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

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

П. И. Балтрукович, М. В. Тумилович, О. С. Медведев

ПРИКЛАДНЫЕ ЗАДАЧИ ПРИМЕНЕНИЯ СЕТЕВЫХ ТЕХНОЛОГИЙ. ПРАКТИКУМ

Рекомендовано УМО по образованию в области информатики и радиоэлектроники в качестве учебно-методического пособия для специальностей 6-05-0611-01 «Информационные системы и технологии», 6-05-0612-01 «Программная инженерия»

Под общей редакцией Т. В. Казак

Минск БГУИР 2026

УДК 004.738(076)
ББК 32.971.35я73
Б20

Р е ц е н з е н т ы:

кафедра инфокоммуникационных технологий
учреждения образования «Белорусская государственная академия связи»
(протокол № 13 от 11.02.2025);

главный научный сотрудник государственного учреждения
«Научно-исследовательский институт Вооруженных Сил Республики Беларусь»
доктор технических наук, профессор С. А. Савенко

Балтрукович, П. И.

Б20 Прикладные задачи применения сетевых технологий. Практикум : учеб.-метод. пособие / П. И. Балтрукович, М. В. Тумилович, О. С. Медведев ; под общ. ред. Т. В. Казак. – Минск : БГУИР, 2026. – 120 с. : ил.

ISBN 978-985-543-843-5.

Содержит теоретические материалы и задания к практическим занятиям по дисциплине «Сетевые технологии».

УДК 004.738(076)
ББК 32.971.35я73

ISBN 978-985-543-843-5

Содержание

Практическое занятие № 1
«Принципы построения, алгоритмы и топологии локальных сетей с использованием коммутаторов»

Цель занятия: изучить работу протокола покрывающего дерева, закрепить базовые навыки работы с Cisco Packet Tracer.

1.1. Краткие теоретические сведения

1.1.1. Основные сведения

Разделяемой средой (shared medium) называется физическая среда передачи данных, к которой непосредственно подключено несколько передатчиков узлов сети. **Особенностью такой среды** является то, что передачу информации в определенный момент времени может осуществлять только одно устройство. Остальные участники сети получают доступ к каналу последовательно, что исключает одновременную передачу данных разными узлами. Данная технология стала основой локальных сетей с момента их возникновения.

Ее ключевое преимущество – относительная простота реализации, требующая минимальных затрат на оборудование.

Однако **существенным недостатком** выступает ограниченная масштабируемость – при добавлении новых устройств общая пропускная способность сети распределяется между ними, что приводит к снижению производительности.

Для преодоления этих ограничений используется сегментирование сети с помощью коммутаторов. Каждый сегмент получает собственную разделяемую среду, а коммутаторы обеспечивают связь между ними. Это решение позволяет:

- локализовать трафик в пределах сегментов;
- уменьшить нагрузку на отдельные участки сети;
- повысить общую пропускную способность.

Работа коммутатора основана на анализе MAC-адресов – уникальных идентификаторов сетевых интерфейсов (например, 00:1A:2P:3C:4D:5E). Устройство автоматически создает таблицу соответствия адресов и портов, изучая входящие пакеты данных. Эта таблица доступна для просмотра через команду **show mac-address-table**.

Если адрес получателя неизвестен, коммутатор использует широковещательную рассылку через все активные порты. После получения ответа от целевого устройства информация о его расположении фиксируется в таблице MAC-адресов. Важной особенностью является поддержка нескольких адресов на одном порту – это необходимо при подключении через концентратор, объединяющий несколько устройств.

1.1.2. Принципы функционирования сетевых коммутаторов

Основу работы современных коммутаторов составляет **стандарт IEEE 802.1D**, определяющий алгоритмы прозрачного моста.

Коммутатор функционирует по следующей схеме: поступающий кадр полностью сохраняется в буферной памяти порта, анализируется (преимущественно по MAC-адресу получателя) и после этого перенаправляется через соответствующий интерфейс.

Буферизация кадров устраняет зависимость от общей разделяемой среды, уменьшая домены коллизий. Кроме того, при подключении отдельных устройств (компьютеров, серверов, концентраторов) к каждому порту реализуется **микросегментация**.

В полудуплексном режиме зона коллизий ограничивается связью «коммутатор – узел», а в **дуплексном режиме** коллизии исключаются полностью.

Если к порту подключен концентратор с несколькими узлами, домен коллизий расширяется на весь сегмент концентратора и линию связи с коммутатором.

Механизмы адресной таблицы

Коммутатор автоматически формирует таблицу коммутации, фиксируя соответствие MAC-адресов источников и портов. Этот **процесс самообучения** основан на анализе входящего трафика:

• при получении кадра адрес источника заносится в таблицу (или обновляется существующая запись);

• адрес назначения проверяется в таблице для определения целевого порта;

• если порт назначения свободен, кадр передается, при занятости порта данные сохраняются в буфере до освобождения линии.

При совпадении портов источника и получателя кадр отбрасывается (происходит фильтрация).

Если адрес назначения отсутствует в таблице, коммутатор рассылает кадр на все порты, кроме исходного (Flooding – подтопление сети). Этот механизм активируется также для широковещательных MAC-адресов, обеспечивая совместимость со стандартными сетевыми протоколами.

Различают следующие **типы записей в таблице коммутации**:

- **динамические записи –** создаются автоматически, имеют ограниченное время жизни (обычно 5–15 мин). Позволяют адаптироваться к изменениям топологии (перемещение устройств, замена оборудования);
- **статические записи**, которые вводятся администратором вручную и не имеют срока действия. Используются для управления доступом или оптимизации маршрутизации.

Изначально пустая таблица заполняется в процессе работы. Неизвестные адреса обрабатываются через Flooding, что особенно характерно на этапе инициализации. Со временем коммутатор накапливает информацию о сетевых узлах, минимизируя избыточную передачу данных.

Несмотря на интеллектуальную обработку кадров, коммутаторы сохраняют **широковещательную рассылку** как обязательный элемент сетевых взаимодействий, обеспечивая работу **ARP** (Address Resolution Protocol – протокол разрешения адресов), **DHCP** (Dynamic Host Configuration Protocol, клиент-серверный протокол динамической конфигурации хоста) и других протоколов канального уровня, где **ARP** – это сетевой протокол, используемый для установления соответствия между IP-адресом устройств и их физическими MAC-адресами в локальных сетях. Протокол ARP работает на **канальном уровне** модели OSI и играет ключевую роль в организации взаимодействия устройств – когда одному узлу требуется отправить пакет другому, он сначала определяет MAC-адрес получателя, используя его известный IP-адрес. Это позволяет обеспечить точную доставку кадров через коммутаторы и маршрутизаторы, сохраняя целостность сетевой коммуникации.

В результате применение коммутаторами широковещательной рассылки сочетает автоматизацию и гибкость управления, делая коммутаторы ключевым элементом в построении эффективных локальных сетей.

Широковещательные штормы

Базовый механизм работы коммутаторов, обрабатывающих кадры с широковещательными MAC-адресами, создает риски возникновения **сетевых петель** (состояние постоянной пересылки кадров) между ними при наличии избыточных соединений, что провоцирует бесконтрольную циркуляцию кадров и перегрузку трафиком – явление, известное как широковещательный шторм.

Сетевая петля (коммутационная петля) возникает, когда кадры бесконечно передаются между устройствами, подключенными к одному сегменту. Это парализует сеть, делая ее недоступной.

Рассмотрим пример с двумя коммутаторами, соединенными двумя каналами (рис. 1.1).

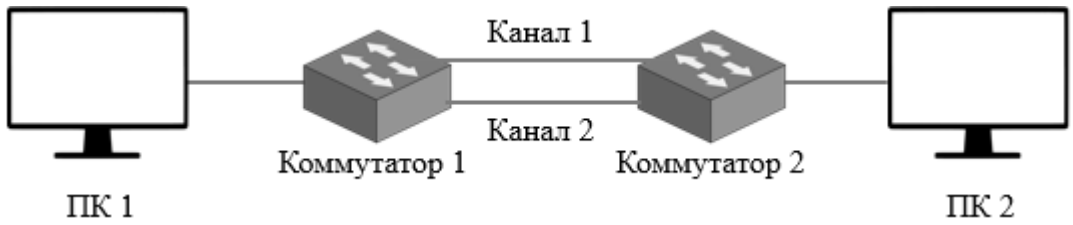

Рис. 1.1. Формирование коммутационной петли в сети с двумя коммутаторами

Если компьютер 1, подключенный к коммутатору 1, в первый раз отправляет данные на компьютер 2 через коммутатор 2, их MAC-адреса отсутствуют в таблицах коммутаторов. Тогда происходит следующее.

Коммутатор 1, не найдя запись о компьютере 2, отправляет кадр через все порты, кроме исходного. Оба канала передают кадр коммутатору 2, который, также не имея информации о получателе, дублирует его на все интерфейсы. В результате кадры начинают циркулировать между устройствами, создавая коммуникационную петлю.

Цикл прерывается, только когда компьютер 2 начинает отвечать, и, соответственно, коммутаторы обновляют таблицы.

Однако широковещательные кадры (например, **ARP**-запросы) не имеют конкретного получателя, поэтому их циркуляция становится бесконечной. Это приводит к перегрузке сети – **коммутаторы тратят ресурсы на обработку дублирующихся кадров, блокируя полезный трафик**.

В сетях с большим количеством коммутаторов, образующих коммутационные петли (например, четыре устройства на рис. 1.2), **риск шторма возрастает**.

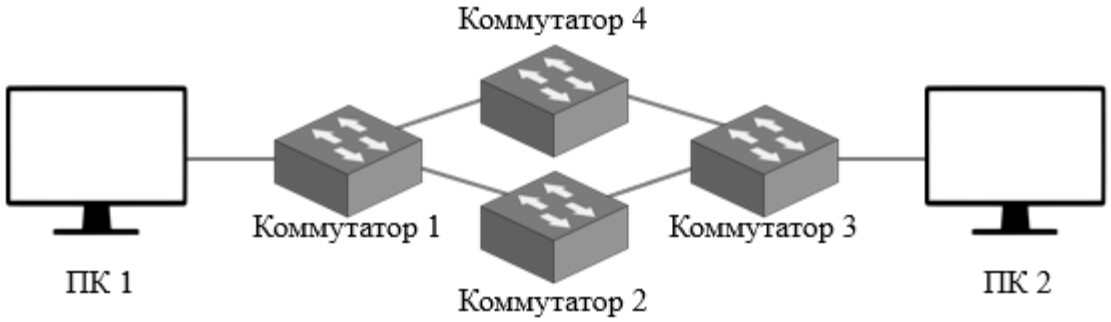

Рис. 1.2. Формирование коммутационной петли в сети с четырьмя коммутаторами

Каждое избыточное соединение множит количество циркулирующих кадров, что быстро исчерпывает пропускную способность сети.

Для того чтобы избежать штормов, сеть должна иметь древовидную топологию с единственным путем между узлами. Однако для отказоустойчивости требуются резервные каналы. Одним из решений является блокировка избыточных соединений, которая может осуществляться двумя способами:

- **ручное управление**, когда в простых сетях администраторы отключают лишние порты;
- **автоматизированный режим**, который применяется в сложных инфраструктурах за счет использования протокола **STP** (Spanning Tree Protocol). Он строит логическое покрывающее дерево, оставляя активными только оптимальные пути, а резервные переводит в пассивный режим. При аварийной ситуации **STP** активирует альтернативные каналы, сохраняя работоспособность сети.

1.1.3. Алгоритм покрывающего дерева (STA) и его роль в сетевой инфраструктуре

Для обеспечения отказоустойчивости и предотвращения петель в сетях с избыточными соединениями используется протокол STP, основанный на **алгоритме STA** (Spanning Tree Algorithm). Этот механизм формирует логическую древовидную топологию, где между любыми узлами существует только один активный путь, а резервные каналы автоматически блокируются. Рассмотрим ключевые понятия, связанные с применением алгоритма STA.

Корневой коммутатор (Root Bridge) – выбранный коммутатор с минимальным **идентификатором**, состоящим из 8 байт, первые 2 байта которого задаются администратором (приоритет), остальные 6 – это MAC-адрес коммутатора. Корневой коммутатор служит точкой отсчета для построения дерева. Все порты корневого устройства остаются активными (назначенными).

Корневой порт (Root Port) – порт на каждом коммутаторе, обеспечивающий кратчайший путь к корневому коммутатору. Расстояние определяется **метрикой** – величиной, обратно пропорциональной пропускной способности канала. Метрика рассчитывается как условное время передачи данных через сегмент. Например, 10 Мбит/с составляет 100 единиц (обновленная шкала), а 1 Гбит/с – 4 единицы.

Назначенный порт (Designated Port) – единственный активный порт в сегменте сети, через который передаются данные к корневому коммутатору. Если несколько устройств претендуют на роль назначенного, выбирается коммутатор с наименьшим расстоянием до корня.

BPDU (**Bridge Protocol Data Unit** – блок данных протокола мостового перенаправления) – служебные пакеты, которыми коммутаторы обмениваются для синхронизации топологии.

Различают следующие **типы BPDU**:

- **конфигурационные** – регулярно рассылаются корневым коммутатором для поддержания структуры дерева;
- **TCN** (**Topology Change Notification** – уведомление об изменении топологии) генерируются при изменениях в сети (например, обрыв связи).

Этапы работы алгоритма STA:

1. **Выбор корневого коммутатора.** Все коммутаторы обмениваются BPDU. Устройство с наименьшим идентификатором (учитывая приоритет и MAC-адрес) становится **корневым**.

2. Определение корневых портов. Каждый коммутатор вычисляет путь к корню, суммируя метрики сегментов. Например, если BPDU от корня приходит через порт с метрикой 4, а следующий коммутатор добавляет метрику своего сегмента (например, 19 для 100 Мбит/с), итоговое расстояние составит 23.

3. Назначение активных портов. Для каждого сегмента выбирается один назначенный порт. Если два коммутатора претендуют на роль назначенного для сегмента, предпочтение отдается устройству с меньшей стоимостью пути до корня.

4. Блокировка избыточных соединений. Все порты, не выбранные в качестве корневых или назначенных, переводятся в состояние **блокировки** (Discarding). Это исключает **коммутационные петли**.

Протокол STP использует три основных таймера для управления процессом построения и поддержания логической древовидной топологии. Измеряемые ими интервалы обеспечивают баланс между скоростью реакции на изменения и устойчивостью сети. Ниже представлены основные таймеры.

1. **Hello Time** – определяет, как часто корневой коммутатор отправляет **BPDU** для синхронизации топологии. Стандартное значение составляет 2 с. Если

коммутатор не получает BPDU в течение **Hello Time · 3** (6 с), то он считает связь с соседом потерянной.

2. Forward Delay – определяет время перехода портов в активное состояние, т. е. задает время, в течение которого порт находится в промежуточных состояниях **Listening** и **Learning** (по 15 с на каждое состояние, итого 30 с) перед переходом в активное состояние **Forwarding**, где:

- **Listening:** порт слушает BPDU, но не передает данные;
- **Learning:** порт изучает MAC-адреса, но все еще не передает трафик;
- **Forward Delay** позволяет предотвратить временные петли, пока все коммутаторы обновляют свои таблицы.

3. Max Age – максимальное время (20 с), в течение которого коммутатор хранит информацию о текущей топологии, полученную через BPDU. Если порт не получает BPDU дольше **Max Age**, коммутатор инициирует пересчет дерева, поскольку увеличение этого значения снижает частоту ложных срабатываний при временных сбоях.

Так, например, **в случае обрыва канала** коммутатор ждет **20 с (Max Age)**, затем тратит **30 с** на активацию резервного пути.

Уменьшение значений таймеров управления процессом поддержания топологии (Hello Time = 1 с, Forward Delay = 10 с, Max Age = 15 с) приведет к ускорению реакции на сбои, но значительно повысит нагрузку на сеть из-за частой отправки BPDU.

Увеличение значений снижает нагрузку, но замедляет восстановление.

При настройке следует учитывать, что все коммутаторы должны использовать **одинаковые настройки таймеров**, иначе возникнут конфликты.

В современных протоколах (RSTP, MSTP) эти интервалы оптимизированы для ускорения конвергенции (до 1–2 с).

Преимущества и недостатки STA

Достоинства:

- автоматическое восстановление сети при авариях за счет активации резервных каналов;
- гарантированная защита от широковещательных штормов.

Недостатки:

- длительное время реконфигурации (до 50 с в крупных сетях);
- устаревшие метрики для высокоскоростных каналов (например, 10 Гбит/с).

Таким образом, STA и STP остаются фундаментом для построения устойчивых сетей с избыточными связями. Несмотря на ограничения в скорости конвергенции, алгоритм обеспечивает баланс между отказоустойчивостью и стабильностью. Для современных сетей часто используются усовершенствованные версии протокола (**RSTP** – Rapid spanning tree protocol – быстрый протокол основного дерева; **MSTP** – Multiple Spanning Tree Protocol – множественный протокол связующего дерева), сокращающие время восстановления.

1.2. Порядок и основные правила выполнения заданий

Для проведения практического занятия используется **симулятор Cisco Packet Tracer**.

1.2.1. Построение и настройка элементов структурной схемы сети

Запустим программу Cisco Packet Tracer. В области **«Логическое пространство»** построим структурную схему сети по типу иерархического дерева с использованием **коммутаторов 2950T-24**, **концентраторов HUB-PT**, **компьютеров PC-PT** и нанесем на нее номера практического занятия и группы, а также фамилию с инициалами.

При обозначении всех устройств необходимо дописать к номеру компьютера последнюю цифру номера группы (например, 2) и номер студента по журналу (например, 9). Структурная схема сети по топологии «иерархическое дерево» представлена на рис. 1.3.

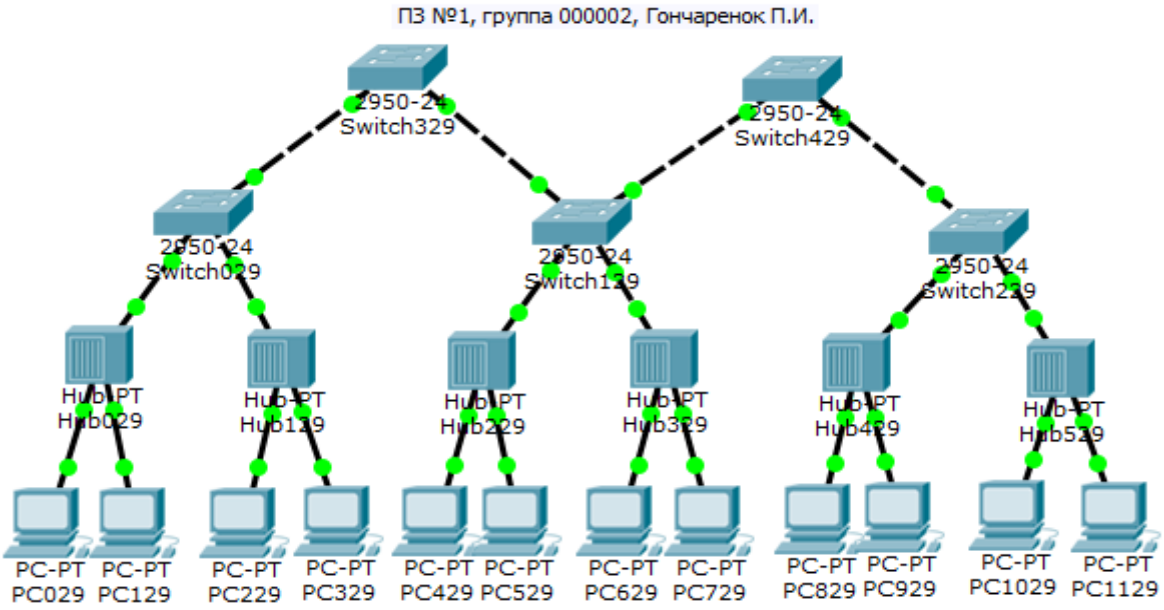

Рис. 1.3. Структурная схема сети по топологии «иерархическое дерево»

Для установления IP-адресов для всех компьютеров будем использовать правило, при котором каждый IP-адрес будет выглядеть следующим образом: 192.100+G.N.C, где G – последняя цифра номера группы; N – порядковый номер студента по журналу; C – порядковый номер компьютера от 1 до 12.

Так, например, если студент обучается в группе № 110902, его номер по журналу 29 и порядковый номер компьютера 2, то его IP-адрес – 192.102.29.2.

За маску подсети во всех случаях будем принимать 255.255.255.0.

Используя это правило, установим IP-адреса для всех компьютеров сети.

Для этого необходимо курсор мыши навести на устройство, например 1, и, нажав ее левую кнопку, в открывшейся вкладке выбрать **Desktop** и далее **IP Configuration**. В соответствующие графы необходимо **ввести IP-адрес и маску подсети** в соответствии с индивидуальными данными каждого компьютера (рис. 1.4).

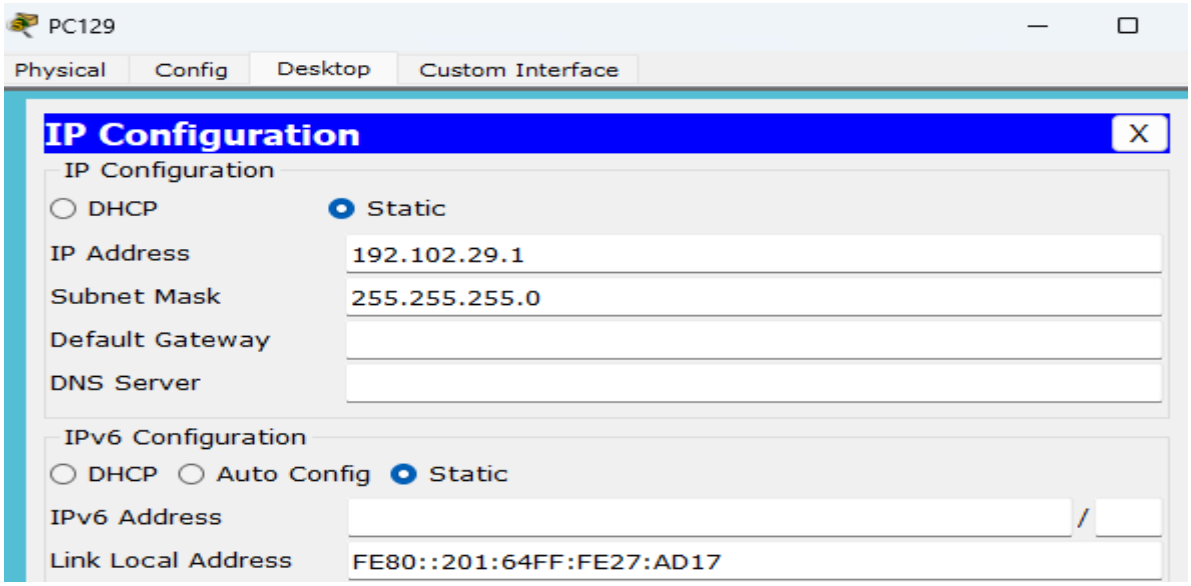

Рис. 1.4. Установление IP-адреса для компьютера 1

После установления IP-адресов всех устройств необходимо проверить правильность настройки сети. Для этого отправим тестовые ICMP-пакеты **Ping** с компьютера PC2 на все другие компьютеры.

С этой целью войдем во вкладку любого компьютера и через кнопку **Desktop** выберем иконку **Command Prompt**. В открывшемся окне наберем команду **Ping** и **IP-адрес компьютера**, соединение с которым нужно проверить, например 192.102.29.11. После чего нажимаем клавишу компьютера **Enter**.

При правильно построенной сети должны появиться сообщения об успешном прохождении утилиты **Ping** до адреса назначения и обратно.

Можно также проверить правильность настройки сети **другим способом**.

С этой целью переключимся в **режим симуляции**. Для этого в нижней правой части рабочего окна активируем вкладку **Simulation**.

В появившемся дополнительном окне **Event List** («Список событий») с помощью вкладки **Edit Filters** («Изменить фильтры») проведем установку только фильтра – **ICMP** (рис. 1.5).

