нового компьютера, на что он отвечает сообщением DHCPACK, которое как бы подтверждает данный выбор. Сообщение DHCPACK отправляется на MAC адрес компьютера, указанного в сообщении DHCPREQUEST.

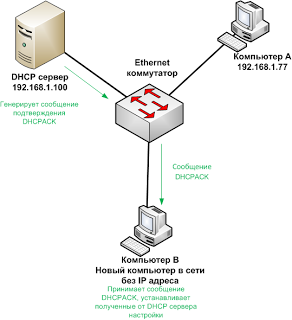


Рис. 5. Отсылка подтверждающего сообщения DHCPACK

Компьютер, запрашивающий конфигурацию, получает сообщения DHCPACK и применяет конфигурацию, которая была получена в сообщении DHCPOFFER. Вот так путем несложного обмена сообщениями функционирует протокол DHCP.

DHCP сервер может быть настроен по разному, и, в зависимости от его конфигурации, он будет выдавать IP адреса запрашивающим компьютерам разными способами. Например, можно настроить DHCP сервер так, чтобы он выдавал запросившим компьютерам любые свободные IP адреса из некоторого диапазона, а можно настроить так, чтобы он выдавал определенные IP адреса устройствам с заданными MAC адресами.

В роли DHCP сервера может выступать сервер под управлением серверной ОС семейства Linux или Windows, некоторые модели коммутаторов, маршрутизаторов и даже обычные компьютеры с клиентскими операционными системами, в случае если на них установлено специализированное программное обеспечение. Обычно под DHCP сервера не отводят отдельного физического сервера или отдельной виртуальной машины, а устанавливают их на одном из уже существующих не сильно загруженных серверов, выполняющих другую роль.

**Служба DNS**

Для определения IP-адреса по доменному имени используется служба DNS, состоящая из множества DNS-серверов, содержащих распределенную базу отображений "доменное имя - IP адрес". В каждой сети должен быть хотя бы один DNS-сервер, который поддерживает локальную базу данных доменных имен и выполняет поиск IP-адреса по доменному имени. Этот поиск осуществляется следующим образом:

* выполняется запрос к локальному DNS-серверу;
* если DNS-сервер знает ответ, то возвращает его клиенту (соответствующая запись находится в его таблице или в кэш-памяти);
* если DNS-сервер не знает ответа, то обращается по ссылке к следующему DNS-серверу, пока соответствующая запись не будет найдена (рекурсивная схема).

**Выполнение работы**

Создайте схему сети, представленную на рис. 6.

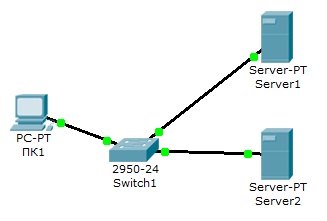


Рис. 6. Схема сети.

***Задание:*** Настроить сеть следующим образом:

1) На сервере1 организовать службу DNS и Web сервер.

2) На сервере2 организовать DHCP сервер.

3) Компьютер ПК1 получает параметры протокола TCP/IP c DHCP сервера и открывает сайт www.rambler.ru на сервере1.

В конфигурации серверов проведите следующие настройки IP адресов:

Server1: IP адрес – 10.0.0.1, маска подсети – 255.0.0.0, DNS Server – 10.0.0.1

Server2: IP адрес – 10.0.0.2, маска подсети – 255.0.0.0

Войдите в конфигурацию рабочей станции ПК1 и включите «Настройка IP» через DHCP сервер (рис. 7).

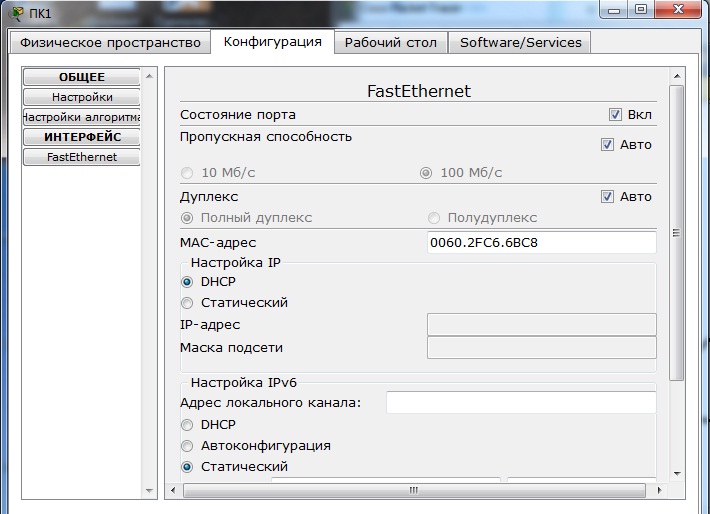


Рис. 7. Настройка IP на рабочей станции ПК1.