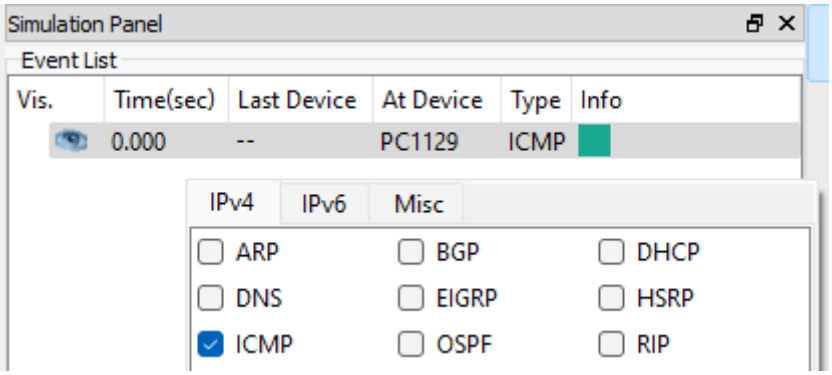

Рис. 1.5. Настройка фильтра контроля сигнала (выбор типа пакета)

После этого закроем дополнительное окно **Event List**.

В правой части окна, в графическом меню с помощью мыши выберем **Add Simple PDU**, установим его (в виде конвертика) на один из компьютеров (например, PC1) и далее укажем узел назначения (например, PC11).

Нажмем вкладку **Auto Capture / Play** («Захват/Вперед») и увидим пошаговое продвижение пакета PDU (рис. 1.6).

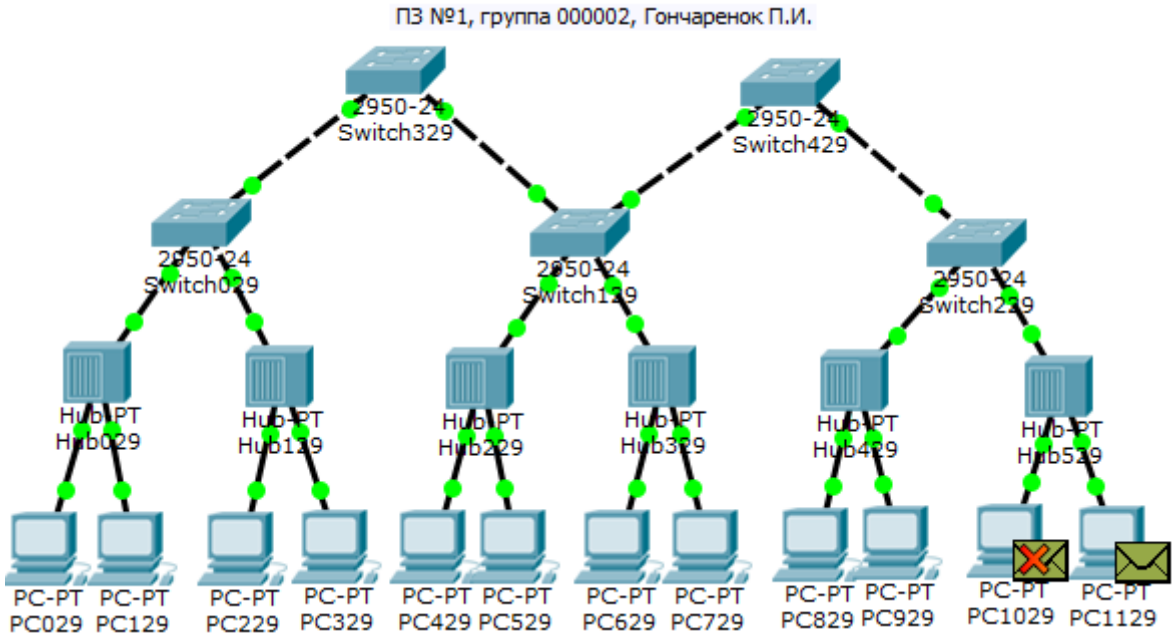

Рис. 1.6. Просмотр событий в режиме симуляции (PDU от PC129 к PC1129)

В это время в окне **Event List** отображаются этапы прохождения тестового пакета. Выбирая тот или иной этап прохождения, можно вернуться и просмотреть параметры движения пакета (рис. 1.7).

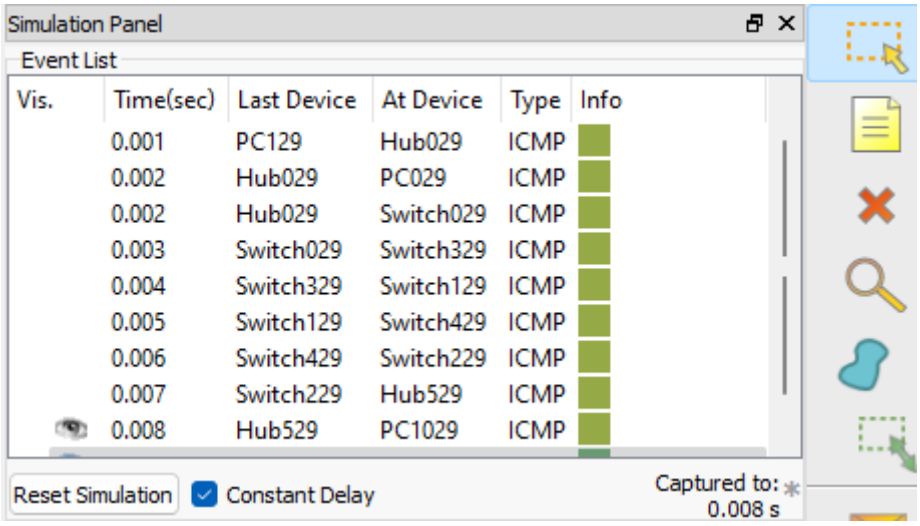

Рис. 1.7. Этапы прохождения пакета от PC129 к PC1129

Просмотрим этапы прохождения пакета через концентраторы, коммутаторы второго и первого уровня с разных компьютеров. Вернемся в **режим Realtime**.

1.2.2. Моделирование работы протокола STP

Соединим оба **коммутатора первого уровня** со всеми коммутаторами второго уровня. Через некоторое время индикаторы на двух портах коммутатора первого уровня (**Switch429**) точки соединения загорятся оранжевым цветом (подчеркнуты линиями на рис. 1.8).

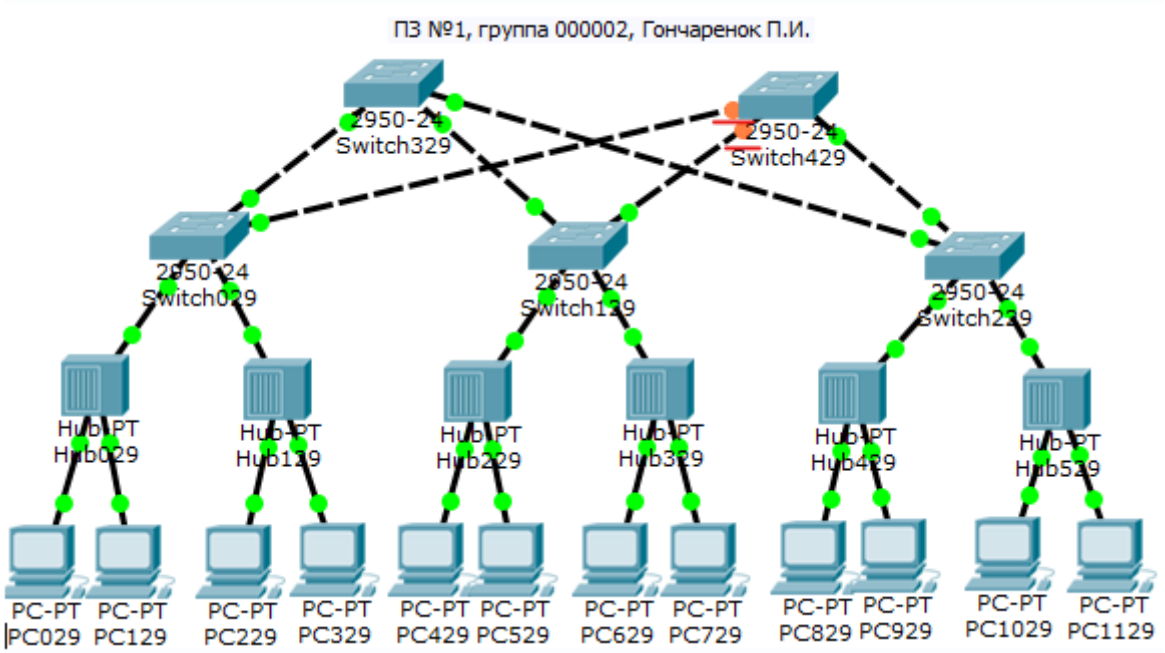

Рис. 1.8. Автоматическое отключение резервных связей на Switch429

Это означает, что пакеты будут проходить по сети, минуя данные порты этого коммутатора.

Протокол STP по умолчанию в управляемых профессиональных коммутаторах инициализируется сразу после включения питания. Поэтому даже при появлении новых избыточных связей наша сеть заработает через 15–30 с.

Благодаря **протоколу STP** коммутатор **Switch329** имеет три активные линии со всеми коммутаторами второго уровня, а коммутатор **Switch429** только с одним. Исходя из этого можно сделать вывод, что коммутатор **Switch329** является **корневым**.

При этом все оставшиеся коммутаторы имеют по одному **корневому порту**, связанному с **корневым коммутатором**.

Проследим для данной схемы за прохождением пакетов в режиме **Simulation**. Выберем **Add Simple PDU компьютера PC129** и направим пакет на **компьютер PC1129**.

Нажав вкладку **Auto Capture / Play**, инициализируем продвижение пакета. Таким образом убедимся в том, что пакет не идет через заблокированные порты коммутатора **Switch429**.

После прохождения пакета **Simple PDU**, нажав вкладку **Auto Capture/ Play**, можно также проследить продвижение пакетов **BPDU протокола STP**.

Посмотрим **содержимое пакета STP BPDU** (розового цвета). Для этого наведем курсор на пакет STP BPDU любого из коммутаторов и щелкнем левой кнопкой мыши. Далее необходимо открыть вкладку **Outbound PDU Details** и прокрутить колесо мыши до конца вниз (рис. 1.9).

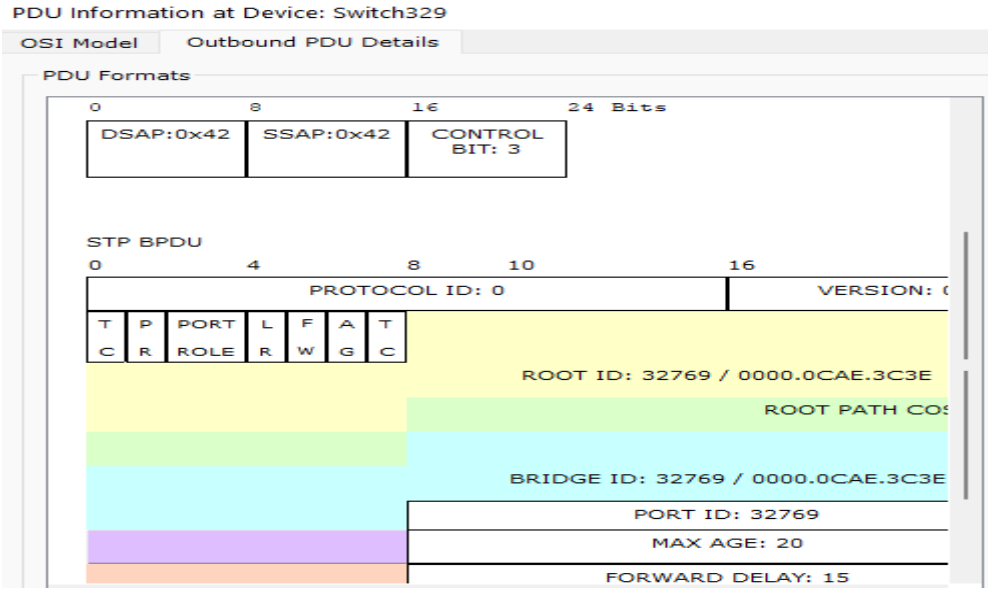

Рис. 1.9. Содержимое пакета STP BPDU

С помощью интерфейса **CLI** (**Command Line Interface** – интерфейс командной строки) корневого коммутатора (Switch329) проведем анализ конфигурации протокола STP. В этом случае будем использовать команду **Show spanning-tree**.

Для решения этой задачи воспользуется **Cisco IOS** (Internetwork Operating System – межсетевая операционная система) **Command Line Interface** коммутатора.

С этой целью путем нажатия левой кнопки мыши откроем таблицу **Switch329**, на которой подключим вкладку **CLI**.

На экране появится **IOS Command Line Interface** (рис. 1.10).

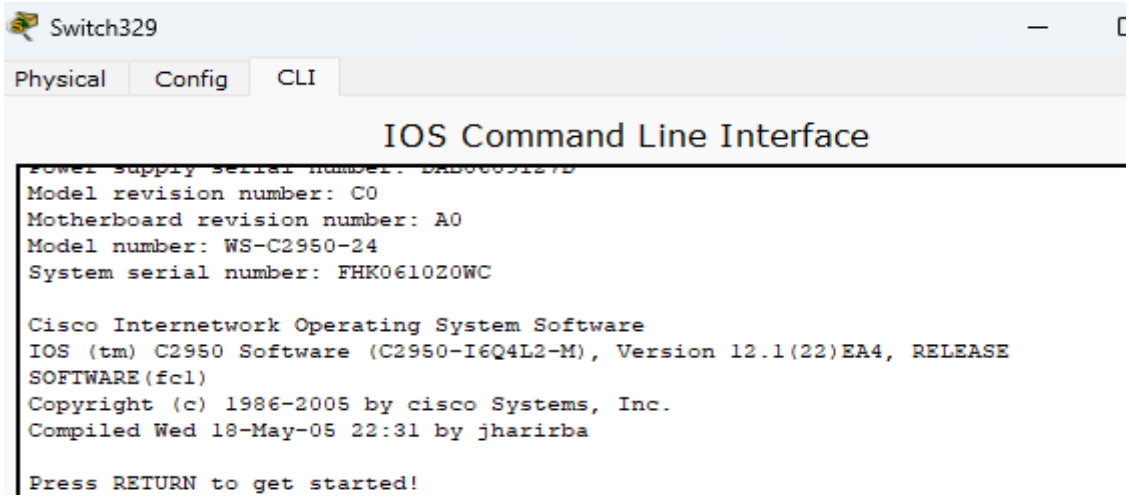

Рис. 1.10. Общий вид вкладки IOS Command Line Interface

Дальнейшее нажатие **Enter** приведет к появлению надписи **Switch>**.

Следует отметить, что **в коммутаторах Cisco для управления используется два режима: пользовательский** (для проверки работы) и **привилегированный** (для управления). Если командная строка начинается со значка «#», мы находимся **в привилегированном режиме**.

Для введения команды **Show spanning-tree** используется пользовательский режим:

Switch>**show spanning-tree**

Полученная информация представлена на рис. 1.11.

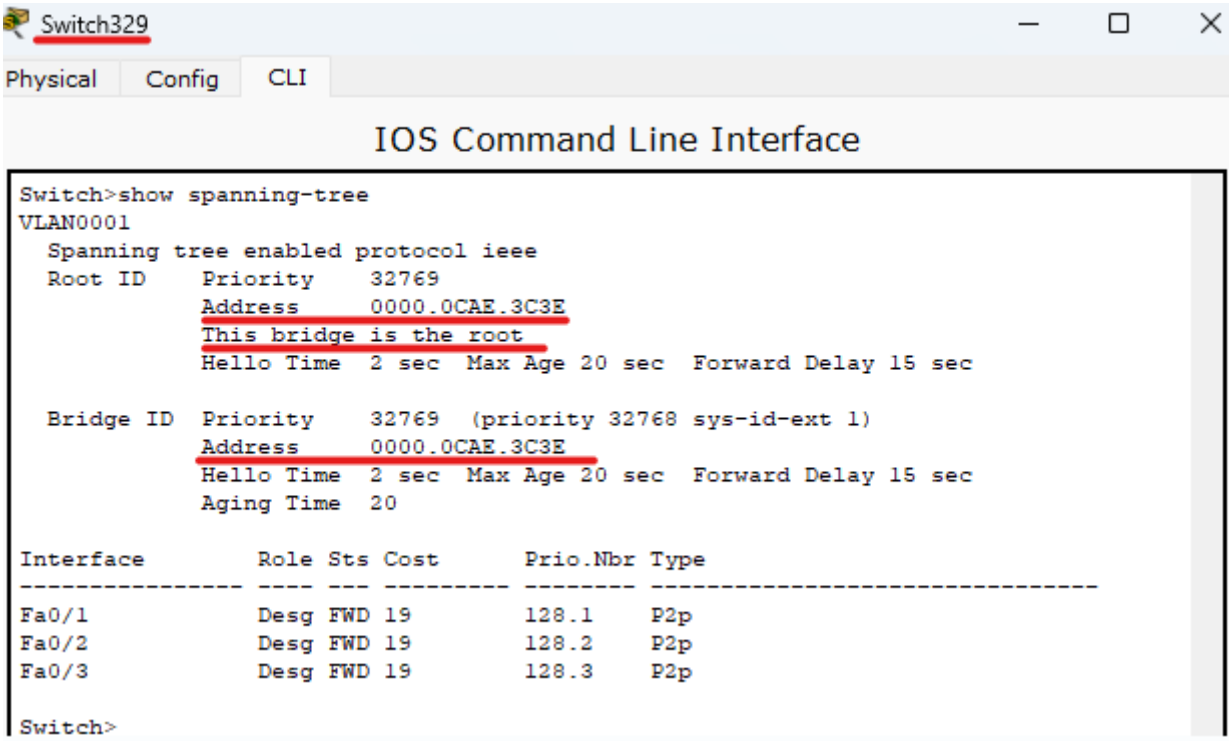

Рис. 1.11. Реализация команды Show spanning-tree для Switch329

На рис. 1.11 имеются сведения о том, что коммутатор Switch329 является корневым и др.

По полученным данным проанализируем степень приоритета каждого порта, его статус: блокированный (**BLK**) или действующий (**FWD**), в том числе назначенных портов каждого коммутатора.

С помощью интерфейса CLI в ручном режиме проведем замену корневого коммутатора.

Этот вопрос следует рассмотреть более подробно.

До того как будет определен корневой коммутатор, каждый из имеющихся коммутаторов претендует на роль корневого и направляет в разделяемую среду **BPDU**, в котором размещен его идентификатор (**ID**) как **ID корневого коммутатора**.

В случае получения **BPDU** другого коммутатора **с меньшим Bridge ID** коммутатор-получатель начинает представлять этот **Bridge ID** как корневой.

Корневым коммутатором становится тот коммутатор, у которого будет самый маленький **Bridge ID**.

Вместе с тем **администратор** может в ручном режиме назначить корневым какой-либо другой коммутатор. В нашем случае определим в качестве корневого коммутатор **Switch429**.

Для этого потребуется выполнить ряд настроек в **привилегированном режиме** данного коммутатора. Для входа в привилегированный режим набираем команду **Enable**:

Switch> **enable**

Switch#

Далее для проведения **глобального конфигурирования** набираем команду **Config terminal**:

Switch#**config terminal**

Switch(config)#

Для определения значения приоритета используется команда **spanning-tree vlan 1 priority ?**

В результате получим следующие сведения:

<0-61440> bridge priority in increments of 4096

Это означает, что допустимые значения ID увеличиваются каждый раз на шаг, равный **4096**.

Следовательно, для назначения коммутатора **Switch429** корневым необходимо уменьшить ID, что выполняется следующей командой:

Switch (config)#**spanning-tree vlan 1 priority 4096**

В результате индикаторы точек соединения двух портов коммутатора **Switch329** загорелись оранжевым цветом (на рис. 1.12 подчеркнуты линиями), что подтверждает передачу роли корневого коммутатора с **Switch329** на **Switch429**.

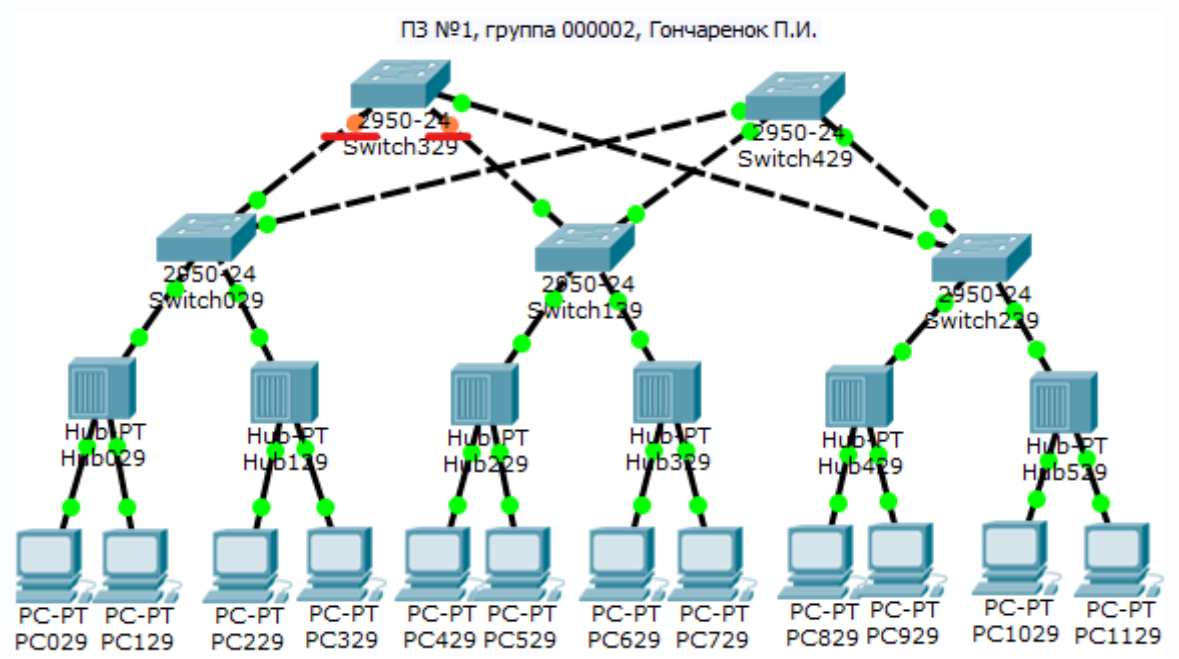

Рис. 1.12. Замена корневого коммутатора с Switch329 на Switch429

При помощи команды **Show spanning-tree** убедимся, что коммутатор Switch429 действительно стал корневым:

Switch>**show spanning-tree**

Полученная информация представлена на рис. 1.13.

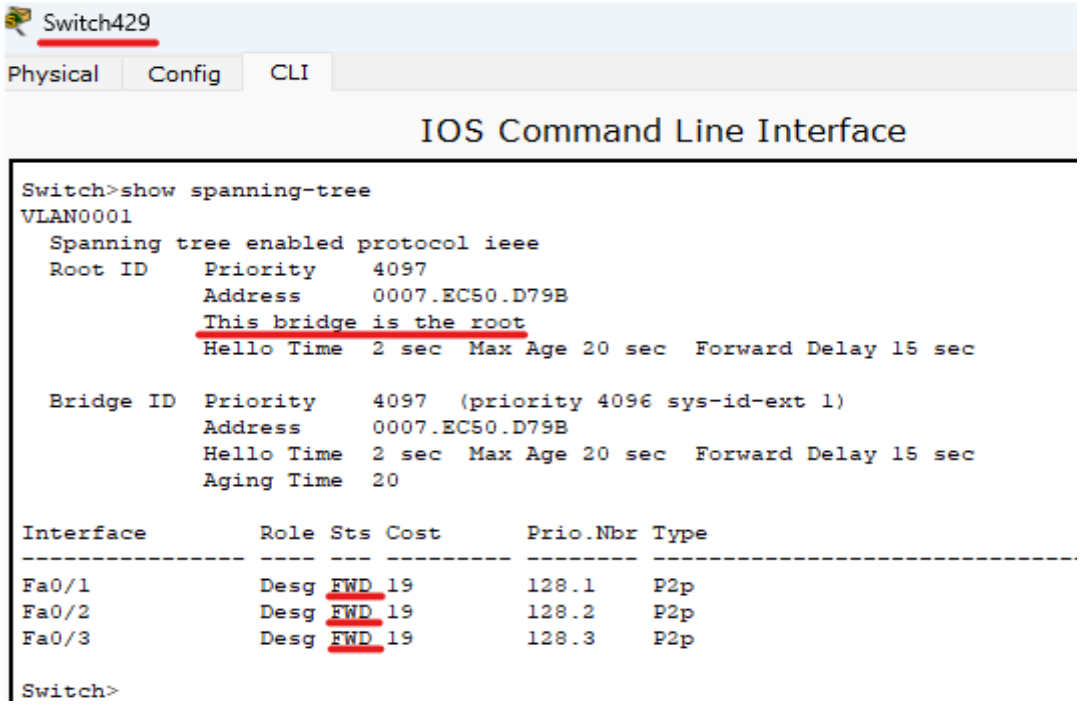

Рис. 1.13. Реализация команды Show spanning-tree для Switch429

При помощи команды **Show spanning-tree** рассмотрим вновь сформированную конфигурацию ранее бывшего корневым коммутатора Switch329, т. е. в новом статусе (рис. 1.14).