**Настройка DNS сервера**

Для настройки DNS сервера необходимо:

* Включить на сервере поддержку DNS сервиса в программе Packet Tracer;
* Добавить доменное имя HTTP сервера и его IP адрес, для дальнейшей работы нашего DNS сервера с компьютерами в сети;
* Указать IP адрес и маску подсети сервера;
* Указать шлюз, через который наш сервер соединен с остальной сетью (в данной сети шлюз не используется).

В конфигурации сервер1 войдите на вкладку DNS и в ресурсной записи типа А (A Record) свяжите доменное имя компьютера (server1.rambler.ru) с его IP адресом (рис. 8) и нажмите кнопку ДОБАВИТЬ.

**Ресурсная запись** – это единица хранения и передачи информации в DNS. Каждая такая запись несет в себе **информацию соответствия** какого-то имени и служебной информации в DNS, например соответствие имени домена – IP адреса. Запись типа **A** – (address record/запись адреса) отображают имя хоста (доменное имя) на адрес IPv4.

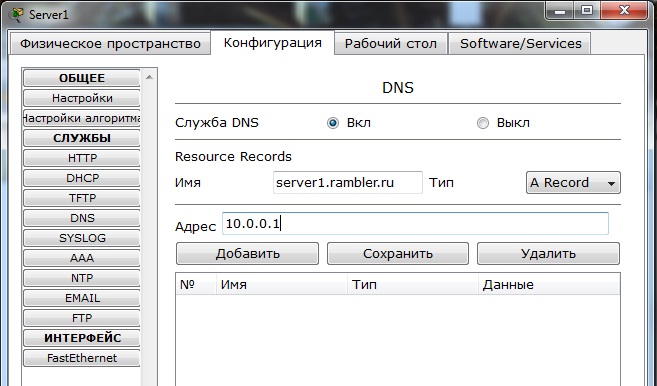


Рис. 8. Ввод ресурсной записи типа А.

В ресурсной записи типа CNAME свяжите псевдоним сайта с компьютером (рис. 9).

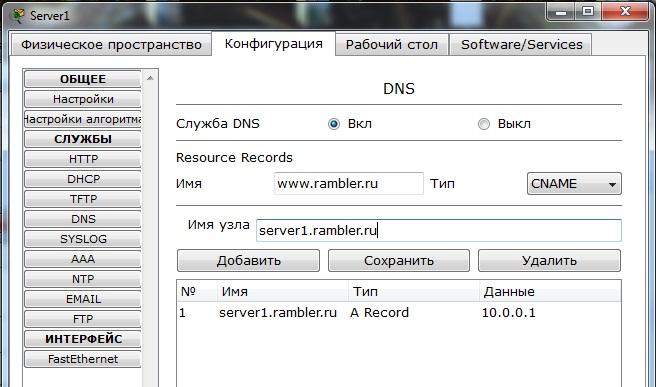


Рис. 9. Ввод ресурсной записи типа CNAME.

Включите командную строку на сервер1 и проверьте работу службы DNS. Для проверки прямой зоны DNS сервера введите команду

SERVER>**nslookup www.rambler.ru**

Если все правильно, то вы получите отклик, представленный на рис. 10 с указанием полного доменного имени DNS сервера в сети и его IP адрес.

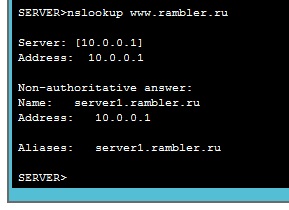


Рис. 10. Проверка прямой зоны DNS.

**Настройка DHCP сервера**

Войдите в конфигурацию сервер2 и на вкладке DHCP настройте службу (рис. 10):

При настройке необходимо исключить (зарезервировать) некоторые адреса. Исключаем из раздаваемых сервером DHCP адресов адреса серверов. Серверам не стоит выдавать адреса по DHCP – к ним часто обращаются, поэтому адрес должен быть введен статически и никогда не меняться.

Зарезервируем первые девять адресов (начальный адрес 10.0.0.10). Раздаваемые адреса образуют пул с именем в нашем случае serverPool. Вводим начальный IP адрес из числа раздаваемых, маску подсети и максимальное количество пользователей.

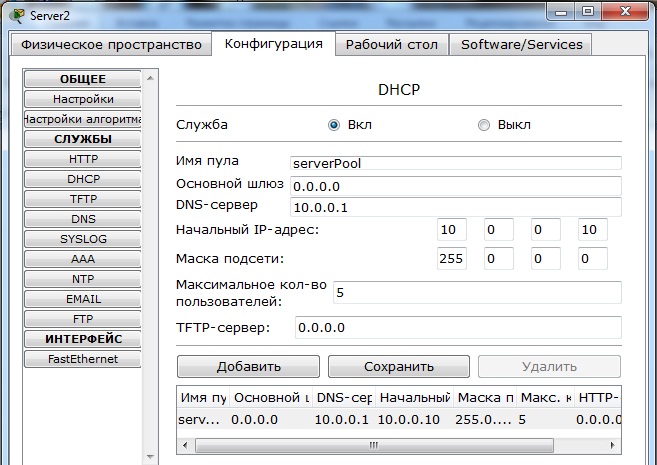


Рис. 11. Настройка DHCP сервера.