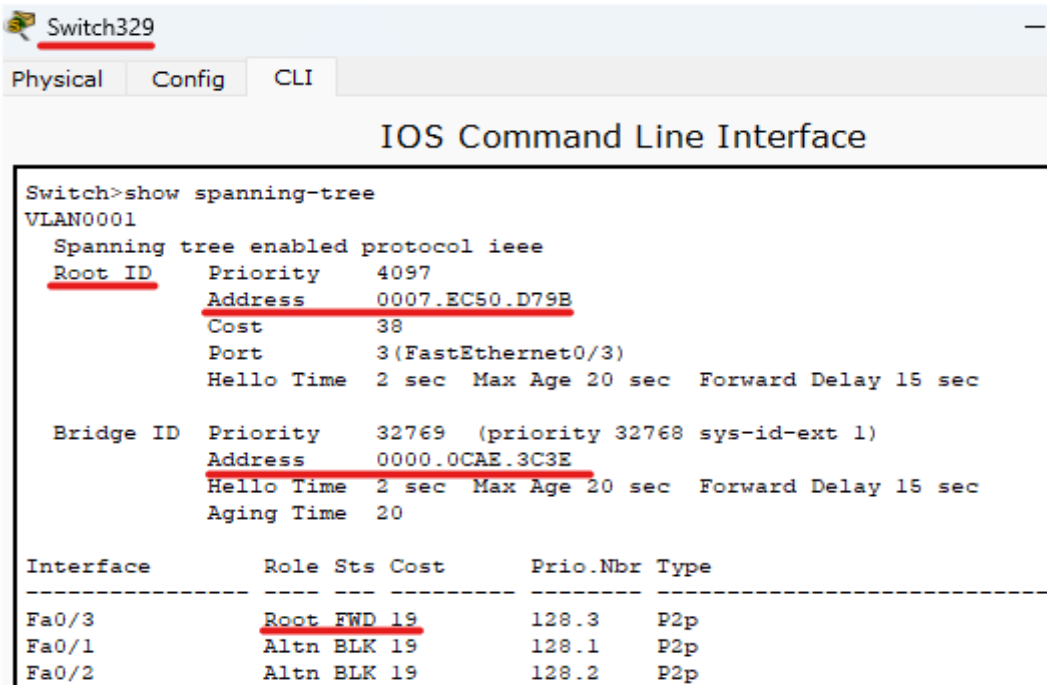

Рис. 1.14. Новый статус коммутатора Switch329

Здесь мы тоже видим, что корневым коммутатором на данный момент является коммутатор с **MAC-адресом 0007.EC50.D79B (Switch429)**.

А коммутатор с **MAC-адресом 0000.0CAE.3C3E (Switch329)** после настройки администратором в ручном режиме, несмотря на более низкий ID, утратил роль корневого коммутатора.

Кроме того, определился его единственный порт входа – **Fa0/3**.

Аналогичным образом при помощи команды **Show spanning-tree** рассмотрим состояние других коммутаторов.

1.2.3. Последствия прекращения функционирования STP

Рассмотрим вариант работы структурной схемы с участием всех коммутаторов при условии отсутствия функционирования STP.

Чтобы обеспечить такие условия, следует отключить STP у всех имеющихся коммутаторов.

С этой целью последовательно выполним настройки, касающиеся коммутатора **Switch329**, представленные на рис. 1.15.

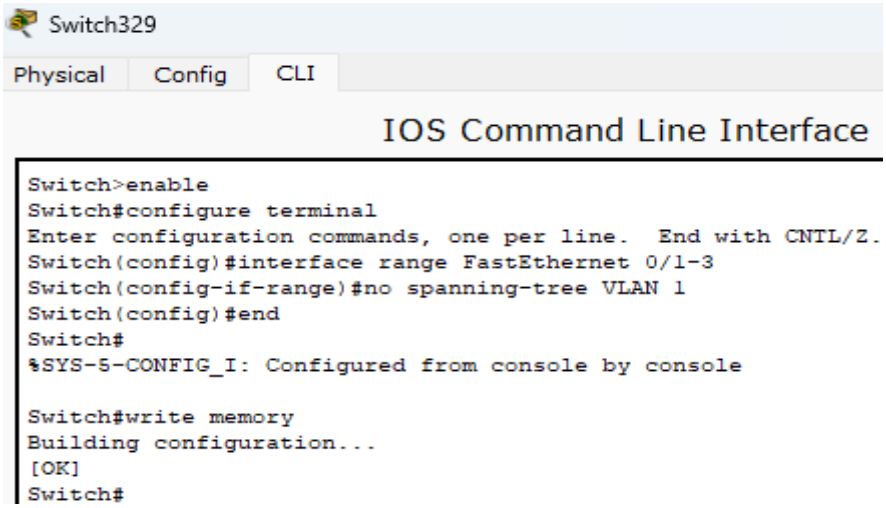

Рис. 1.15. Перечень настроек для отключения STP у коммутаторов

В данном случае запись **FastEthernet 0/1-3** информирует о количестве трех портов, предназначенных для отключения STP.

У коммутаторов **Switch029**, **Switch129** и **Switch229** имеется по четыре порта.

Проведем аналогичные действия **для всех коммутаторов**.

После отключения протокола STP на всех коммутаторах при попытке направить пакет от компьютера к компьютеру мы увидим мигание зеленой подсветки по всей сети.

Данное явление называется **широковещательным штормом**. В нашем случае возникло как результат отключения протокола STP и привело к увеличению широковещательных сообщений и в конечном итоге к парализации работы всей сети.

При этом на вставках **Cisco Packet Tracer** видно, что в режиме **Simulation** при попытке послать пакет любым компьютером появляется отказ в работе

коммутаторов, что подтверждается появившимися обозначениями (рис. 1.16) и окном с надписью (рис. 1.17).

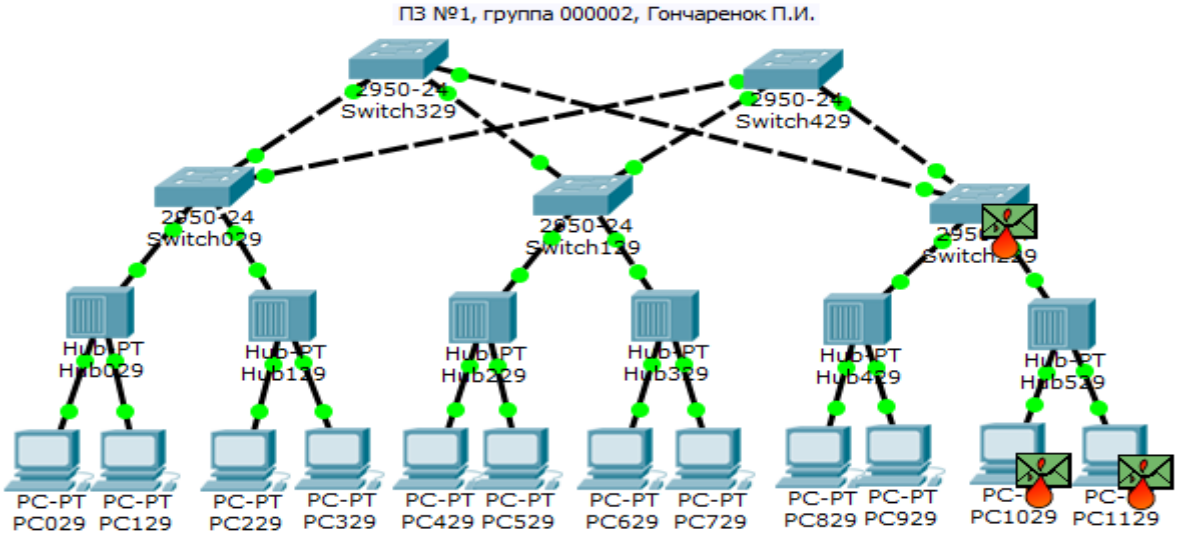

Рис. 1.16. Компьютерная сеть в состоянии широковещательного шторма

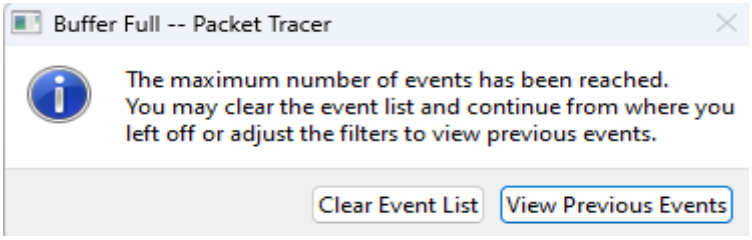

Рис. 1.17. Окно с надписью о переполнении буфера

При этом физическая топология при отключенном STP сохраняется.

1.2.4. Устранение условий возникновения широковещательного шторма

Удалим один коммутатор первого уровня (рис. 1.18).

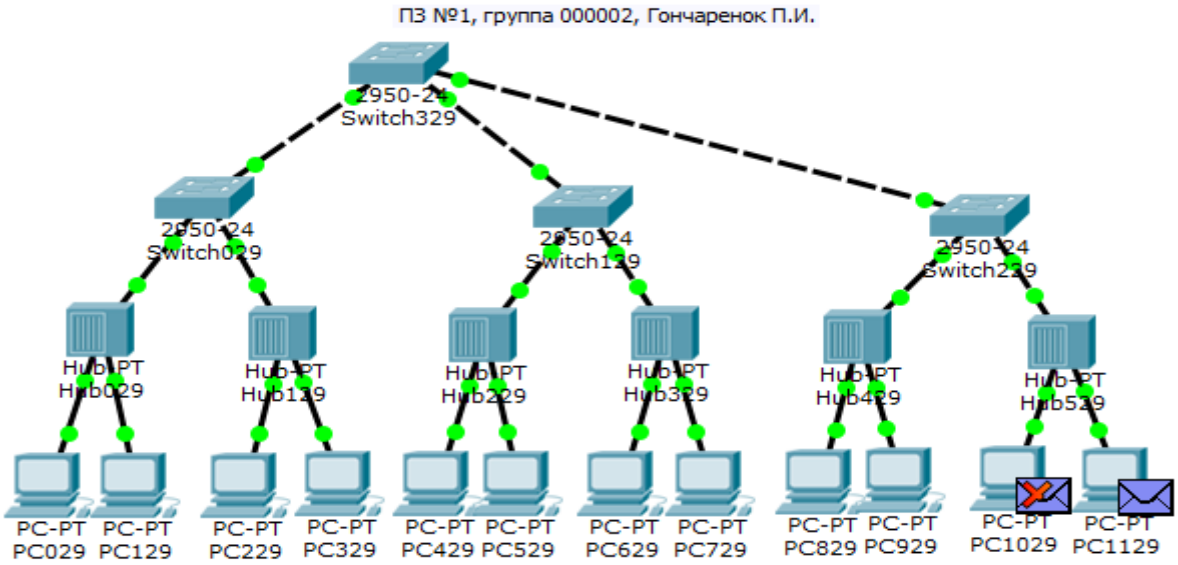

Рис. 1.18. Восстановление работоспособности сети

Из рис. 1.18 видно, что работоспособность сети восстановлена. Это связано с отсутствием избыточных связей, а следовательно, отсутствием причин для появления петель коммутации.

1.3. Практическое задание

1.3.1. Изучить сведения, изложенные в теоретических материалах данного практического занятия и, при необходимости, в дополнительных материалах.

1.3.2. С целью практического изучения работы алгоритма покрывающего дерева STA необходимо выполнить следующие действия.

1.3.2.1. Построить структурную схему локальной сети, состоящую из коммутаторов, концентраторов и компьютеров. При этом наименования компонентов сети, их IP-адреса обозначить согласно требованиям, изложенным в пп. 1.2.1–1.2.2.

1.3.2.2. Проверить правильность настройки сети, отправив тестовый ICMP-пакет Ping.

1.3.2.3. Смоделировать работу протокола STP.

1.3.2.4. Осуществить просмотр параметров и событий, а также управление коммутатором через консольный интерфейс коммутатора.

1.3.2.5. Проанализировать и выполнить общую конфигурацию коммутаторов с помощью интерфейса CLI.

1.3.2.6. Изменить корневой коммутатор с помощью интерфейса CLI.

1.3.2.7. Провести остановку работы протокола STP и ознакомиться с явлением широковещательного шторма.

1.3.2.8. Устранить условия возникновения широковещательного шторма.

1.3.3. Отчет о проделанной работе представить преподавателю непосредственно на компьютере или в виде сканирования ключевых эпизодов выполнения работы по п. 1.3.2.

1.3.4. Ответить на контрольные и дополнительные вопросы.

Контрольные вопросы

1. Назовите особенности разделяемой среды.
2. Для чего полезно проводить логическую сегментацию?
3. Что позволяет получить команда Show mac-address-table?
4. В чем заключается суть явления широковещательного шторма?
5. Чем характеризуется протокол STP и в чем заключается целесообразность его применения?
6. В чем заключается особенность микросегментации применительно к коммутатору?
7. На основании чего коммутатор строит адресную таблицу?
8. Как взаимосвязаны адресная таблица коммутатора и режим обучения?
9. Какой простой способ избегания широковещательного шторма?
10. Каким образом определяются корневые коммутаторы?
11. В чем заключаются особенности идентификатора коммутатора?

12. На основании какого стандарта работает алгоритм прозрачного моста?
13. С чем связана необходимость применения BPDU?
14. Чем отличается таблица фильтрации коммутатора от таблицы адресации?
15. В чем заключается потенциальная опасность петлеобразных маршрутов сети?
16. В чем заключается отличие статических и динамических адресных таблиц коммутации?
17. Какие таймеры используются для управления процессом построения и поддержки логической древовидной топологии?
18. Что представляет собой таймер Forward Delay?
19. Что представляет собой таймер Hello Time?
20. Какие имеются типы записей в таблице коммутации?

Практическое занятие № 2
«Методика расчета количества хостов и подсетей на основе IP-адреса и маски»

Цель занятия: углубить знания по работе протокола IPv4, структуре IP-адреса, классам IP-адресов и отработать навыки применения методики расчета количества хостов и подсетей на основе IP-адреса и маски.

2.1. Краткие теоретические сведения

2.1.1. Межсетевой обмен в сетях TCP/IP. Протокол IP

В настоящее время **стек протоколов TCP/IP** служит фундаментом для организации связи в современных сетях, включая Интернет. Эта модель объединяет набор протоколов, которые регулируют передачу данных между устройствами, обеспечивая их совместимость и эффективность.

Рассмотрим ее ключевые компоненты и принципы работы.

Модель TCP/IP разделена на четыре уровня, каждый из которых решает специфические задачи:

- **прикладной уровень (Application Layer)** управляет взаимодействием пользовательских приложений (браузеров, почтовых клиентов) с сетью (**HTTP** – HyperText Transfer Protocol – протокол передачи гипертекста; **FTP** – File Transfer Protocol – протокол передачи файлов по сети; **SMTP** – Simple Mail Transfer Protocol – простой протокол передачи почты);
- **транспортный уровень (Transport Layer)** обеспечивает передачу данных между устройствами с контролем качества (TCP и UDP);
- **сетевой уровень (Internet Layer)** отвечает за маршрутизацию и доставку пакетов через сеть (IP);
- **канальный уровень (Link Layer)** регулирует передачу данных в пределах локальной сети (Ethernet, Wi-Fi).

TCP (Transmission Control Protocol) гарантирует **надежную доставку данных**. Перед передачей данный протокол устанавливает соединение (рукопожатие), разбивает информацию на сегменты и проверяет их целостность. При потере пакета TCP повторяет отправку, что крайне важно для веб-страниц, электронной почты, файловых загрузок.

IP (Internet Protocol) определяет **логическую адресацию устройств** (IPv4, IPv6) и маршрутизацию пакетов. Он разбивает данные на пакеты (дейтаграммы), которые независимо перемещаются по сети к получателю.

Вместе с тем данный протокол не гарантирует доставку. Эту задачу берут на себя протоколы транспортного уровня (например, TCP).

В этом случае TCP готовит данные к передаче, а IP доставляет их до адресата. Например, при загрузке страницы браузер использует TCP для корректного получения HTML-файла, а IP направляет пакеты через маршрутизаторы к серверу.

Каждое устройство в сети имеет уникальный **IP-адрес**, который позволяет идентифицировать его и находить оптимальный маршрут для передачи данных.

В настоящее применяются две версии **IP**:

- **IPv4 –** 32-битный адрес в формате 192.168.100.1. Из-за ограниченного количества адресов IPv4 постепенно заменяется на IPv6;
- **IPv6 –** 128-битный адрес, например 2001:0db8:85a3::8a2e:0370:7334, решающий проблему нехватки IPv4 и способствующий повышению эффективности функционирования сети.

Широко востребованным протоколом транспортного уровня является **UDP (User Datagram Protocol)**, который обеспечивает **быструю, но ненадежную доставку** без установки соединения, а пакеты могут теряться или приходить в неправильном порядке, но минимизация задержек компенсирует этот недостаток. **UDP** используется там, где скорость важнее точности, например в стриминге видео (Zoom, YouTube), онлайн-играх, VoIP-звонках.

Стек протоколов TCP/IP универсален и поддерживает:

- **интернет вещей (IoT).** Умные устройства обмениваются данными через IP;
- **облачные сервисы.** Виртуальные машины и хранилища используют TCP для синхронизации;
- **мобильную связь.** 4G/5G-сети интегрированы с IP для доступа в Интернет.

Можно сказать TCP/IP – это «кровеносная система» цифрового мира, обеспечивающая связь между миллиардами устройств. Благодаря гибкости и многоуровневой архитектуре стек адаптируется к новым технологиям, сохраняя актуальность в эпоху облачных вычислений и глобальной цифровизации.

2.1.2. Протокол IPv4

Существует несколько версий протокола IP. Номера версий протокола IP приведены в RFC 1700.

В настоящее время **широко используется версия IPv4 (RFC 791)**.

Протокол IPv6, за которым будущее, применяется в высокоскоростных сетях.

IPv4 представляет собой дейтаграмму. Он содержит заголовок и полезную нагрузку.

Протокол IP разбивает большое количество информации **на пакеты**. В этой связи рассмотрим пакет с заголовком **протокола IPv4**.

Данные заголовка позволяют определить сетевой интерфейс получателя (IP-адрес получателя) пакета и направить пакет либо на сетевой интерфейс данной сети, либо на соответствующий шлюз.

При этом если пакет слишком долго «гуляет» по сети, то очередной шлюз может уничтожить этот IP-пакет и отправить на машину-отправителя уведомление (ICMP-пакет, Internet Control Message Protocol – протокол межсетевых управляющих сообщений) о том, что надо использовать другой шлюз.

На этом принципе работает **программа Ping**, которая используется для определения маршрутов прохождения пакетов по сети.

При обычной процедуре инкапсуляции IP-пакет помещается в поле данных кадра (фрейма) протокола канального уровня.

Если же это невозможно, то пакет разбивается на более мелкие фрагменты. Для восстановления исходного IP-пакета его «нарезанные» фрагменты должны содержать информацию об их местоположении в исходном IP-пакете. Для этой цели используются поля «флаги» (flags) и «смещение фрагмента» (fragmentation offset).

В этих полях определяется, какая часть пакета получена в данном фрейме. Размер максимально возможного фрейма, который передается по сети, определяется величиной **MTU** (Maximum Transsion Unit – максимальная единица передачи).

Давайте посмотрим на **две большие части**, из которых состоит **IP-пакет** (рис. 2.1).

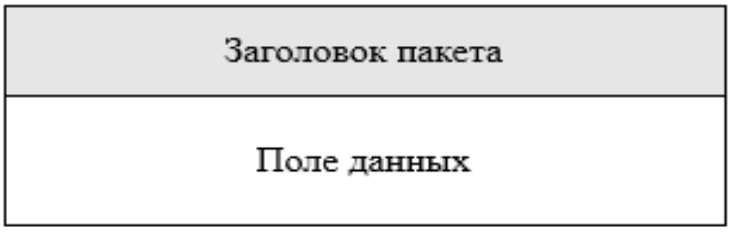

Рис. 2.1. Общий вид IP-пакета

Заголовок пакета **IPv4** содержит 14 полей, из которых **13 являются обязательными**, при этом четырнадцатое поле предназначено для необязательных опций. Среди них важную роль играет функция ECN (Explicit Congestion Notification, явное уведомление о перегрузке). Структура заголовка представлена на рис. 2.2.

Рис. 2.2. Общий вид заголовка пакета IPv4

Протокол **IPv4** поддерживает **три режима адресации**.

1. Одноадресный. При использовании этого режима **данные** передаются только на один сетевой узел, причем каждый из них может являться как отправителем, так и получателем. Поле адреса назначения содержит 32-битный IP-адрес устройства-получателя. Одноадресный режим используется чаще всего при обращении к интернет-протоколу.

2. Широковещательный. При его использовании все устройства, подключенные к сети с множественным доступом, имеют возможность получения и обработки дейтаграмм, передаваемых **по стеку протоколов TCP/IPv4**. Для этого поле **IP-адреса** назначения включает в себя **специальный широковещательный код идентификации**.

3. Многоадресный. Согласно правилам обработки данных **по протоколу IPv4** сюда входят адреса **в диапазоне от 224.0.0.0 до 239.255.255.255**. Режим объединяет два предыдущих, определяется наиболее значимой моделью 1110.

В этом пакете адрес назначения содержит специальный код, который начинается с 224.x.x.x и может использоваться более чем одним узлом.

2.1.3. Принципы построения IP-адресов

Компьютерам, серверам и роутерам в Интернете нужно понимать, куда отправлять данные, чтобы они не потерялись в сети на пути от сервера к хосту и наоборот.

Как мы уже выяснили, один из помощников в этом деле – IP-адрес. Он представляет собой что-то вроде дорожного указателя, маяка, который содержит данные о месте конкретного устройства в структуре глобальной сети.

Чтобы узнать **IP-адрес устройства**, можно открыть **Windows PowerShell** и ввести команду **Ipconfig**.

IPv4 определяет IP-адрес, который представляет собой серию из 32 двоичных бит (единиц и нулей).

Так как человек невосприимчив к большому однородному ряду чисел, такому как **11100010101000100010101110011110** (здесь 32 бита информации, так как 32 числа в **IPv4** в двоичной системе), было решено разделить ряд на четыре 8-битных байта и получилась следующая последовательность: **11100010.10100010.00101011.10011110**.

Это не сильно изменило ситуацию и **было принято решение перевести данную последовательность в привычную нам последовательность из четырех чисел в десятичной системе**, т. е. в нашем случае **11100010.10100010.00101011.10011110 будет представлено в виде 226.162.43.158**.

Четыре разряда **по 8 бит** называются **октетами**. По такой схеме адресации (**IPv4**) можно создать **4 294 967 296 (2^{32}) IP-адресов**.

Максимальным возможным числом **в любом октете будет 255** (так как в двоичной системе это **восемь единиц**), а минимальным – **0** (соответственно, **восемь нулей**).

IP-адрес состоит из двух частей: **адреса сети** и **номера хоста** (обычно под хостом понимают компьютер, подключенный к сети).

В настоящее время понятие **«хост»** имеет более широкое толкование. Это может быть не только компьютер с сетевой картой, но и любое устройство, которое имеет свой сетевой интерфейс (например: принтер, робот, холодильник и т. д.).

Свой уникальный IP-адрес есть у каждого сетевого устройства, при этом в глобальной сети не может существовать два устройства с одинаковым IP.

IP-адреса делятся на **пять классов (A, B, C, D, E)**. **A**, **B** и **C** – это классы коммерческой адресации, **D** – для многоадресных рассылок, а класс **E** – для экспериментов (табл. 2.1).

Таблица 2.1

Классы IP-адресов (A, B, C, D, E) и их назначение

Класс	Первые биты	Наименьший номер сети	Наибольший номер сети	Максимальное число узлов в сети
A	0	1.0.0.0 (0 – не используется)	126.0.0.0 (127 – зарезервирован)	2^{24}, поле 3 байта
B	10	128.0.0.0	191.255.0.0	2^{16}, поле 2 байта
C	110	192.0.0.0	223.255.255.0	2^{8}, поле 1 байт
D	1110	224.0.0.0	239.255.255.255	Групповые адреса
E	11110	240.0.0.0	247.255.255.255	Зарезервировано

Исходя из приведенной структуры адресов и информации из таблицы, можно сделать несколько выводов.

1. Класс A среди классов включает минимальное количество сетей, что обеспечивает уникальность для крупных организаций. Максимально возможное число устройств в одной сети достигает **16 777 214** ($2^{24} – 2$), что делает этот класс оптимальным для глобальных инфраструктур (например, корпоративных сетей провайдеров). Первый октет (байт) определяет сеть, остальные три – идентификаторы узлов.

2. Класс B включает значительно больше сетей, чем в классе A, – до **65 534** устройств ($2^{16} – 2$) на одну сеть, что подходит для средних предприятий. Первые два октета выделены под номер сети, последние два – под хосты.

3. Класс C – наиболее многочисленный класс, предназначенный для небольших локальных сетей, ограничен **254** устройствами ($2^{8} – 2$) на сеть. Три первых октета определяют сеть, последний – хост.

4. Класс D (групповая рассылка) предназначен не для идентификации отдельных устройств, а для **многоадресной передачи** данных группе интерфейсов, которые могут находиться в разных сетях. Адреса начинаются с битовой последовательности **1110** в первом октете (диапазон: 224.0.0.0–239.255.255.255).

5. Класс E (экспериментальный) зарезервирован для будущих технологий и исследовательских задач. Начинается с последовательности **11110** (диапазон: 240.0.0.0–255.255.255.255). Не используется в публичных сетях.

Любой **IPv4-адрес** содержит две логические части:

- **номер сети** определяет, к какой сети принадлежит устройство;
- **номер хоста** уникально идентифицирует устройство внутри сети.

Например, адрес 192.168.1.34 (класс C) можно разделить на сеть 192.168.1 и хост 34 (рис. 2.3).

Рис. 2.3. Разделение IP-адреса на номера сети и хоста

Маска подсети (255.255.255.0 для класса C) помогает определить границы между этими частями.

При отправке пакета **маршрутизатор** анализирует **номер сети** получателя. Если сеть локальная, пакет доставляется напрямую. Если удаленная – передается через шлюз.

В IP-сетях методы передачи данных делятся на два основных типа: **unicast** (индивидуальная передача) и **multicast** (групповая передача). Они различаются не только форматом адресов, но и принципами маршрутизации, сценариями применения и влиянием на сетевую инфраструктуру.

Unicast остается основой для большинства интернет-взаимодействий, обеспечивая надежную двустороннюю связь. **Multicast**, в свою очередь, решает задачи массовой рассылки, минимизируя нагрузку на сеть. Выбор метода зависит от требований к масштабу, надежности и эффективности использования ресурсов. Для работы multicast необходима инфраструктура, поддерживающая групповую маршрутизацию, в то время как unicast реализуется в любой IP-сети.

Система IP-адресации представляет собой строго регулируемую структуру, которая обеспечивает уникальность и корректную маршрутизацию данных в глобальной сети. Ее функционирование основано на международных стандартах, иерархии провайдеров и технологических инновациях, преодолевающих ограничения устаревших протоколов.

IP-адресация – это динамическая система, балансирующая между наследием IPv4 и инновациями IPv6.

Корпорация ICANN (Internet Corporation for Assigned Names and Numbers – корпорация по управлению доменными именами и IP-адресами) через **подразделение IANA** (Internet Assigned Numbers Authority – администрация адресного пространства Интернет) контролирует распределение ключевых интернет-ресурсов. Роль ICANN и провайдеров критична для предотвращения коллизий и обеспечения стабильности сети.

Следует также подчеркнуть, что все устройства, например, для класса С, идентификаторы которых начинаются **с 192.168.1**, **находятся в одной сети** (рис. 2.4).

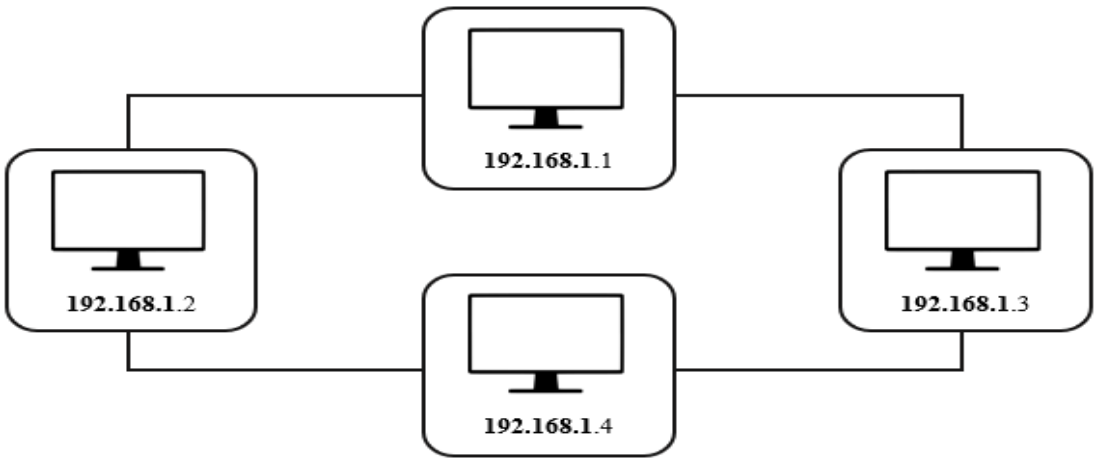

Рис. 2.4. Общий вид расположения нескольких устройств в одной сети

Устройство, идентификатор которого начинается, например, **с 192.168.2**, **будет принадлежать к другой сети** и не сможет связываться с устройствами из сети 192.168.1.

Чтобы это сделать, **понадобится роутер**, который соединит две сети между собой. Он **будет мостом**, по которому данные переходят из одной сети в другую (рис. 2.5).

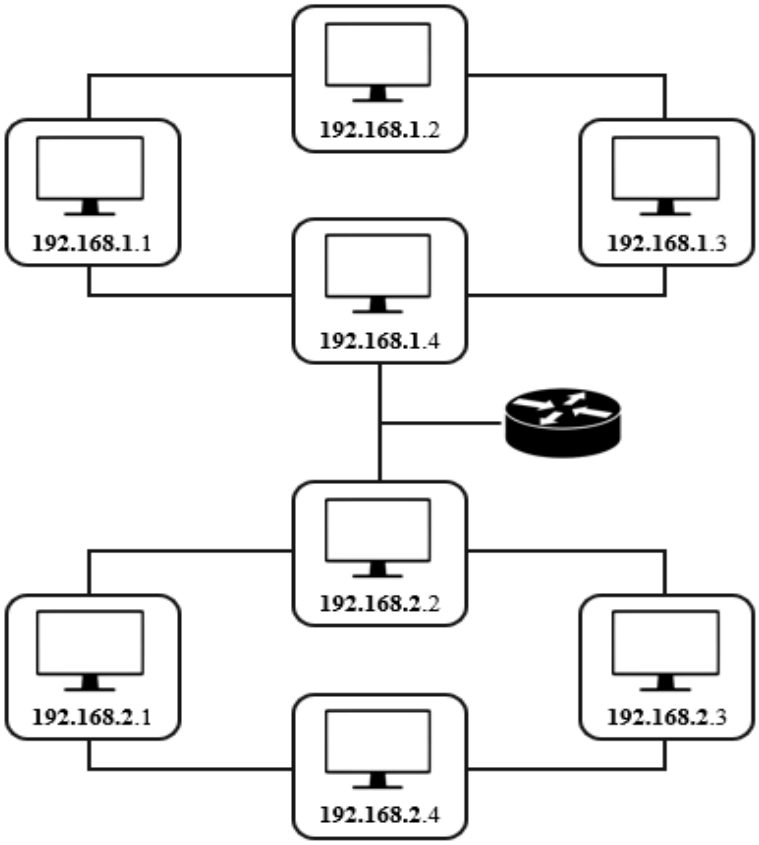

Рис. 2.5. Вариант соединения устройств двух сетей

IP-адреса (как IPv4, так и IPv6) выделяются региональным интернет-регистраторам (RIR), которые далее распределяют их между провайдерами.

Доменные имена регистрируются через аккредитованных регистраторов. Каждая регистрация домена сопровождается выплатой комиссии ICANN, что поддерживает инфраструктуру DNS (Domain Name System – система доменных имен).

Частные сети используют зарезервированные IANA диапазоны, которые не маршрутизируются в публичном Интернете:

- **10.0.0.0/8** – для крупных корпоративных сетей;
- **172.16.0.0/12** – гибкий диапазон для средних организаций;
- **192.168.0.0/16** – стандарт для домашних роутеров и малых офисов.

Эти адреса становятся «публичными» через **NAT** (Network Address Translation – преобразование сетевых адресов), преобразующий «серые» IP в уникальный внешний адрес.

Бывает, **что хостов в сети больше, чем доступных IP-адресов**, – в современном Интернете дела обстоят именно так.

В этом случае **интернет-провайдеры выдают устройствам адреса формата IPv6**. При этом адрес IPv4 можно легко переделать в формат IPv6, а вот IPv6 в IPv4 нельзя.

Однако не все интернет-провайдеры перешли на новую версию IP-адресов, и это **создало новую проблему**: невозможно напрямую отправлять данные с устройств, поддерживающих IPv4, на устройства с IPv6.

Проблему решили с помощью туннелирования – создали специальный канал между двумя устройствами.

Технологии туннелирования и NAT стали временными «костылями», тогда как массовый переход на IPv6 остается единственным устойчивым решением для растущего интернета вещей и 5G-сетей. Пользователи, в свою очередь, сталкиваются с «невидимым» слоем инфраструктуры, где IP-адреса динамично меняются, сохраняя бесперебойность связи.

2.1.4. Маска подсети: структура, назначение и практическое применение

Маска подсети – это ключевой инструмент в IP-сетях, позволяющий гибко управлять разделением адресного пространства на логические сегменты. В отличие от устаревшей классовой системы она предоставляет администраторам возможность точно настраивать размер сети и оптимизировать использование IP-адресов.

Маска состоит из 32 бит (для IPv4) или 128 бит (для IPv6) и записывается в том же формате, что и IP-адрес. Ее биты делятся на две части:

- **сетевой префикс –** последовательность единиц, определяющая принадлежность к конкретной сети;
- **идентификатор хоста –** нулевые биты, выделенные для устройств внутри этой сети.

Для IPv4 в двоичном коде такая маска выглядит как 1111 1111 1111 0000. Нули показывают, где находится номер хоста, а единицы – номер сети.

Чтобы **применить маску**, нужно **воспользоваться логическими операторами «И» и «НЕ»**. **Первый** работает по следующим правилам:

1 И 1 = 1
1 И 0 = 0
0 И 1 = 0
0 И 0 = 0

Оператор «НЕ» просто меняет все нули на единицы, а единицы на нули. И делает он это **справа налево**:

НЕ 1 = 0
НЕ 0 = 1

Рассмотрим, как маска подсети 255.255.255.0 определяет структуру сети для IP-адреса 192.168.1.34. Это пример классической настройки для домашних или малых офисных сетей, где требуется простота управления и достаточный размер подсети:

11000000 10101000 00000001 00100010
и
11111111 11111111 11111111 00000000
=
11000000 10101000 00000001 00000000
(или 192.168.1.0 в десятичной системе счисления)

Здесь видно, как мы сначала **перевели IP-адрес и маску подсети в двоичную систему счисления**. А **затем побитово** справа налево применили **операцию логического «И»**.

Маска помогла удалить ненужную часть адреса, и мы **выделили номер сети – 192.168.1.0**. Коротко в десятичном виде эта запись выглядит так: **192.168.1.0/24**.

Чтобы **выделить номер хоста**, нужно сначала применить **операцию логического «НЕ» к маске подсети**:

НЕ 11111111 11111111 11111111 00000000
=
00000000 00000000 00000000 11111111

Затем **операцию логического «И» к IP-адресу и полученной маске**.

Так мы **получили маску для выделения номера устройства**.

11000000 10101000 00000001 00100010
и
00000000 00000000 00000000 11111111
=
00000000 00000000 00000000 00100010

У нас получился адрес 0.0.0.34. Это и есть номер хоста.

2.1.5. Бесклассовая адресация CIDR

До **CIDR** (Classless Inter-Domain Routing – бесклассовая адресация) Интернет опирался на классы (A, B, C), где размер сети жестко определялся первыми битами адреса. Это приводило к **расточительству адресов**, когда малым организациям выделяли класс C (254 хоста), а крупным – класс B (около 65 тыс. хостов), что часто не соответствовало реальным потребностям.

Кроме того, каждая сеть класса C добавляла отдельную запись в таблицы маршрутизаторов, что приводило к **росту таблиц маршрутизации**, вызывая их переполнение (к 1993 году – более 10 тыс. записей). При этом мы учитываем, что темпы роста таблиц маршрутизации являются важным показателем для анализа состояния и развития Интернета, особенно в контексте масштабируемости и производительности сетевого оборудования.

CIDR устранила эти проблемы, введя **гибкое распределение адресного пространства** и **агрегацию маршрутов**.

Принципы работы CIDR включают в себя следующие основные аспекты:

• **префикс произвольной длины.** Адресное пространство делится на блоки с помощью масок переменной длины (от /8 до /32 в IPv4). Например, блок 172.16.0.0/12 охватывает 1 048 576 адресов (от 172.16.0.0 до 172.31.255.255), что эквивалентно 16 классам B;

• **агрегация маршрутов.** Провайдеры объединяют подсети клиентов в суперсети. Например, блок 203.0.113.0/24 может быть частью более крупного 203.0.112.0/23, сокращая число записей в глобальных таблицах;

• **иерархическая структура.** Региональные интернет-регистраторы (RIR) выделяют провайдерам крупные блоки (например, /18), которые те дробят на /24…/30 для клиентов.

В целом CIDR позволяет контролировать адресацию непрерывных блоков IP-адресов. Это намного удобнее, чем подсеть.

Применение CIDR позволило предотвратить экспоненциальный рост маршрутных таблиц Internet. К 2023 году размер полной таблицы BGP (Border Gateway Protocol – протокол граничного шлюза) – около 900 тысяч записей. Без CIDR это число превысило бы 2 млн.

Для сетевых инженеров понимание CIDR – не просто навык, а необходимость, позволяющая проектировать масштабируемые и экономичные сети. В эпоху облачных технологий и интернета вещей CIDR остается фундаментом, без которого немыслима современная цифровая инфраструктура.

Таким образом, CIDR стала спасательным кругом для IPv4, продлив его жизненный цикл и заложив основы для IPv6. Ее роль выходит за рамки технического стандарта – это философия эффективного использования ресурсов, где каждый бит адреса имеет значение.

2.2. Порядок и основные правила выполнения заданий

При проведении расчетов количества хостов и подсетей целесообразно использовать сведения, приведенные в табл. 2.2.

Таблица 2.2

Основные сведения для расчетов количества хостов и подсетей

Маска подсети	Префикс маски	Двоичная запись маски
0.0.0.0	/0	00000000.00000000.00000000.00000000
128.0.0.0	/1	10000000.00000000.00000000.00000000
192.0.0.0	/2	11000000.00000000.00000000.00000000
224.0.0.0	/3	11100000.00000000.00000000.00000000
240.0.0.0	/4	11110000.00000000.00000000.00000000
248.0.0.0	/5	11111000.00000000.00000000.00000000
252.0.0.0	/6	11111100.00000000.00000000.00000000
254.0.0.0	/7	11111110.00000000.00000000.00000000
255.0.0.0	/8	11111111.00000000.00000000.00000000
255.128.0.0	/9	11111111.10000000.00000000.00000000
255.192.0.0	/10	11111111.11000000.00000000.00000000
255.224.0.0	/11	11111111.11100000.00000000.00000000
255.240.0.0	/12	11111111.11110000.00000000.00000000
255.248.0.0	/13	11111111.11111000.00000000.00000000
255.252.0.0	/14	11111111.11111100.00000000.00000000
255.254.0.0	/15	11111111.11111110.00000000.00000000
255.255.0.0	/16	11111111.11111111.00000000.00000000
255.255.128.0	/17	11111111.11111111.10000000.00000000
255.255.192.0	/18	11111111.11111111.11000000.00000000
255.255.224.0	/19	11111111.11111111.11100000.00000000
255.255.240.0	/20	11111111.11111111.11110000.00000000
255.255.248.0	/21	11111111.11111111.11111000.00000000
255.255.252.0	/22	11111111.11111111.11111100.00000000
255.255.254.0	/23	11111111.11111111.11111110.00000000
255.255.255.0	/24	11111111.11111111.11111111.00000000
255.255.255.128	/25	11111111.11111111.11111111.10000000
255.255.255.192	/26	11111111.11111111.11111111.11000000
255.255.255.224	/27	11111111.11111111.11111111.11100000
255.255.255.240	/28	11111111.11111111.11111111.11110000
255.255.255.248	/29	11111111.11111111.11111111.11111000
255.255.255.252	/30	11111111.11111111.11111111.11111100
255.255.255.254	/31	11111111.11111111.11111111.11111110
255.255.255.255	/32	11111111.11111111.11111111.11111111

Решение типичных заданий рассмотрим на примерах.

Пример 1

Предположим, у нас есть **IP-адрес сети 192.168.0.0/24**. Это означает, что у нас есть **24 бита для сети** и **8 бит для хоста**. Для расчета количества доступных

хостов в этой сети мы используем формулу $2^n - 2$, где **n** – количество битов для хоста. В данном случае **n = 8**, поэтому количество доступных хостов будет $2^8 - 2 = 254$.

Таким образом, **диапазон IP-адресов для хостов** в этой сети будет в пределах **от 192.168.0.1 до 192.168.0.254**.

Пример 2

Предположим, у нас есть **IP-адрес сети 10.0.0.0/16**. Это означает, что мы имеем **16 бит для сети** и **16 бит для хоста**. Для расчета количества доступных хостов в этой сети мы используем формулу $2^n - 2$, где **n** – количество битов для хоста. В данном случае **n = 16**, поэтому количество доступных хостов будет $2^{16} - 2 = 65534$. Следовательно, диапазон IP-адресов для хостов в этой сети будет **от 10.0.0.1 до 10.0.255.254**.

Пример 3

Предположим, у нас есть **IP-адрес сети 172.16.0.0/20**. Это означает, что у нас есть **20 бит для сети** и **12 бит для хоста**. Для расчета количества доступных хостов в этой сети мы используем формулу $2^n - 2$, где **n** – количество битов для хоста. В данном случае **n = 12**, поэтому количество доступных хостов будет $2^{12} - 2 = 4094$. Диапазон IP-адресов для хостов в этой сети будет **от 172.16.0.1 до 172.16.15.254**.

Пример 4

Предположим, что необходимо создать две подсети, что требуется для изолирования серверов в отдельной зоне в интересах повышения безопасности локальной сети. Пусть локальной сети определен IP-адрес **172.16.1.0/24**. В данном случае **номер сети** представляет собой первые три октета IP-адреса (**172.16.1.**), а **идентификатор хоста представлен четвертым октетом**.

Для расчета количества доступных хостов используем формулу $2^n - 2$, где **n** – количество битов для хоста. В данном случае **n = 8**, поэтому количество доступных хостов будет $2^8 - 2 = 254$. На рис. 2.6 представлена локальная сеть до деления на подсети.

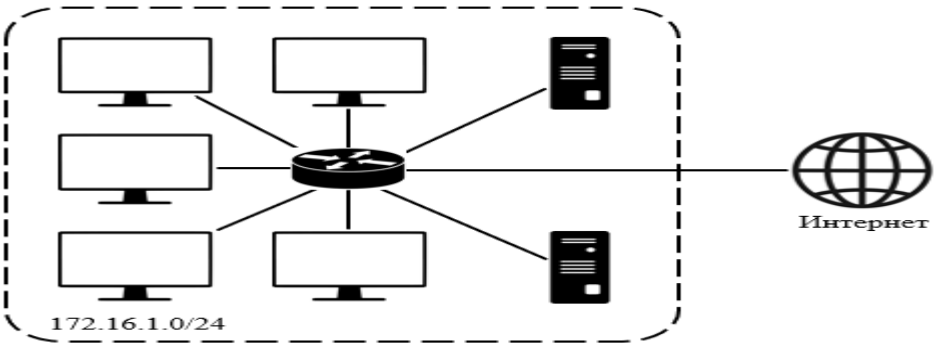

Рис. 2.6. Локальная сеть организации до деления на подсети

Для деления сети **172.16.1.0** на две подсети воспользуемся возможностью заимствования одного бита из идентификатора хоста, тогда **маска подсети станет 25-битной (255.255.255.128 или /25)**.

Переданный **бит идентификатора хоста** представляет собой либо нуль, либо единицу, что позволяет иметь **две подсети: 172.16.1.0/25 и 172.16.1.128/25**.

А следовательно, **локальная сеть будет состоять из двух подсетей: 1 и 2** (рис. 2.7).

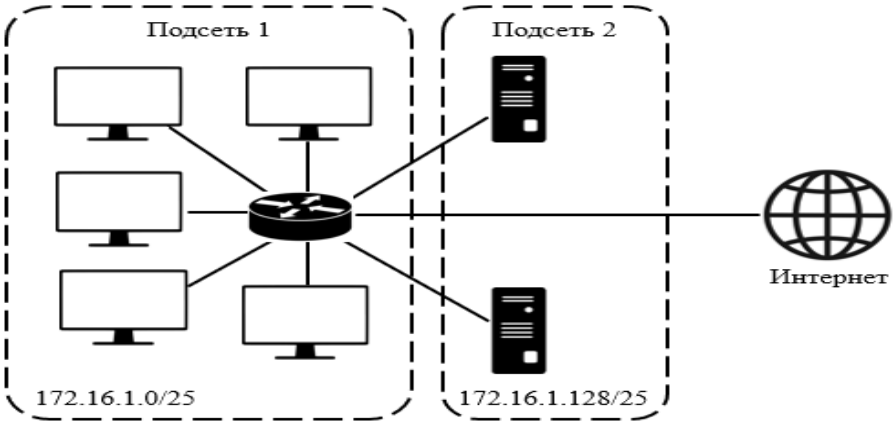

Рис. 2.7. Формирование подсетей после деления сети

Для расчета количества доступных хостов в 25-битной подсети мы снова используем формулу $2^n - 2$. В данном случае n = 7 бит, поэтому количество доступных хостов будет $2^7 - 2 = 126$ хостов.

Таким образом, диапазон IP-адресов для хостов в подсети 1 будет от 172.16.1.1 до 172.16.1.126, а в подсети 2 – от 172.16.1.129 до 172.16.1.254.

При этом адрес 172.16.1.0 с маской 255.255.255.128 будет адресом подсети 1, а 172.16.1.127 с маской 255.255.255.128 – ее широковещательным адресом.

Соответственно, для подсети 2 адресом является 172.16.1.128, ее широковещательный адрес – 172.16.1.255.

Пример 5

Предположим, что для функционирования предприятия необходимо создать восемь подсетей.

Пусть для локальной сети определен IP-адрес 172.16.1.0/24. В данном случае номер сети представляют собой первые три октета IP-адреса (172.16.1.), а идентификатор хоста представлен четвертым октетом.

Для расчета количества доступных хостов используем формулу $2^n - 2$, где n – количество битов для хоста. В данном случае n = 8, поэтому количество доступных хостов будет $2^8 - 2 = 254$.

Как показано в примере 4 данного практического занятия, воспользуемся возможностью заимствования битов. В этом случае необходимо использовать **3 бита** четвертого октета.

Маска подсети будет выглядеть как **11111111.11111111.11111111.11100000** или **255.255.255.224**.

В каждой подсети имеется по **5 бит идентификатора хоста**. Рассчитав по ранее указанной формуле, мы получим $\mathbf{2^5 - 2 = 30}$ **хостов**.

Кроме того, учитываем, что **идентификатор хоста из всех нулей – это сама подсеть, а из всех единиц – ее широковещательный адрес**.

Сведения об адресах хостов восьми подсетей, полученные в результате несложного расчета, представлены в табл. 2.3.

Таблица 2.3

Распределение адресов для подсетей (на примере сети 172.16.1.0)

Номер подсети	Адрес подсети	Первый адрес	Последний адрес	Широковещательный адрес
1	0	1	30	31
2	32	33	62	63
3	64	65	94	95
4	96	97	126	127
5	128	129	158	159
6	160	161	190	191
7	192	193	222	223
8	224	225	254	255

Пример 6

Рассмотрим случай, когда организация имеет сеть с адресом **172.16.242.210/26** и маской подсети **255.255.255.192**. Нужно разбить сеть на три подсети, из которых одна будет отличаться от двух других.

При расчете **IP-адресов** в подсети и сетевых масок применяются те же методика и формула, которые используются в предыдущих примерах.