Проверьте работу DHCP клиента. Войдите в конфигурации компьютера ПК1 на рабочий стол, выберите настройку IP адреса. Включите кнопку DHCP. Будет сформирован запрос к DHCP серверу и получен ответ с IP адресом данного компьютера.

Добавьте на схему еще один компьютер ПК2. Войдите в конфигурацию ПК1 на рабочий стол и в командной строке (Command promt) сконфигурируйте протокол TCP/IP.

1) Командой PC>**ipconfig /release** сбросьте старые параметры IP адреса.

2) Командой PC>**ipconfig /renew** получите параметры с DHCP сервера (рис. 11).

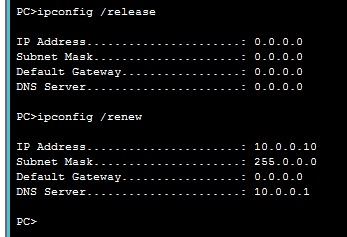


Рис. 12. Конфигурация протокола TCP/IP клиента.

### Создание WEB-документов на сервере1

Создаём HTTP-сервер. Открываем вкладку [1] и создаём первую страницу, она называется index.html [3]. Необходимо, чтобы служба HTTP была включена [2].

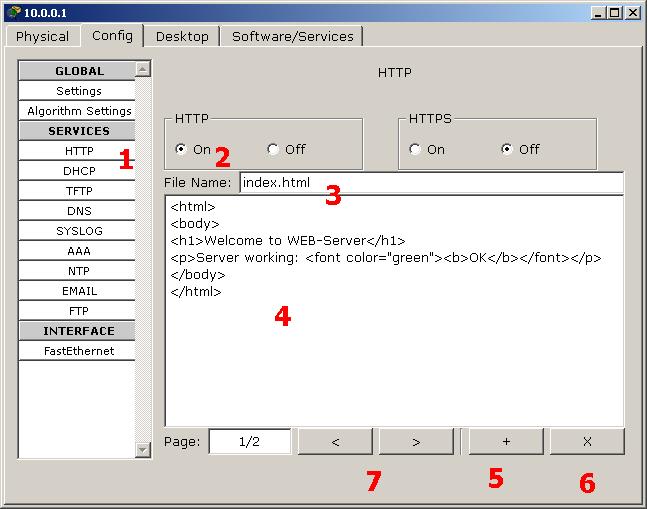


Рис. 13. Вкладка HTTP

В окно кода [4] вводим следующий текст для проверки (это простой HTML-документ):

<html>

<body>

<h1>Welcome to WEB-Server</h1>

<p>Server working: <font color=”green”><b>OK</b></font></p>

</body>

</html>

При желании можно добавить новую страницу [5] или удалить текущую [6]. Переключение между страницами осуществляется кнопками [7].

### Проверяем работоспособность HTTP-сервера

Для того чтобы проверить работоспособность сервера, открываем клиентскую машину ПК1 и на вкладке “Рабочий стол” открываем приложение Web Browser. После чего в окошке URL набираем адрес WEB-сервера (пока только адрес, позднее с настройкой DNS можно будет набирать и доменное имя).

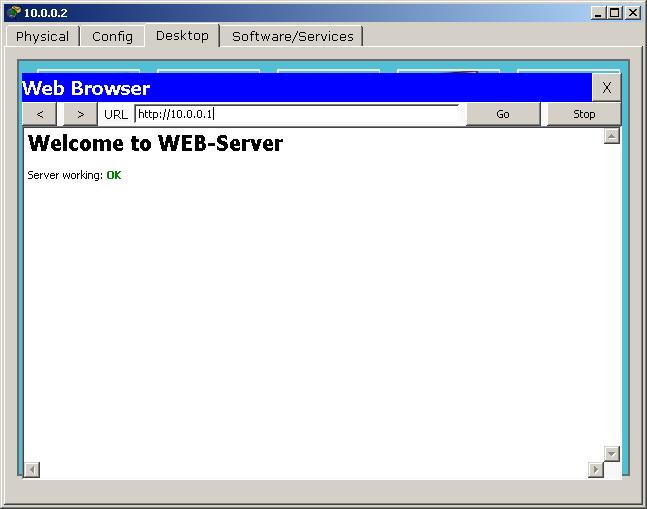


Рис. 14. WEB-сервер работает