При выбранных параметрах полученные основные сведения по подсетям представлены в табл. 2.4.

Таблица 2.4

Основные сведения по полученным подсетям

Номер подсети	1	2	3
Требуемый размер	30 + 2	14 + 2	14 + 2
Выделено адресов	32	16	16
Остаток свободных адресов	0	0	0
IP-адрес подсети	172.16.242.192	172.16.242.224	172.16.242.240
Маска подсети	255.255.255.224	255.255.255.240	255.255.255.240
Префикс маски	/27	/28	/28
Диапазон адресов	172.16.242.193–172.16.242.222	172.16.242.225–172.16.242.238	172.16.242.241–172.16.242.254
Широковещание	172.16.242.223	172.16.242.239	172.16.242.255

2.3. Практическое задание

2.3.1. Провести расчеты количества хостов и подсетей на основе IP-адреса и маски в соответствии с вариантом, выданным преподавателем.

2.3.2. Результаты представить в табличном виде по форме, указанной в табл. 2.5.

Таблица 2.5

Форма отчета по расчету параметров подсетей

Номер подсети			
Требуемый размер			
Выделено адресов			
Остаток свободных адресов			
IP-адрес подсети			
Маска подсети			
Префикс маски			
Диапазон адресов			
Широковещание			

2.3.3. Ответить на контрольные и дополнительные вопросы.

Варианты заданий

Варианты заданий с учетом IP-адреса и маски указаны в табл. 2.6.

Таблица 2.6

Перечень вариантов заданий

Вариант	IP-адреса и количество подсетей	Вариант	IP-адреса и количество подсетей
1	192.168.1.0/24, на три подсети	16	219.32.6.0/25, на пять подсетей
2	219.32.6.0/25, на две подсети	17	02.28.36.1/24, на три подсети
3	192.166.1.0/25, на три подсети	18	10.71.28.3/24, на две подсети
4	219.31.5.0/25, на две подсети	19	132.41.88.36/25, на четыре подсети
5	191.168.1.0/26, на три подсети	20	164.12.98.0/25, на три подсети
6	221.32.6.0/25, на две подсети	21	192.168.1.0/24, на четыре подсети

Окончание табл. 2.6

Вариант	IP-адреса и количество подсетей	Вариант	IP-адреса и количество подсетей
7	192.168.1.0/24, на восемь подсетей разного размера	22	164.12.98.0/24, на три подсети
8	219.32.6.0/26, на две подсети	23	132.41.88.36/24, на две подсети
9	03.09.136.2/25, на четыре подсети	24	164.12.98.0/26, на три подсети
10	219.32.18.1/25, на две подсети	25	192.138.1.0/23, на три подсети
11	132.41.88.36/25, на две подсети	26	168.71.19.0/24, на три подсети
12	219.32.6.0/27, на две подсети	27	132.41.88.36/24, на четыре подсети
13	01.162.1.0/24, на пять подсетей	28	168.71.21.0/25, на четыре подсети
14	219.32.6.0/28, на две подсети	29	192.168.1.0/24, на три подсети
15	132.41.88.36/25, на три подсети	30	168.71.19.0/26, на пять подсетей

Контрольные вопросы

1. Перечислите особенности проведения межсетевого обмена в сетях.
2. Каковы роль и значение стека протоколов TCP/IP на современном этапе?
3. Назовите порядок работы стека протоколов TCP/IP.
4. Опишите протокол IPv4.
5. Как работает протокол IPv4?
6. Как выглядит заголовок пакета протокола IPv4?
7. Укажите особенности Р-адреса протокола IPv4.
8. Дайте характеристику работы (соединения) одной сети.
9. Укажите особенности работы (соединения) двух сетей.
10. Кто выдает IP-адреса?
11. Что такое маска подсети?
12. Какие классы имеют IP-адреса протокола IPv4?
13. Опишите процесс использования IP-адресов.
14. Какие логические операторы используются при расчете масок подсетей?
15. Что такое бесклассовая адресация CIDR?
16. Назовите формулу расчета количества битов для хостов.
17. Как определить IP-адрес компьютера?
18. Перечислите режимы адресации протокола IPv4.
19. Для чего и как применяется бесклассовая адресация?
20. Опишите сеть класса С.

Практическое занятие № 3
«Основные принципы работы протоколов TCP и UDP»

Цель работы: изучить основные принципы работы протоколов транспортного уровня TCP и UDP.

3.1. Краткие теоретические сведения

3.1.1. Транспортный уровень в модели TCP/IP

Транспортный уровень в модели TCP/IP является важным звеном, обеспечивающим передачу данных между приложениями на разных устройствах. **Его основная задача** – организовать доставку информации независимо от того, какие данные передаются и между какими узлами сети.

Основные задачи транспортного уровня:

- **деление данных на части.** Транспортный уровень разбивает большие объемы данных на более мелкие части – сегменты (для TCP) или дейтаграммы (для UDP). Это позволяет эффективно передавать информацию через сеть;
- **регулирование скорости передачи.** Протоколы транспортного уровня регулируют скорость передачи данных, чтобы избежать перегрузки сети или получателя;
- **обеспечение надежности.** TCP, например, проверяет целостность данных и при необходимости запрашивает повторную передачу;
- **мультиплексирование.** Благодаря уникальным номерам портов несколько приложений могут одновременно использовать сетевые ресурсы.

К протоколам транспортного уровня относятся два протокола.

1. TCP (Transmission Control Protocol). Это надежный протокол, который устанавливает соединение между отправителем и получателем перед началом передачи данных. TCP гарантирует, что данные будут доставлены без ошибок и в правильном порядке.

Он используется в приложениях, где важна точность передачи, например, в веб-браузерах (HTTP/HTTPS), электронной почте (IMAP – Internet Message Access Protocol – протокол прикладного уровня для доступа к электронной почте) и передаче файлов (FTP).

2. UDP (User Datagram Protocol). Это более простой и быстрый протокол, который не устанавливает соединение и не гарантирует доставку данных.

UDP подходит для приложений, где важна скорость, а небольшие потери данных допустимы. Например, его используют для потокового видео, VoIP (голосовая связь через Интернет) и онлайн-игр.

Для идентификации приложений на устройстве используются **порты –** числовые идентификаторы в диапазоне от 0 до 65535. Порты делятся на три категории:

- **общеизвестные порты (0–1023)** закреплены за популярными службами, такими как HTTP (порт 80), HTTPS (порт 443) или FTP (порт 21);

- **зарегистрированные порты (1024–49151)** используются для менее распространенных приложений;
- **динамические порты (49152–65535)** назначаются операционной системой временно для конкретных задач.

Чтобы однозначно идентифицировать приложение в сети, используется **сокет –** комбинация IP-адреса и номера порта. Например:

- **TCP-сокет:** 172.16.1.10:80 (веб-сервер на порту 80);
- **UDP-сокет:** 10.0.1.5:53 (DNS-сервер на порту 53).

Порядок передачи данных заключается в следующем.

Приложение отправляет данные на транспортный уровень, указывая целевой IP-адрес и порт. Транспортный уровень упаковывает данные в сегменты (TCP) или дейтаграммы (UDP). Эти данные передаются на сетевой уровень (IP), где добавляется IP-заголовок.

На канальном уровне информация передается в виде кадров через физическую сеть.

На стороне получателя данные проходят обратный процесс, начиная с канального уровня и заканчивая приложением.

Транспортный уровень обеспечивает гибкость и надежность передачи данных, адаптируясь к требованиям различных приложений. Благодаря ему приложения могут эффективно взаимодействовать в сети независимо от их типа и назначения.

Протокол TCP обеспечивает надежную и упорядоченную доставку информации между устройствами. Для передачи данных TCP использует **сегменты**. Каждый сегмент содержит заголовок и полезные данные (payload). Заголовок TCP-сегмента включает несколько важных полей, которые обеспечивают корректную работу протокола (рис. 3.1).

<table>
<tr><th>Бит</th><th>0-3</th><th>4-6</th><th>7-15</th><th>16-31</th></tr>
<tr><td>0</td><td colspan="3">Порт источника, Source Port</td><td>Порт назначения, Destination Port</td></tr>
<tr><td>32</td><td colspan="4">Порядковый номер, Sequence Number (SN)</td></tr>
<tr><td>64</td><td colspan="4">Номер подтверждения, Acknowledgement Number (ACK SN)</td></tr>
<tr><td>96</td><td>Длина заголовка, (Data offset)</td><td>Зарезервировано</td><td>Флаги</td><td>Размер окна, Window size</td></tr>
<tr><td>128</td><td colspan="3">Контрольная сумма, Checksum</td><td>Указатель важности, Urgent Point</td></tr>
<tr><td>160</td><td colspan="4">Опции (необязательное, но используется практически всегда)</td></tr>
<tr><td>160/192+</td><td colspan="4">Данные</td></tr>
</table>

Рис. 3.1. Структура заголовка сегмента TCP

Заголовок TCP-сегмента состоит из следующих полей.

Порт источника (Source Port) идентифицирует приложение на стороне отправителя, которое инициировало передачу данных. Он также применяется для отправки ответа клиенту.

Порт назначения (Destination Port) указывает порт приложения на стороне получателя, которому предназначены данные.

Порядковый номер (Sequence Number, SN) – уникальный номер, который идентифицирует каждый байт данных в потоке. При установлении соединения (флаг SYN) используется начальный порядковый номер (ISN), который генерируется случайным образом для повышения безопасности. Первый байт полезных данных в сессии имеет номер ISN + 1.

Номер подтверждения (Acknowledgment Number, ACK SN) указывает порядковый номер следующего ожидаемого байта данных и подтверждает успешное получение всех предыдущих данных.

Длина заголовка (Data Offset) определяет размер заголовка в 32-битных словах. Минимальный размер заголовка – 20 байт, максимальный – 60 байт.

Зарезервированные биты (Reserved) – биты, которые зарезервированы для будущего использования и должны быть установлены в нуль.

Флаги (управляющие биты) определяют состояние соединения:

- **NS (ECN-nonce)** используется для защиты от злоупотреблений механизмом **ECN** (Explicit Congestion Notification, явное уведомление о перегруженности);
- **CWR (Congestion Window Reduced)** указывает на уменьшение окна перегрузки;
- **ECE (ECN-Echo)** сигнализирует о перегрузке в сети;
- **URG (Urgent)** указывает на наличие срочных данных;
- **ACK (Acknowledgment)** подтверждает получение данных;
- **PSH (Push)** требует немедленной передачи данных приложению;
- **RST (Reset)** сбрасывает соединение;
- **SYN (Synchronize)** инициирует установление соединения;
- **FIN (Finish)** завершает соединение.

Размер окна (Window Size) определяет количество байт, которые получатель готов принять без подтверждения. Используется для управления потоком данных.

Контрольная сумма (Checksum) проверяет целостность заголовка и данных. Вычисляется как 16-битное дополнение к сумме всех 16-битных слов заголовка и данных.

Указатель важности (Urgent Pointer) указывает на конец срочных данных, если установлен флаг URG.

Опции (Options) – дополнительные параметры, которые могут использоваться для расширения функциональности протокола.

Процесс установления соединения по протоколу TCP, известный как **трехэтапное рукопожатие** (three-way handshake), включает три этапа.

Для детального понимания процесса установления соединения в компьютерной сети приведем в качестве информации таблицу состояний сеанса протокола TCP, сведения из которой будут использованы при рассмотрении всех этапов (табл. 3.1).

Таблица 3.1

Описание состояний соединения TCP

Название команды	Состояние сеанса TCP
CLOSED	Начальное состояние узла. Фактически фиктивное
LISTEN	Сервер ожидает запросы установления соединения от клиента
SYN-SENT	Клиент отправил серверу запрос на установление соединения и ожидает ответа
SYN-RECEIVED	Сервер получил запрос на соединение, отправил ответный запрос и ожидает подтверждения
ESTABLISHED	Соединение установлено, идет передача данных
FIN-WAIT-1	Одна из сторон (узел 1) завершает соединение, отправив сегмент с флагом FIN
CLOSE-WAIT	Другая сторона (узел 2) переходит в это состояние, отправив, в свою очередь, сегмент ACK, и продолжает одностороннюю передачу
FIN-WAIT-2	Узел 1 получает ACK, продолжает чтение и ждет получения сегмента с флагом FIN
LAST ACK	Узел 2 заканчивает передачу и отправляет сегмент с флагом FIN
TIME-WAIT	Узел 1 получил сегмент с флагом FIN, отправил сегмент с флагом ACK и ждет $2 \cdot MSL$ секунд перед окончательным закрытием соединения
CLOSING	Обе стороны инициировали закрытие соединения одновременно: после отправки сегмента с флагом FIN узел 1 получает сегмент FIN, отправляет ACK и находится в ожидании сегмента ACK (подтверждение на свой запрос о разъединении)

Этап 1 – инициация соединения.

Клиент, желающий установить соединение, отправляет серверу сегмент с установленным флагом **SYN** и указанием номера последовательности. Сервер, получив этот сегмент, сохраняет номер последовательности и пытается создать сокет для обслуживания нового клиента. Если сокет успешно создан, сервер отправляет клиенту сегмент с флагами **SYN** и **ACK**, переходя в состояние **SYN-RECEIVED**. В случае ошибки сервер отправляет клиенту сегмент с флагом **RST**, сигнализируя об отказе.

Клиент, получив сегмент с флагом **SYN**, сохраняет номер последовательности и отправляет серверу сегмент с флагом **ACK**. Если клиент одновременно получает флаг **ACK** (что происходит в большинстве случаев), он переходит в состояние **ESTABLISHED**. Если же клиент получает сегмент с флагом **RST**, попытки установить соединение прекращаются. При отсутствии ответа от сервера в течение 10 с клиент повторяет процесс.

Сервер, находясь в состоянии **SYN-RECEIVED**, при получении сегмента с флагом ACK переходит в состояние **ESTABLISHED**. Если ответа от клиента не поступает, сервер закрывает сокет и переходит в состояние **CLOSED**.

Этап 2 – передача данных.

Во время обмена данными получатель использует номера последовательностей в полученных сегментах для восстановления их исходного порядка. Получатель уведомляет передающую сторону о номере последовательности, до которой данные были успешно получены, включая этот номер в поле «Номер подтверждения». Данные, относящиеся к уже подтвержденным последовательностям, игнорируются. Если номер последовательности в полученном сегменте превышает ожидаемый, данные буферизируются, но номер подтвержденной последовательности не изменяется.

При получении сегмента с ожидаемым номером последовательности порядок данных восстанавливается автоматически. Для контроля скорости передачи данных TCP использует механизм управления потоком, основанный на поле «Окно».

В некоторых случаях передающая сторона может использовать флаг **PSH** для немедленной передачи данных без буферизации. Например, в интерактивных приложениях, таких как сетевые терминалы, флаг **PSH** применяется для оперативной обработки введенных пользователем команд.

Этап 3 – завершение соединения. Проводится за три шага.

Шаг 1. Клиент отправляет серверу сегмент с флагом FIN, сигнализируя о намерении закрыть соединение.

Шаг 2. Сервер отвечает клиенту сегментом с флагами ACK и FIN, подтверждая закрытие соединения.

Шаг 3. Клиент, получив эти флаги, закрывает соединение и отправляет серверу сегмент с флагом ACK в качестве окончательного подтверждения.

Таким образом, процесс установления, передачи данных и завершения соединения в TCP обеспечивает надежную и упорядоченную коммуникацию.

3.1.2. Протокол UDP и его дейтаграммы

UDP – это транспортный протокол стека протоколов TCP/IP, предназначенный для передачи данных без предварительного установления соединения. В отличие от TCP он не требует сложных процедур согласования (рукопожатий), что делает его менее ресурсоемким, но и одновременно менее надежным.

Приложения, использующие UDP, отправляют пакеты информации, называемые дейтаграммами, напрямую между узлами сети, не создавая выделенных каналов связи.

Отмечаются следующие особенности UDP:

- **отсутствие гарантий доставки.** Дейтаграммы могут теряться, дублироваться или приходить в неправильном порядке;

- **мультиплексирование.** Номера портов позволяют одному узлу обрабатывать несколько сетевых сервисов одновременно;
- **контроль ошибок.** Проверка контрольной суммы помогает выявить искажения данных, но восстановление информации не выполняется – эту задачу решают вышестоящие приложения;
- **управление потоком.** Отсутствие встроенных механизмов регулирования скорости передачи требует реализации дополнительной логики на уровне приложений, если необходима надежная доставка;
- **минимальные накладные расходы.** Благодаря простой структуре заголовка UDP обеспечивает высокую скорость передачи, что критично для приложений, чувствительных к задержкам;
- **широкое применение.** Используется в DNS-запросах, VoIP-сервисах, потоковом вещании (IPTV), онлайн-играх и VPN, где важна скорость, а потеря отдельных пакетов некритична.

Структура дейтаграммы UDP включает заголовок в четыре поля, каждое из которых занимает 2 байта (рис. 3.2).

Биты	0-15	16-31
0-31	Порт отправителя (Source port)	Порт получателя (Destination port)
32-63	Длина дейтаграммы (Length)	Контрольная сумма (Checksum)
64- …	Данные (Data)	

Рис. 3.2. Структура дейтаграммы UDP

Рассмотрим их более подробно.

Порт получателя – определяет конечную точку доставки данных на целевом узле.

Порт отправителя – указывает порт для возможного ответа. В **IPv6** это поле может опускаться.

Длина дейтаграммы – общий размер заголовка и данных в байтах. Минимальное значение – 8 байт (только заголовок), максимальное – 65 535 байт (из них 65 527 байт отводится на данные).

Контрольная сумма – проверяет целостность заголовка и данных. В IPv4 это поле необязательно и может заполняться нулями, в IPv6 – обязательно.

Таким образом, **UDP** идеально подходит для сценариев, где скорость и минимальная задержка важнее гарантированной доставки, а ответственность за обработку ошибок возложена на конечное программное обеспечение.

3.1.3. Протокол DHCP: автоматизация сетевых настроек

Назначение IP-адресов в сети может осуществляться двумя способами:

1. Статическое назначение. Администратор вручную прописывает IP-адрес для каждого устройства. Такой подход подходит для небольших сетей (домашних или локальных офисных), где число узлов ограничено.

2. Динамическое назначение. Устройства автоматически получают временные IP-адреса при подключении к сети. Этот метод эффективен в крупных сетях, таких как корпоративные офисы, университеты или публичные Wi-Fi-зоны, где количество подключаемых устройств непостоянно.

Протокол динамической настройки узлов (DHCP) обеспечивает автоматизацию распределения IP-адресов и других сетевых параметров (маска подсети, шлюз по умолчанию, DNS-серверы). Работая по модели «клиент – сервер», DHCP минимизирует ручное вмешательство и предотвращает конфликты адресов.

Участники процесса в этом случае:

- **клиент –** устройство (смартфон, ПК, IoT-гаджет), запрашивающее сетевые настройки;
- **сервер –** служба, управляющая пулом IP-адресов и отслеживающая их уникальность.

Этапы взаимодействия (DORA)

Процесс настройки включает четыре шага, обозначаемые аббревиатурой **DORA** (рис. 3.3).

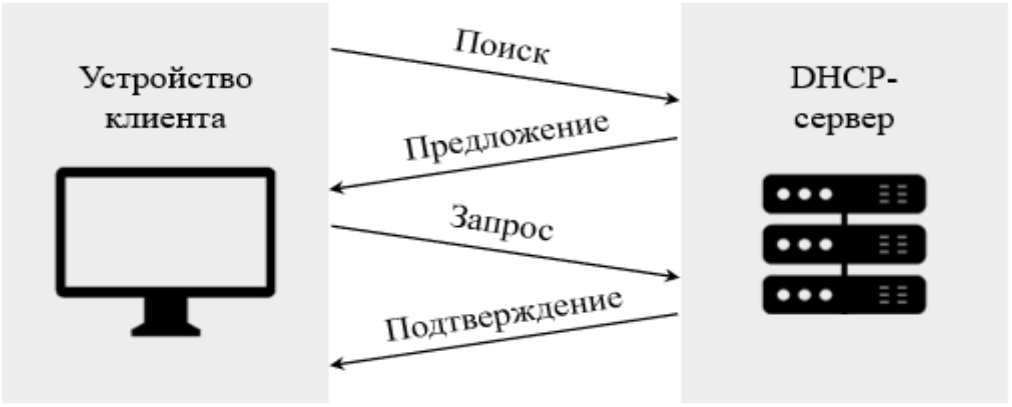

Рис. 3.3. Порядок взаимодействия DHCP-сервера и клиента

Обнаружение (Discover). Клиент, не имеющий IP-адреса, отправляет широковещательный запрос **DHCPDISCOVER** в локальный сегмент сети. Это сообщение достигает всех узлов, но отвечают только DHCP-серверы.

Предложение (Offer). Серверы, получившие запрос, резервируют свободный IP-адрес из своего пула и отправляют клиенту предложение **DHCPOFFER**. В нем указывается адрес, срок аренды и дополнительные параметры. Если клиент получает несколько предложений, он выбирает одно (часто первое полученное).

Запрос (Request). Клиент подтверждает выбор, отправляя **DHCPREQUEST** выбранному серверу. Это сообщение также широковещательное, что уведомляет другие серверы о занятии ранее зарезервированных адресов.

Подтверждение (Acknowledgement, ACK). Сервер финализирует настройки, отправляя **DHCPACK** с окончательными параметрами. Клиент

применяет их и получает доступ к сети. Если IP-адрес становится недоступен (например, занят), сервер отвечает пакетом **DHCPNAK**, инициируя повтор процесса.

DHCP-сервер назначает адреса на ограниченное время (например, 24 ч). По истечении половины срока клиент пытается продлить аренду (**DHCPREQUEST**). Если сервер недоступен, попытки повторяются до окончания срока, после чего адрес освобождается.

Преимущества DHCP:

- **масштабируемость.** Поддержка множества устройств без ручного управления;
- **гибкость.** Настройка параметров сети централизованно через сервер;
- **экономия ресурсов.** Автоматическое освобождение неиспользуемых адресов.

Таким образом, **DHCP** является ключевым элементом современных сетей, обеспечивая простоту подключения и эффективное использование IP-ресурсов.

3.1.4. Взаимодействие протоколов стека TCP/IP при обработке запросов

Когда пользователь вводит URL-адрес в браузере или переходит по ссылке, запускается цепочка процессов, обеспечивающих доставку данных с сервера на клиентское устройство. Рассмотрим этапы работы стека TCP/IP в этом сценарии.

Инициация запроса. Пользователь через браузер формирует **HTTP-запрос**, содержащий требования к серверу: например, получение **HTML**-страницы, изображений или скриптов (рис. 3.4).

Рис. 3.4. Порядок прохождения HTTP-сообщений

Однако сам по себе HTTP не определяет, как найти сервер или гарантировать доставку данных. Для этого задействуются нижележащие протоколы – TCP и IP.

Определение IP-адреса через DNS. Перед отправкой запроса система должна определить, на какой сервер его направить. Для преобразования доменного имени (например, bsuir.by) в числовой **IP-адрес** используется **система доменных имен (DNS)**. Это распределенная база данных, хранящая записи о соответствии доменов IP-адресам, а также информацию о почтовых серверах (MX-записи) и службах (SRV-записи).

DNS-resolver, встроенный в операционную систему или предоставляемый провайдером, выполняет поиск в иерархии серверов, пока не получит нужный IP.

Сегментация данных и маршрутизация. После определения IP-адреса сервера протокол **IP** отвечает за маршрутизацию данных через сеть. Информация, которую сервер должен передать (например, веб-страница), разбивается на **пакеты** размером до 64 КБ. Каждый пакет содержит:

- **заголовок IP** с адресами отправителя и получателя;
- **полезную нагрузку –** фрагмент передаваемых данных;
- **идентификатор сегмента** для последующей сборки.

Гарантия доставки через TCP. Протокол **TCP** обеспечивает надежную передачу пакетов. Перед началом обмена данными между клиентом и сервером устанавливается соединение через **трехэтапное рукопожатие**. Далее TCP контролирует целостность данных:

- каждый полученный пакет подтверждается ответом **ACK**;
- при потере или искажении пакета TCP инициирует его повторную отправку;
- данные собираются в правильном порядке благодаря нумерации сегментов.

Финальный этап: обработка данных браузером. Когда все пакеты успешно доставлены, TCP восстанавливает исходный порядок данных, после чего браузер:

- **интерпретирует код** (HTML, CSS, JavaScript);
- **загружает дополнительные ресурсы** (изображения, шрифты) через новые HTTP-запросы;
- **визуализирует страницу**, преобразуя код в визуальные элементы.

Это многоуровневое взаимодействие позволяет пользователям получать доступ к веб-ресурсам, даже если серверы находятся за тысячи километров, а данные проходят через десятки промежуточных узлов.

3.2. Порядок и основные правила выполнения заданий

Для изучения основных принципов работы **протоколов TCP и UDP** используется **симулятор Cisco Packet Tracer**.

Порядок и правила выполнения заданий рассмотрим в ходе конкретного примера.

В работе будут также задействованы:

- **DNS;**
- **HTTP;**
- **DHCP.**

3.2.1. Построение структурной схемы локальной сети

Запустим программу **Cisco Packet Tracer**. В области **Логическое пространство** построим структурную схему, состоящую из компьютера (PC-PT), двух серверов (Server-PT), коммутатора (2950-24) и маршрутизатора (2901), как указано на рис. 3.5.

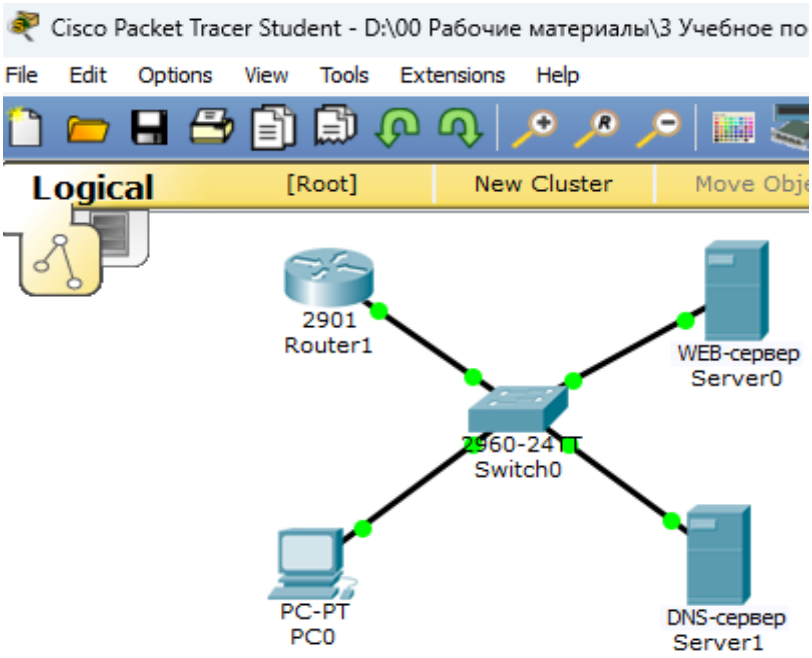

Рис. 3.5. Структурная схема сети

3.2.2. Конфигурирование маршрутизатора (Router1) с настроенным DHCP-сервером

Для конфигурирования **Router1** необходимо определить его интерфейс, который подключен к **Switch0**.

На рис. 3.5 линия связи от **Switch0** соединена с интерфейсом **Gig0/0**. Следовательно, настройке подлежит интерфейс **GigabitEthernet0/0**.

Наводим курсор на **Router1** и нажимаем левую кнопку мыши. Далее задействуем вставки **Config → INTERFACE → GigabitEthernet0/0**.

В появившейся вкладке включаем **Port Status** путем нажатия кнопки **On** и задаем для **Router1** на **интерфейсе GigabitEthernet0/0 IP-адрес – 192.168.1.1 с маской 255.255.255.0** (рис. 3.6).

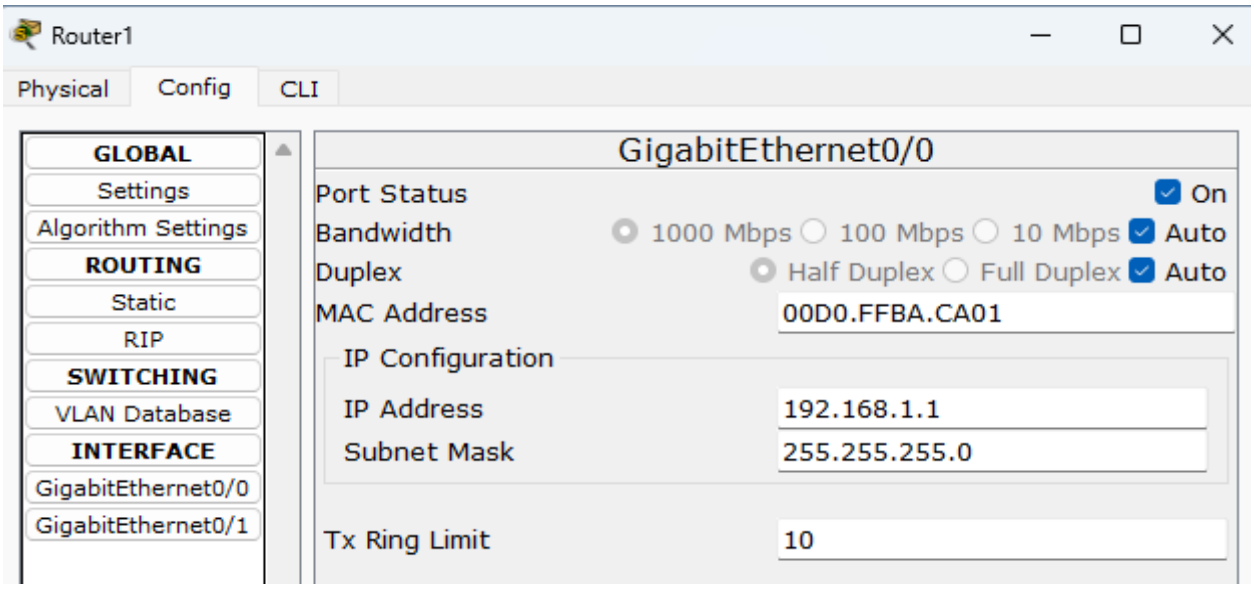

Рис. 3.6. Настройка интерфейса маршрутизатора

Затем через вкладку **CLI** заходим в **IOS Command Line Interface** для конфигурирования **DHCP-сервера** и исключения статических адресов, которые определены роутеру 192.168.1.1 и серверам (**DNS-сервер** – 192.168.1.2 и **веб-сервер** – 192.168.1.3). С учетом проведенной настройки интерфейса маршрутизатора необходимо провести действия, представленные на рис. 3.7.

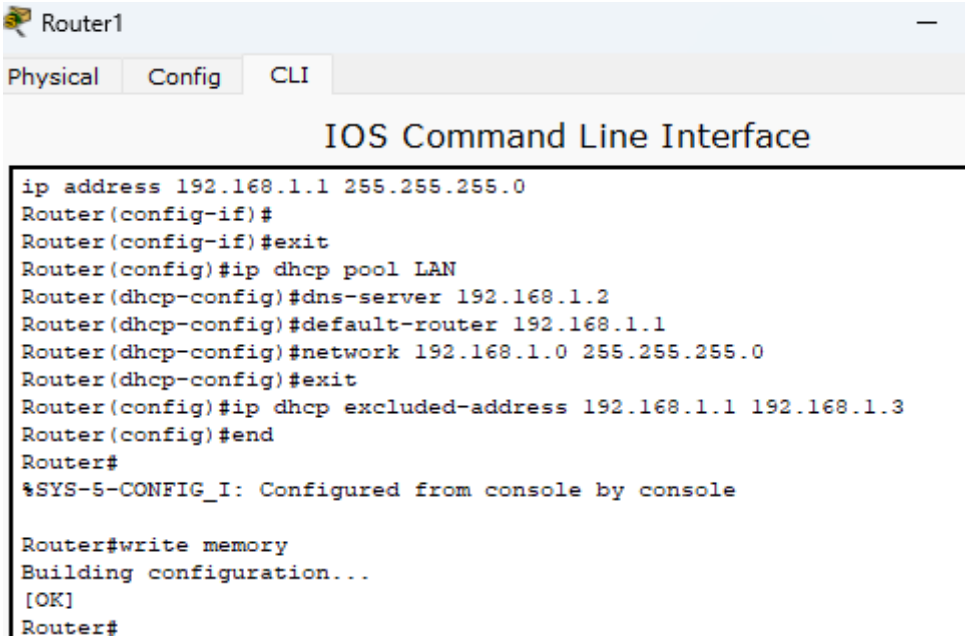

Рис. 3.7. Конфигурирование DHCP-сервера и исключение статических адресов

3.2.3. Настройка DNS-сервера (Server1)

Для настройки **Server1** необходимо определить его интерфейс, который подключен к **Switch0**.

На рис. 3.5 линия связи от **Switch1** присоединена к интерфейсу **Fa0 DNS-сервера**. Следовательно, настройке подлежит интерфейс **FastEthernet0/0**.

Для настройки **Server1** необходимо на нем открыть вкладку **Config** и далее вкладку **FastEthernet0/0**. Заносим ранее выданный **Switch1 IP-адрес – 192.168.1.2 и маску сети 255.255.255.0** (рис. 3.8).

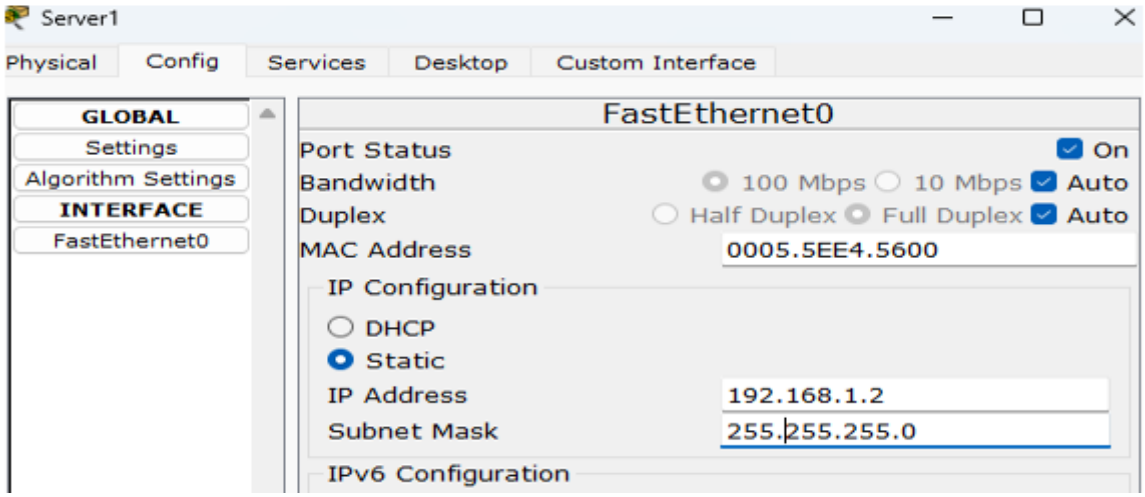

Рис. 3.8. Заполнение IP-адреса DNS-сервера

Далее следует открыть вкладку **Services → DNS** и нажать кнопку **On**.

Заполняем позицию **Name** любым названием сайта, например **ipe-bsuir.by**, и в графе **Adress** указываеv **IP-адрес сервера**, обслуживающего данный сайт. В нашем случае **веб-сервер** имеет **IP-адрес 192.168.1.3**. После этого выбираем вкладку **Add** и указанная информация формируется внизу в виде таблицы (рис. 3.9).

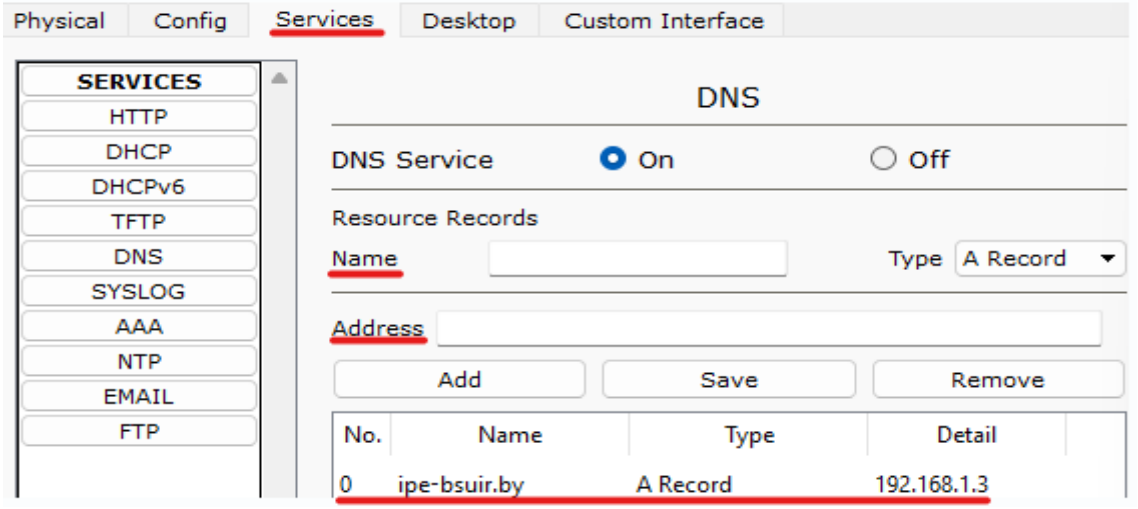

Рис. 3.9. Общий вид заполненной DNS-вкладки DNS-сервера

3.2.4. Настройка веб-сервера (Server0)

Для настройки **Server0** необходимо определить его интерфейс, который подключен к **Switch0**.

На рис. 3.5 линия связи от **Switch0** присоединена к интерфейсу **Fa0 веб-сервера**. Следовательно, настройке подлежит интерфейс **FastEthernet0/0**.

Для настройки **Server0** необходимо открыть вкладку **Config** и далее вкладку **FastEthernet0/0**. Заносим ранее выданный **Server0 IP-адрес – 192.168.1.3 и маску сети 255.255.255.0** (рис. 3.10).

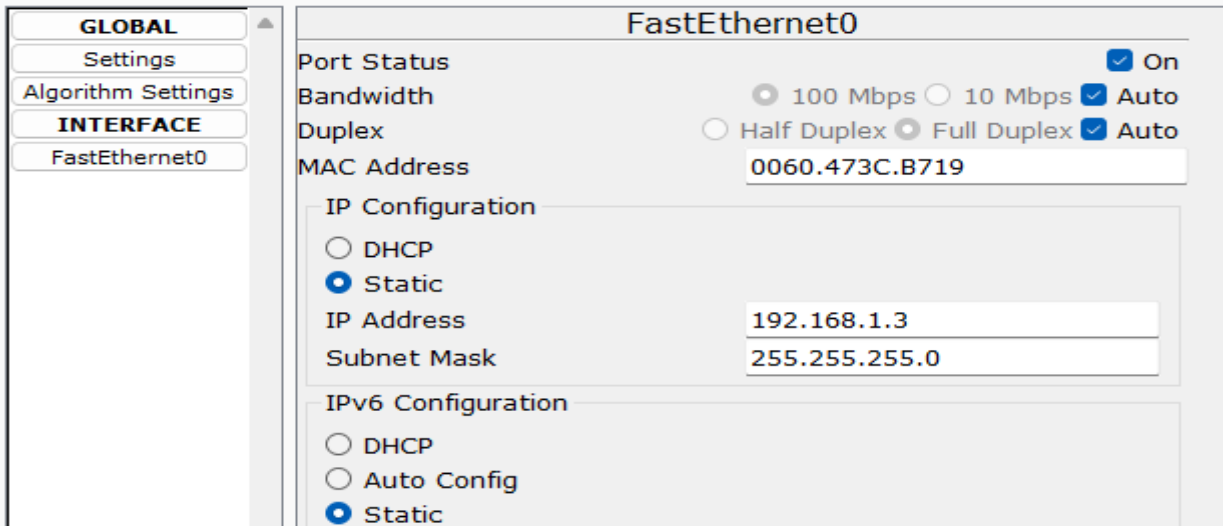

Рис. 3.10. Заполнение IP-адреса веб-сервера

Затем следует открыть вкладки **Services → HTTP** и, при необходимости, включить **HTTP** и **HTTPS**, нажав кнопку **On** (рис. 3.11).

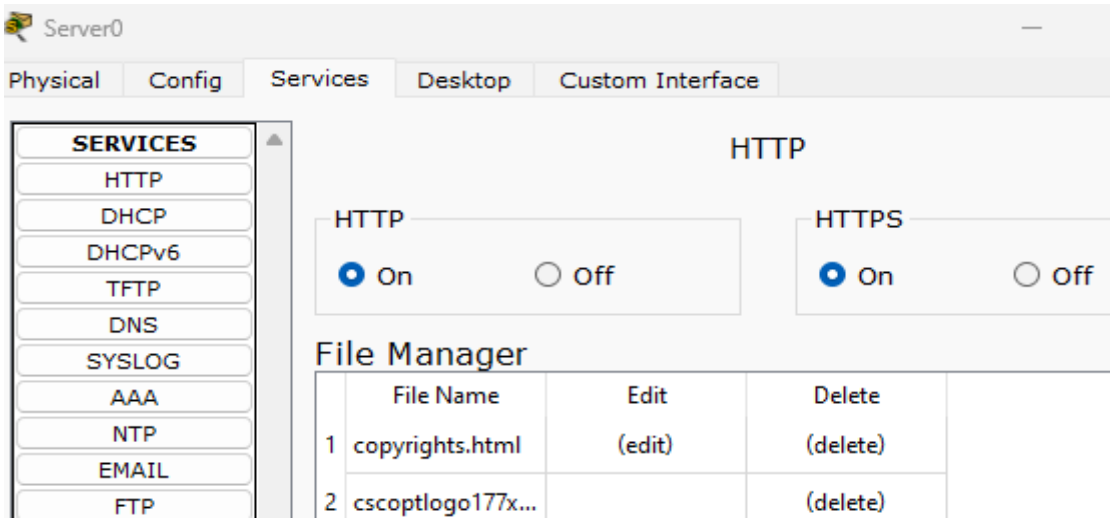

Рис. 3.11. Состояние настроенного веб-сервера

3.2.5. Настройка хоста (компьютера PC0)

Открываем вкладки компьютера **Desktop → IP Configuration**. Во вкладке **IP Configuration** активируем кнопки **DHCP**. После этого **протокол DHCP** автоматически назначает компьютеру **PC0 IP-адрес** – **192.168.1.4** (рис. 3.12).

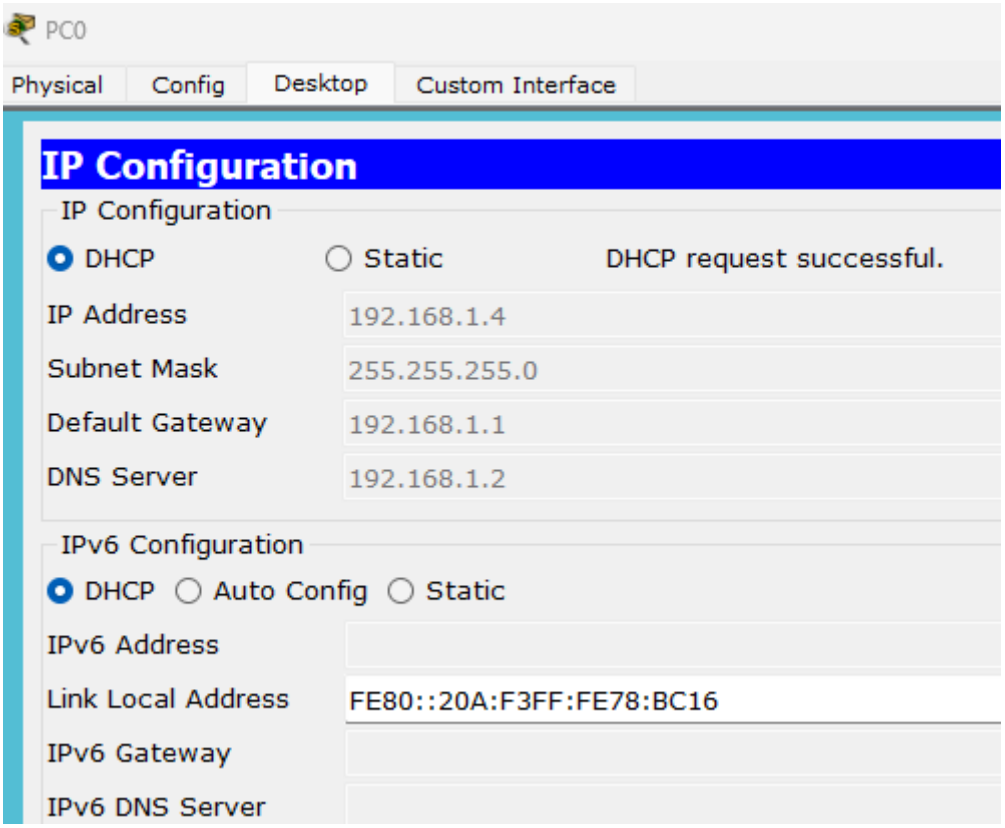

Рис. 3.12. Назначение компьютеру IP-адреса при помощи протокола DHCP

Таким образом, построенная нами локальная сеть настроена.

3.2.5. Контроль функционирования локальной сети

Для осуществления контроля функционирования локальной сети необходимо открыть **Web Browser** компьютера **PC0** и в строке **URL** записать выбранный ранее сайт – **ipe-bsuir.by**. После нажатия кнопки **Go** появляется надпись, которая свидетельствует о возможности получения информации от веб-сервера (рис. 3.13).

Рис. 3.13. Успешный выход на сайт ipe-bsuir.by

3.2.6. Анализ этапов прохождения пакетов

Для наглядного представления функционирования компьютерной сети переведем **Cisco Packet Tracer** в **режим симуляции**.

Откроем **Web Broser** компьютера **PC0** и в строке **URL** запишем домен ранее выбранного условного сайта – **ipe-bsuir.by**. Далее нажимаем кнопку **Go** и для запуска поэтапного движения **пакета** на панели **Cisco Packet Tracer** нажимаем кнопку **Capture/Forward**.

Теперь у нас появилась возможность наблюдать на **Event List Simulation Panel** этапы продвижения пакетов (рис. 3.14).

Time(sec)	Last Device	At Device	Type	Info
0.000	--	PC0	DNS	
0.001	PC0	Switch0	DNS	
0.002	Switch0	Server1	DNS	
0.003	Server1	Switch0	DNS	
0.004	--	PC0	TCP	
0.004	Switch0	PC0	DNS	
0.004	--	PC0	TCP	
0.005	PC0	Switch0	TCP	
0.006	Switch0	Server0	TCP	

Рис. 3.14. Этапы продвижения пакетов по сети

0.007	Server0	Switch0	TCP	
0.008	Switch0	PC0	TCP	
0.008	--	PC0	HTTP	
0.009	PC0	Switch0	TCP	
0.009	--	PC0	HTTP	
0.010	PC0	Switch0	HTTP	
0.010	Switch0	Server0	TCP	
0.011	Switch0	Server0	HTTP	
0.012	Server0	Switch0	HTTP	
0.013	--	PC0	TCP	
0.013	Switch0	PC0	HTTP	
0.013	--	PC0	TCP	
0.014	PC0	Switch0	TCP	
0.015	Switch0	Server0	TCP	
0.016	Server0	Switch0	TCP	
0.017	Switch0	PC0	TCP	
0.018	PC0	Switch0	TCP	
0.019	Switch0	Server0	TCP	
0.998	--	Switch0	STP	

Рис. 3.14, лист 2

Проведем анализ прохождения запросов (requests flow) посредством мониторинга продвижения **ключевых пакетов**, направленных компьютером **PC0** по компьютерной сети.

В ходе анализа для наглядности будем наблюдать прохождение пакета (пакетов) параллельно с изменением информации на устройстве (**PDU, Protocol Data Unit** – обобщенное название фрагмента данных **на разных уровнях модели OSI: кадр Ethernet, IP-пакет, UDP-дейтаграмма, TCP-сегмент и т. д.**).

Отсчет будем осуществлять по времени прохождения пакетов (**Time, sec**).

000. После нажатия кнопки **Go** сформировался **пакет** – запрос на **DNS-сервер** (рис. 3.15).

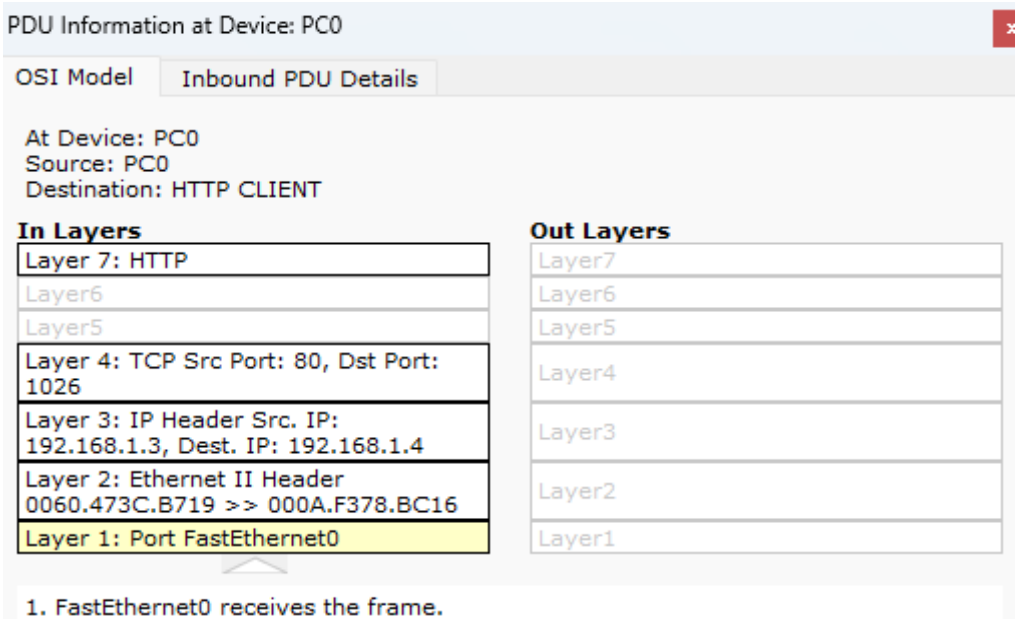

Рис. 3.15. Состояние пакета на время 000 с

001. Идет рассылка. Перемещение пакета происходит при помощи технологии FastEthernet. Для получения IP-адреса, соответствующего указанному адресу сайта **ipe-bsuir.by**, браузер отправляет пакет (запрос) **DNS-серверу** при помощи протокола **UDP**. Запрос проходит **Switch0** (коммутатор 2-го уровня) для последующего перенаправления на **IP-адрес 192.168.1.2** DNS-сервера (рис. 3.16).

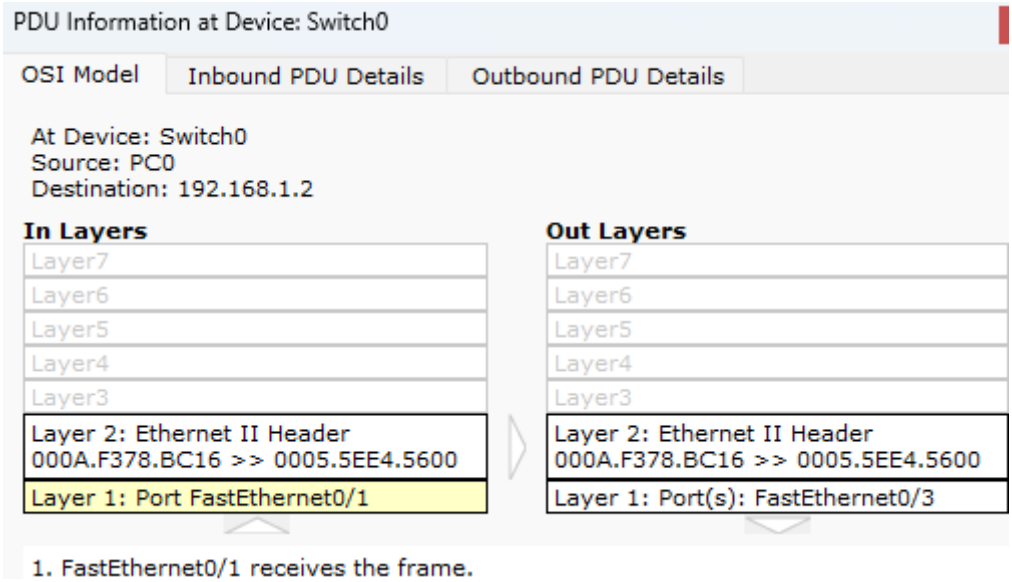

Рис. 3.16. Состояние продвижения пакета на время 001 с

002. DNS-сервер получает пакет, в котором адресом назначения является его собственный (рис. 3.17).

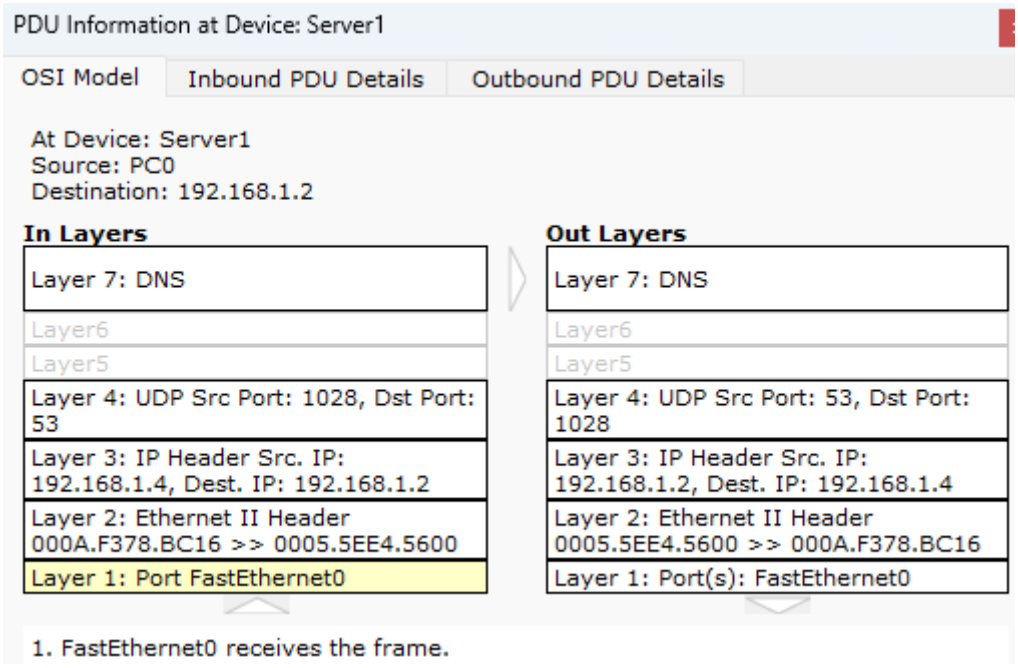

Рис. 3.17. Состояние продвижения пакета на время 002 с

003. Сервер формирует ответ, меняет местами адрес назначения и адрес источника и отправляет пакет с запрашиваемой информацией на **Switch0** для **PC0** (рис. 3.18).

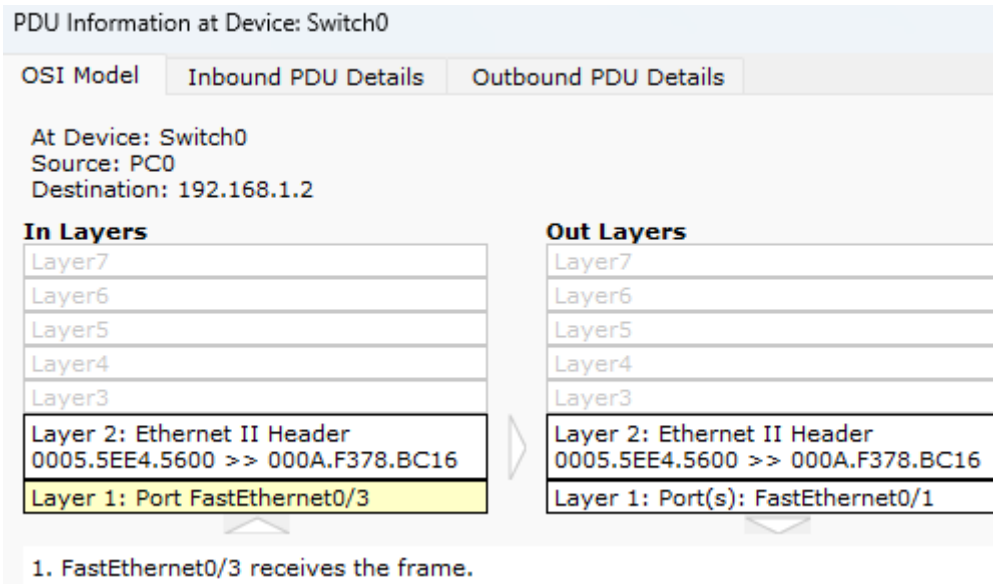

Рис. 3.18. Состояние продвижения пакета на время 003 с

004. Получив ответ от DNS-сервера, компьютер формирует три пакета, один из которых связан с информацией от **DNS-сервера**, которая перемещается при помощи протокола **UDP**, а два других образованы для установления соединения с соответствующим веб-сервером, а, следовательно, для продвижения информации с помощью протокола **TCP** через **Switch0 на IP-адрес 192.168.1.3 веб-сервера** (рис. 3.19).

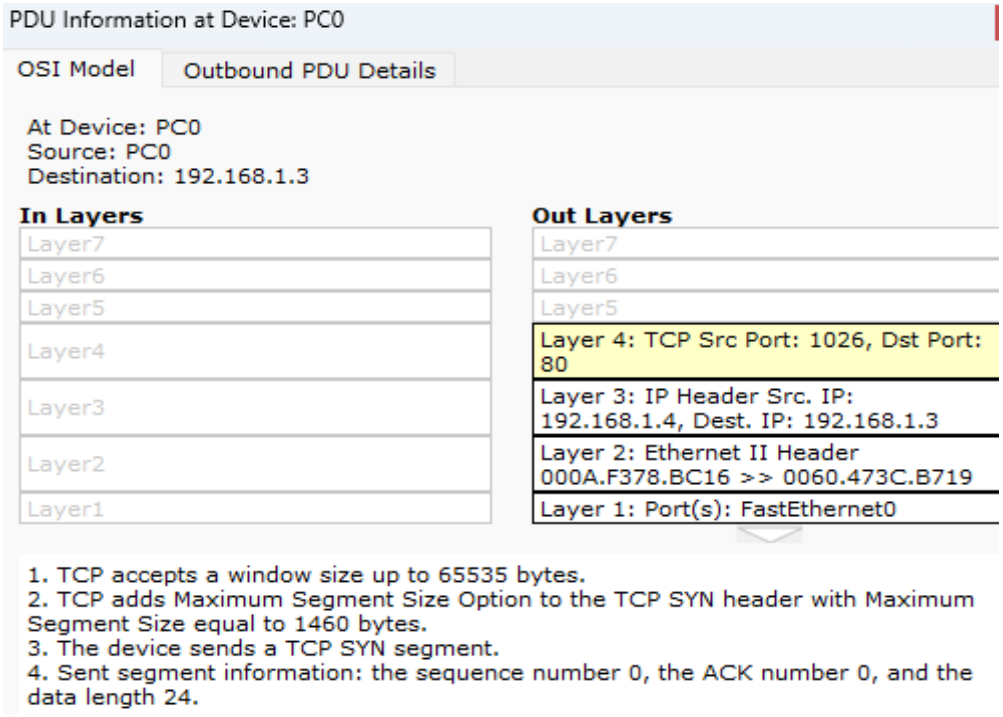

Рис. 3.19. Состояние направления сегмента TCP SYN на время 004 с

006. Пакет прибывает на **веб-сервер**. Веб-сервер принимает HTTP-запрос от браузера, контролирует его правильность и находит запрашиваемый ресурс на основе домена указанного сайта (рис. 3.20).

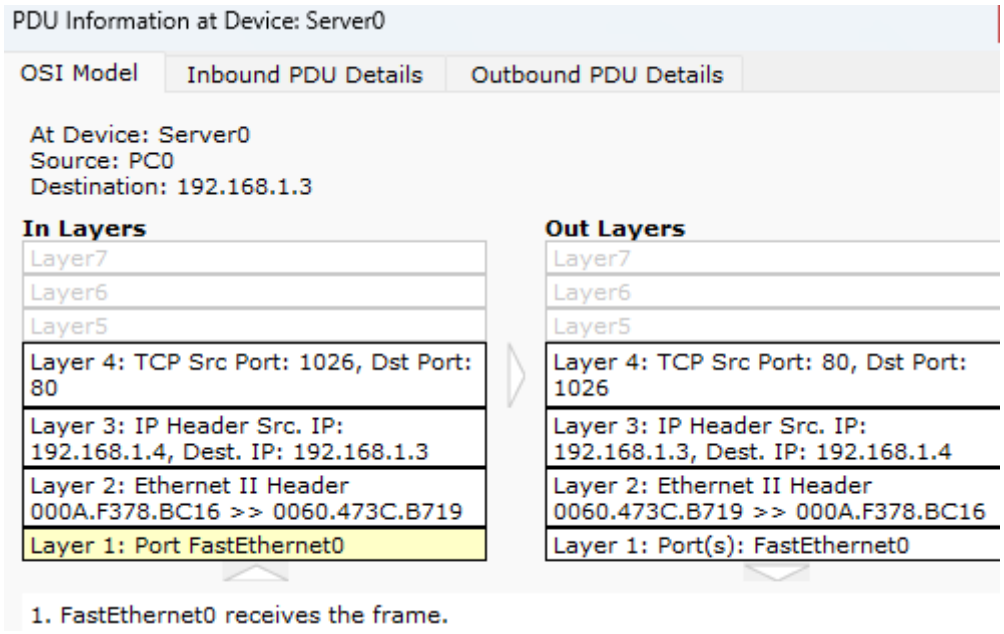

Рис. 3.20. Состояние прибытия пакета на веб-сервер на время 006 с

008. Информация с подтверждением от **веб-сервера** приходит на **PC0**. Кроме того, браузер формирует HTTP-запрос, который сообщает **веб-серверу**, какую информацию необходимо предоставить (рис. 3.21).

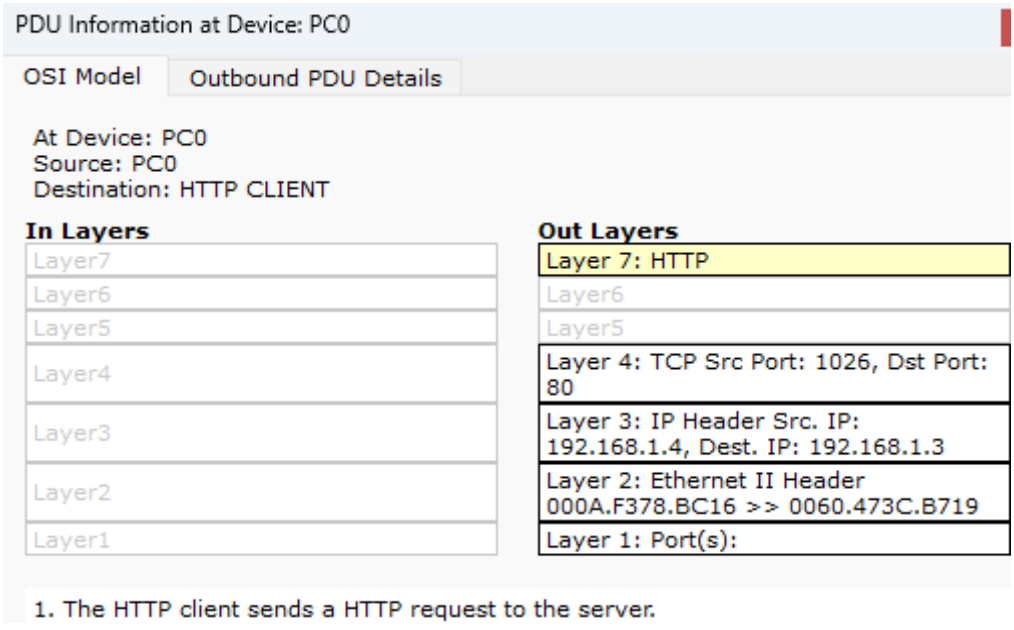

Рис. 3.21. Состояние HTTP-запроса на веб-сервер из PC0 на время 008 с

011. HTTP-запроса от **PC0** прибыл на **веб-сервер**. Веб-сервер формирует и отправляет браузеру HTTP-ответ, который включает статус ответа, заголовки ответа и тело ответа, содержащее информацию о запрошенном ресурсе (рис. 3.22).

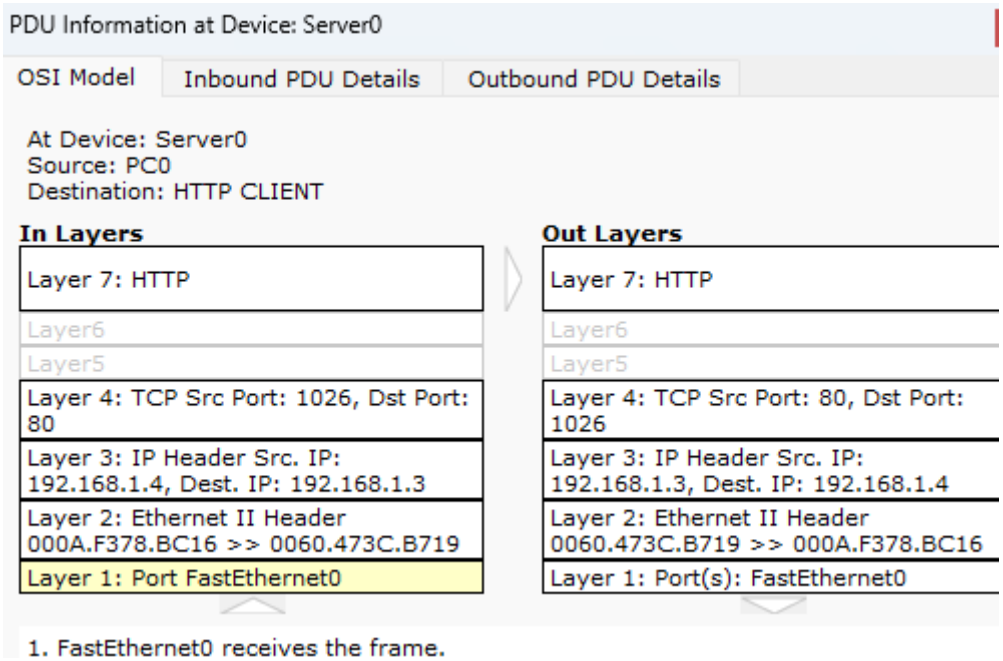

Рис. 3.22. Состояние прибытия HTTP-запроса на веб-сервер на время 011 с

013. Браузер компьютера получает HTTP-ответ от веб-сервера и обрабатывает его. Он проверяет статус ответа, его заголовки и формат содержащейся информации (рис. 3.23).

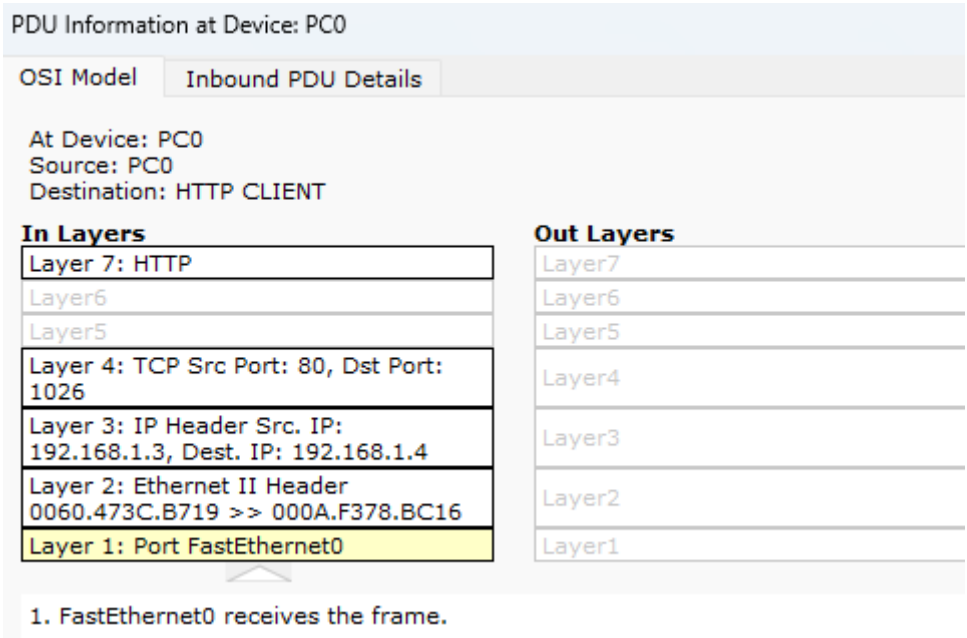

Рис. 3.23. Состояние прибытия HTTP-запроса на PC0 на время 013 с

Если ответ **веб-сервера** содержит требуемую веб-страницу, то браузер компьютера рассматривает ее по **HTML-коду**, загружает все связанные с ней ресурсы, включая изображения, стили и скрипты, и показывает страницу, фиксируя ее содержимое.

015. После завершения просмотра запрашиваемой информации от компьютера на веб-сервер прибывают сведения о получении информации в ответ на **HTTP-запрос** (рис. 3.24).

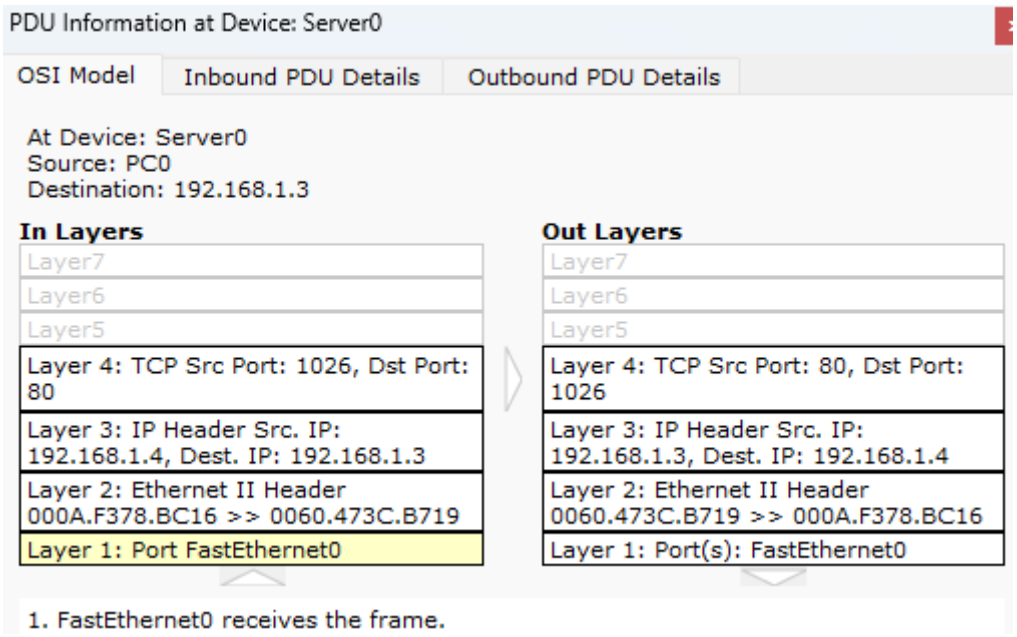

Рис. 3.24. Подтверждение получения HTTP-запроса от PC0 веб-сервером

Далее идет последовательная взаимная информация от компьютера и веб-сервера о завершении соединения между ними и окончании сеанса (рис. 3.25).

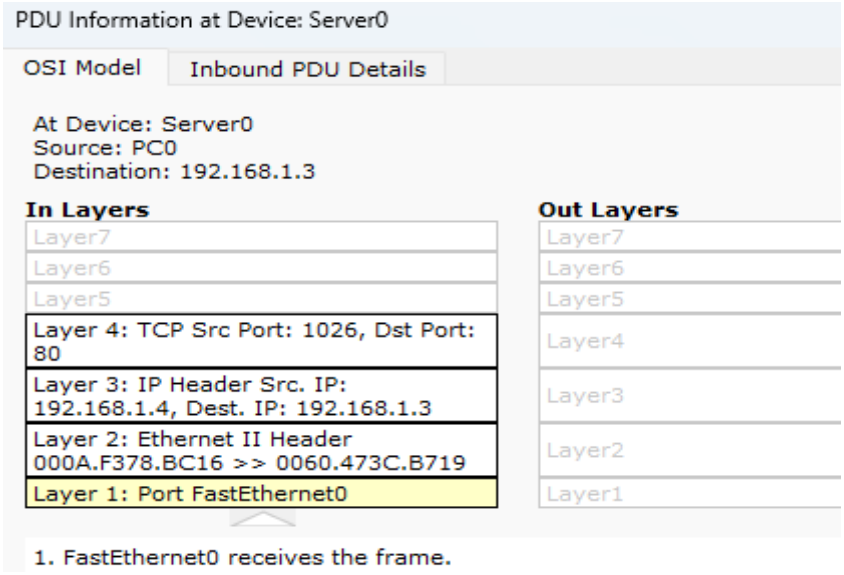

Рис. 3.25. Состояние получения веб-сервером информации о завершении соединения на время 019 с

0998. В рамках реализации протокола **STP** начинается процедура направления конфигурационного **BPDU** – фрейма (кадра) протокола управления сетевыми мостами. **Swictch0** инкапсулирует **BPDU** в кадр **Ethernet** и передает его в порты доступа (рис. 3.26).

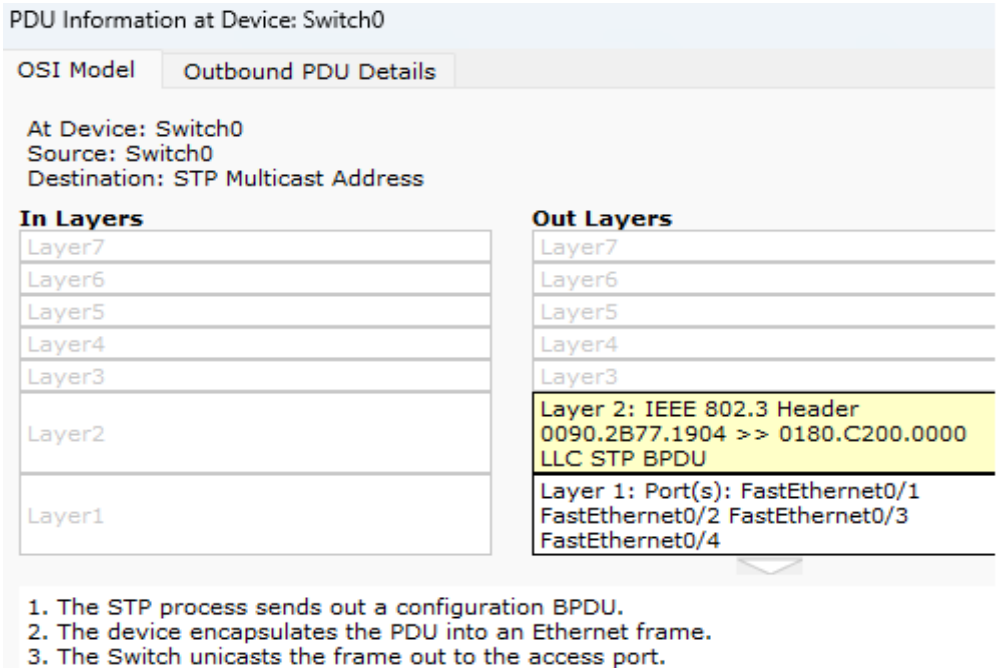

Рис. 3.26. Начало реализации протокола STP

Таким образом, на данном примере мы имели возможность углубить знания о принципах работы протоколов транспортного уровня **TCP** и **UDP**, их взаимодействии с сетевыми службами **DNS** и **DHCP**, организации их совместного функционирования, а также основных особенностях организации работы отдельного устройства с **веб-сервером**.

3.3. Практическое задание

3.3.1. Изучить сведения, изложенные в теоретических материалах данного практического занятия и, при необходимости, в дополнительных материалах.

3.3.2. С целью углубления знаний о принципах работы протоколов транспортного уровня TCP и UDP необходимо выполнить следующие действия.

3.3.2.1. Построить для всех вариантов одинаковую структурную схему локальной сети, состоящую из маршрутизатора, DNS-сервера и веб-сервера.

Количество и типы компьютеров, IP-адреса сети и наименования сайтов для направления на них запросов использовать согласно указанному преподавателем номеру варианта задания, приведенному в табл. 3.2.

3.3.2.2. Провести конфигурирование маршрутизатора с настроенным DHCP-сервером.

3.3.2.3. Настроить DNS-сервер.

3.3.2.4. Настроить веб-сервер.

3.3.2.5. Провести настройку хостов.

3.3.2.6. Осуществить контроль функционирования локальной сети.

3.3.3. Отчет о проделанной работе представить преподавателю непосредственно на компьютере или в виде сканирования ключевых эпизодов выполнения работы по п. 3.3.2.

3.3.4. Ответить на контрольные и дополнительные вопросы.

Варианты заданий

Таблица 3.2

Перечень вариантов заданий

Вариант	IP-адреса и количество сегментов сети	Вариант	IP-адреса и количество сегментов сети
1	1. ПК и Laptop. 2. IP-адрес сети – 192.168.10.0. 3. Сайт: ipe-new1.by	7	1. Три ПК. 2. IP-адрес сети – 192.168.70.0. 3. Сайт: ipe-new4.by
2	1. Два ПК. 2. IP-адрес сети – 192.168.20.0. 3. Сайт: ipe-new7.by	8	1. ПК и два Laptop. 2. IP-адрес сети – 192.168.80.0. 3. Сайт: ipe-new10.by
3	1. Два ПК и Laptop. 2. IP-адрес сети – 192.168.30.0. 3. Сайт: ipe-new2.by	9	1. ПК и Laptop. 2. IP-адрес сети – 192.168.90.0. 3. Сайт: ipe-new5.by
4	1. ПК и Laptop. 2. IP-адрес сети – 192.168.40.0. 3. Сайт: ipe-new8.by	10	1. Два ПК и Laptop. 2. IP-адрес сети – 192.168.100.0. 3. Сайт: ipe-new11.by
5	1. Два ПК и Laptop. 2. IP-адрес сети – 192.168.50.0. 3. Сайт: ipe-new3.by	11	1. ПК и два Laptop. 2. IP-адрес сети – 192.168.110.0. 3. Сайт: ipe-new6.by
6	1. Три ПК. 2. IP-адрес сети – 192.168.60.0. 3. Сайт: ipe-new9.by	12	1. Два ПК и Laptop. 2. IP-адрес сети – 192.168.120.0. 3. Сайт: ipe-new12.by

Контрольные вопросы

1. Какие основные особенности протокола TCP?
2. Назовите основные особенности протокола UDP.
3. В чем особенность пакетов в протоколе TCP?
4. В чем особенность пакетов в протоколе UDP?
5. Назовите основные составляющие заголовка сегмента TCP.
6. Из каких уровней состоит стек протоколов TCP/IP?
7. Какие основные сетевые службы привлекаются для функционирования стека протоколов TCP/IP?
8. Какие управляющие биты используются в заголовке протокола TCP?
9. Как организовывается сеанс протокола TCP?
10. Перечислите особенности применения протокола DHCP.
11. Каков порядок прохождения запросов при работе стека протоколов TCP/IP?
12. В чем заключаются основные отличия между протоколами TCP и UDP?
13. Для чего предназначен сокет?
14. Перечислите основные термины, связанные с состояниями сеанса протокола TCP.

15. Что такое указатель важности?
16. Из чего состоит модель OSI?
17. Каково основное назначение транспортного уровня модели OSI?
18. Что такое порт для устройства сети?
19. Для чего применяется флаг ACK?
20. Для чего и во взаимодействии с каким протоколом транспортного уровня применяется DNS?

Практическое занятие № 4
«Технология преобразования сетевых адресов (NAT) в компьютерных сетях»

Цель занятия: изучить технологию преобразования (трансляции) сетевых адресов **NAT** и порядок ее настройки.

4.1. Краткие теоретические сведения

4.1.1. Необходимость внедрения технологии NAT в современных компьютерных сетях

Современное общество невозможно представить без глобальной сети Интернет, которая стала неотъемлемой частью повседневной жизни. Она используется в самых разных сферах: от управления бизнес-процессами до организации образовательных программ и социального взаимодействия.

С развитием технологий, таких как мобильная связь стандартов 3G, 4G и 5G, а также благодаря снижению стоимости интернет-услуг, количество подключенных устройств и пользователей сети продолжает стремительно расти. Однако этот прогресс сопровождается рядом технических вызовов, одним из которых является **ограниченность адресного пространства протокола IPv4**.

Протокол IPv4, на котором долгое время базировалась работа Интернета, предусматривает использование примерно 4,3 миллиарда уникальных IP-адресов. Этого количества, казавшегося достаточным на заре развития сети, сегодня уже не хватает для удовлетворения потребностей растущего числа устройств, подключаемых к Интернету. В связи с этим возникла необходимость в поиске решений, которые позволили бы эффективно использовать имеющиеся ресурсы и обеспечить дальнейшее развитие сетевой инфраструктуры.

Для преодоления дефицита IP-адресов было предложено несколько подходов. Один из них предполагает более строгий учет и распределение адресного пространства. Это включает в себя выявление и повторное использование неактивных или заброшенных IP-адресов, что позволяет временно снизить нагрузку на существующую инфраструктуру. Однако такой подход не решает проблему в долгосрочной перспективе, так как количество доступных адресов остается ограниченным.

Другим решением является внедрение протокола IPv6, который был разработан как преемник IPv4. Этот протокол предлагает значительно большее адресное пространство – порядка 340 секстиллионов уникальных адресов, что практически исключает возможность их исчерпания в обозримом будущем. Кроме того, IPv6 обладает улучшенными механизмами безопасности, такими как встроенное шифрование данных и проверка целостности пакетов, что делает его более устойчивым к современным киберугрозам. Однако переход на IPv6 сопряжен с рядом трудностей, включая несовместимость с IPv4, необходимость

модернизации сетевого оборудования и значительные финансовые затраты. Эти факторы замедляют процесс внедрения новой версии протокола.

В таких условиях особую актуальность приобретает технология трансляции сетевых адресов (NAT). NAT позволяет нескольким устройствам в локальной сети использовать один публичный IP-адрес для выхода в Интернет, что значительно снижает потребность в уникальных адресах. Это решение стало временным, но эффективным способом решения проблемы нехватки IPv4-адресов, особенно в условиях, когда полный переход на IPv6 остается сложной задачей. Технология NAT не только экономит адресное пространство, но и обеспечивает дополнительный уровень безопасности, скрывая внутреннюю структуру сети от внешних угроз.

Таким образом, применение технологии NAT стало важным шагом в развитии компьютерных сетей, позволяющим поддерживать их стабильную работу в условиях ограниченных ресурсов и постепенного перехода на новые стандарты.

4.1.2. Общие принципы функционирования NAT

Основная задача NAT заключается в преобразовании внутренних (частных) IP-адресов устройств локальной сети в один или несколько внешних (публичных) IP-адресов, которые могут быть маршрутизированы в глобальной сети (рис. 4.1).

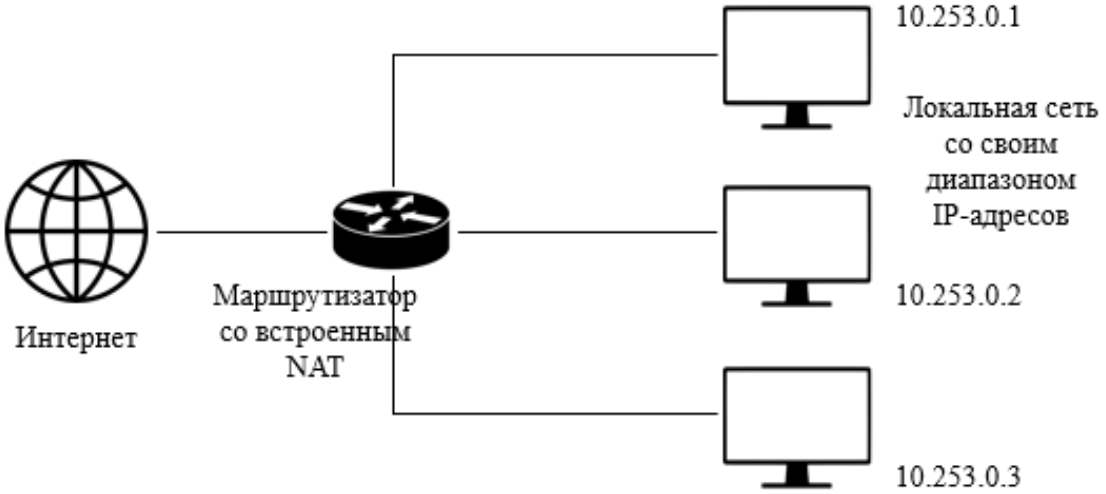

Рис. 4.1. Общий вид соединения сетей с использованием технологии NAT

Этот процесс осуществляется с помощью NAT-сервера, который может быть реализован как программное обеспечение на выделенном компьютере или как функция аппаратного маршрутизатора. NAT-сервер действует как посредник между внутренней сетью и внешним миром, обеспечивая прозрачное взаимодействие.

Когда устройство внутри локальной сети инициирует запрос к внешнему серверу, NAT-сервер фиксирует исходный IP-адрес и порт устройства в специальной таблице. Затем он заменяет эти данные на внешний IP-адрес и порт,

которые используются для передачи пакетов во внешнюю сеть. Таким образом, для внешнего сервера все запросы из локальной сети выглядят так, как будто они исходят от одного устройства – NAT-сервера.

Выделяют четыре этапа работы NAT:

1. Инициализация запроса клиентом. Приложение на устройстве внутри локальной сети формирует запрос, указывая в пакете свой внутренний IP-адрес и порт в качестве источника, а также внешний IP-адрес и порт назначения. Пакет направляется на NAT-сервер, который выступает в роли шлюза по умолчанию.

2. Преобразование адресов на NAT-сервере. NAT-сервер принимает пакет, сохраняет исходные данные (IP-адрес и порт клиента) в своей таблице и заменяет их на внешний IP-адрес и порт. После этого пакет отправляется во внешнюю сеть.

3. Обработка запроса внешним сервером. Внешний сервер получает пакет и обрабатывает его, считая, что запрос поступил от NAT-сервера. Ответный пакет отправляется на внешний IP-адрес и порт NAT-сервера.

4. Обратное преобразование на NAT-сервере. NAT-сервер получает ответный пакет, находит в своей таблице соответствующий внутренний IP-адрес и порт клиента, заменяет адрес назначения на внутренний и передает пакет в локальную сеть.

К основным преимуществам NAT относятся следующие:

- **экономия IP-адресов.** NAT позволяет множеству устройств использовать один публичный IP-адрес, что значительно снижает потребность в уникальных адресах IPv4;
- **повышение безопасности.** Устройства внутри локальной сети остаются скрытыми от внешних угроз, так как их внутренние IP-адреса не видны извне. NAT также блокирует несанкционированные входящие соединения, если они не были инициированы изнутри сети;
- **прозрачность для пользователей.** NAT работает автоматически, не требуя дополнительных настроек со стороны пользователей или приложений.

Ограничения NAT

Несмотря на свои преимущества, NAT имеет некоторые ограничения. Например, он может создавать сложности при работе с приложениями, которые требуют прямого доступа к устройствам внутри локальной сети (например, видеоконференции или онлайн-игры). Кроме того, NAT не является полноценной заменой брандмауэра, хотя и обеспечивает базовый уровень защиты.

4.1.3. Терминология и основные типы технологии NAT

Технология трансляции сетевых адресов предполагает использование различных терминов и типов преобразования, которые зависят от направления трафика (входящий или исходящий) и расположения устройства (внутри локальной сети или в глобальной сети Интернет). Для понимания работы NAT

важно разобраться в ключевых терминах, которые описывают адресацию в контексте этой технологии.

Рассмотрим более подробно **терминологию NAT**.

Внутренний локальный адрес (Inside Local Address) – это IP-адрес устройства, используемый внутри локальной сети. Он не является уникальным в глобальном масштабе и предназначен для внутреннего использования. Например, это может быть адрес из диапазона 192.168.x.x.

Внутренний глобальный адрес (Inside Global Address) – это публичный IP-адрес, который заменяет внутренний локальный адрес при передаче данных во внешнюю сеть. Этот адрес виден из Интернета и используется для маршрутизации пакетов.

Внешний локальный адрес (Outside Local Address) – это адрес внешнего устройства (например, сервера), который виден из локальной сети. Он может отличаться от реального адреса устройства в глобальной сети.

Внешний глобальный адрес (Outside Global Address) – это реальный IP-адрес внешнего устройства, используемый в глобальной сети. Обычно он совпадает с внешним локальным адресом, но в некоторых конфигурациях NAT может быть изменен.

Эти термины всегда рассматриваются с точки зрения устройства, чей адрес транслируется. Например, для устройства внутри локальной сети его внутренний локальный адрес будет преобразован во внутренний глобальный при отправке данных во внешнюю сеть.

Технология NAT может быть реализована в различных формах, каждая из которых имеет свои особенности и области применения. Основные типы NAT представлены ниже.

Статический NAT (Static NAT) – этот тип предполагает прямое сопоставление одного внутреннего локального адреса с одним внешним глобальным адресом. Такое преобразование является постоянным и не изменяется со временем. Статический NAT часто используется для устройств, которым требуется постоянный доступ из внешней сети, например, для веб-серверов или почтовых серверов.

Динамический NAT (Dynamic NAT) – в этом случае внутренние локальные адреса динамически сопоставляются с пулом внешних глобальных адресов. Когда устройство инициирует соединение, ему назначается один из доступных адресов из пула. После завершения сеанса этот адрес освобождается и может быть использован другим устройством. Динамический NAT подходит для сетей, где количество активных соединений не превышает размер пула внешних адресов.

Перегруженный NAT (NAT Overload или PAT) – это наиболее распространенный тип NAT, также известный как трансляция адресов портов (PAT). В этом случае множество внутренних локальных адресов транслируются в один внешний глобальный адрес, но с использованием уникальных номеров портов для каждого соединения. Это позволяет эффективно использовать

ограниченное количество публичных IP-адресов, обеспечивая доступ в Интернет для большого числа устройств. PAT широко применяется в домашних и корпоративных сетях.

Инверсная маска (Wildcard Mask)

При настройке **технологии NAT** обязательно используется **Wildcard bits (mask)**. **Wildcard mask –** это **перевернутая маска**, или как ее еще называют – **инверсная**. Эта маска показывает, какая часть (сколько бит) IP-адреса может меняться. Она может применяться при объявлении сетей в протоколах маршрутизации, таких как **IGRP, EIGRP, OSPF, и в списках доступа**. Принцип работы маски тоже такой же, как у обычной маски, за исключением того, что вместо единиц ставятся нули, а вместо нулей единицы.

Пример. Задан адрес с инверсной маской – 192.168.2.3 0.0.1.255 – это соответствует маске 00000000 00000000 00000001 11111111.

Следовательно, на соответствие шаблону будет проверяться только 23 левых бита рассматриваемого адреса.

4.2. Порядок и основные правила выполнения заданий

Для изучения применения NAT в компьютерных сетях будем использовать **симулятор Cisco Packet Tracer**.

Порядок и основные правила выполнения заданий рассмотрим в ходе выполнения конкретного примера – соединения небольшого офиса с сетью провайдера при помощи **технологии NAT**.

4.2.1. Построение структурных схем сетей

Запустим **программу Cisco Packet Tracer**. В области **«Логическое пространство»** построим две структурные схемы сетей (рис. 4.2):

1. Для офисной сети: три компьютера (PC-PT), сервер (Server-PT), коммутатор (2950-24) и маршрутизатор (1841).
2. Для сети провайдера: сервер (Server-PT) и маршрутизатор (1841).

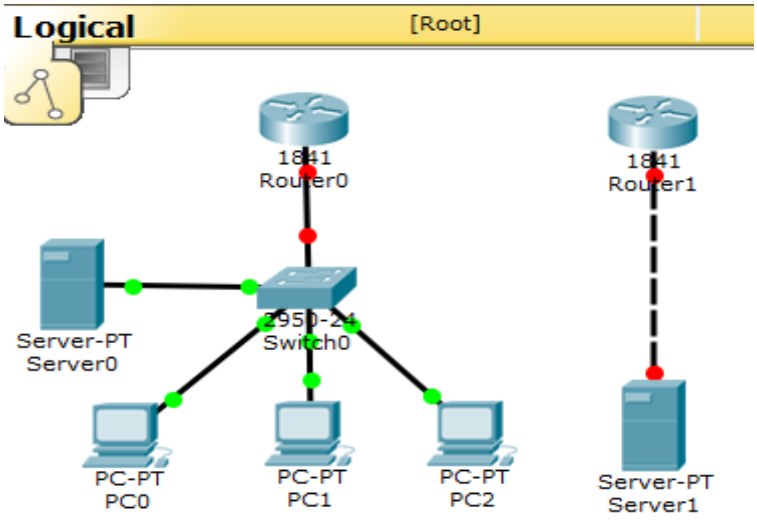

Рис. 4.2. Структурные схемы

4.2.2. Настройка коммутатора

Создадим два сегмента для сети офиса: **VLAN2** для сервера (**Server0**) и **VLAN3** для компьютеров (**PC0, PC1 и PC2**).

Для решения этой задачи воспользуется **Cisco IOS Command Line Interface** коммутатора. С этой целью нажатием левой кнопки мыши открываем таблицу **Switch0**, на которой переходим на вкладку **CLI**.

Далее для настройки коммутатора выполняем ряд последовательных действий путем использования стандартных команд:

Switch>**enable**

Switch#**configure terminal**

Switch(config)#

Вначале создадим сегмент сети **VLAN2 для сервера** и присвоим ему имя, например VLAN2. С этой целью выполняем следующие действия:

Switch(config)#**vlan2**

Switch(config-vlan)#**name VLAN2**

Switch(config-vlan)#**exit**

Switch(config)#

Подобным же образом создаем второй сегмент для компьютеров и присваиваем ему имя **VLAN3**. После присвоения имен возвращаемся в режим **Switch#** путем выхода из режима создания сегментов:

Switch(config)#**End**

Switch#

Теперь определяем наши компьютеры в нужный сегмент.

Компьютер PC0 подключен к интерфейсу Fa0/1, PC1 к Fa0/2, PC2 к Fa0/3 и Server0 к Fa0/4.

Настроим интерфейс **Fa0/1**, **Fa0/2** и **Fa0/3**. Для этого выполним действия, представленные на рис. 4.3.

Рис. 4.3. Определение компьютеров в сегмент VLAN3

Подобным образом настраиваем интерфейс **Fa0/4 в сегменте VLAN2**.

Теперь нам необходимо настроить **Trunk port (магистральный порт)**, который соединяет коммутатор с маршрутизатором.

Осуществим подключение **Trunk port**. С этой целью откроем вкладку **Config**, далее **FastEthernet 0/5**, подключим порт **Trunk** (не **Access**) и убедимся, что в следующей графе подключены все заданные **VLAN** – **VLAN1, VLAN2 и VLAN3** (рис. 4.4).

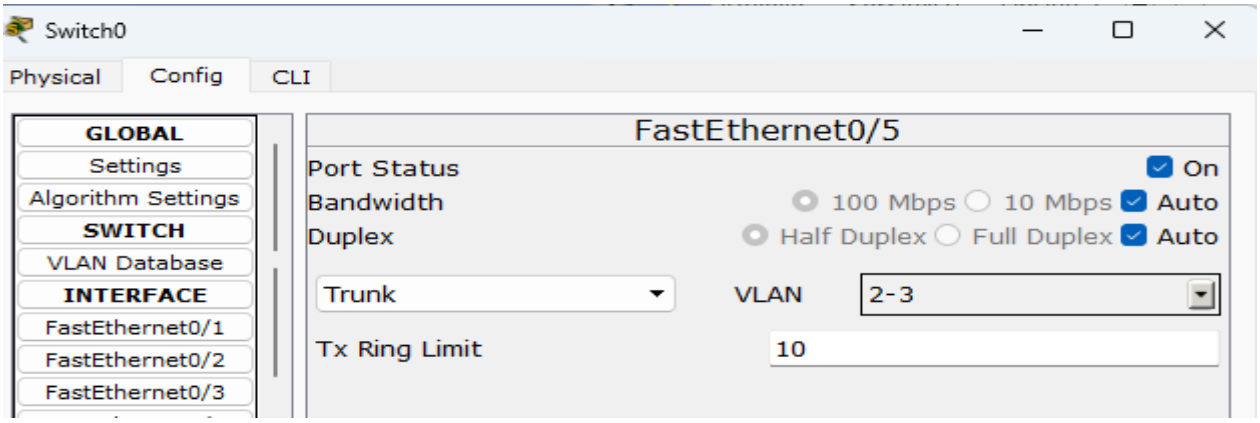

Рис. 4.4. Состояние включенного магистрального порта на коммутаторе

После завершения настройки всех портов выходим из режима конфигурации интерфейсов коммутатора.

4.2.3. Настройка маршрутизатора (Router0)

Если у коммутатора все порты по умолчанию «подняты», то у маршрутизаторов и межсетевых экранов порты по умолчанию находятся в режиме **down**. Поэтому необходимо в первую очередь поднять физический порт, в нашем случае **Fa0/0**.

С этой целью при помощи команд выполняем ряд последовательных действий во вкладке CLI маршрутизатора (рис. 4.5).

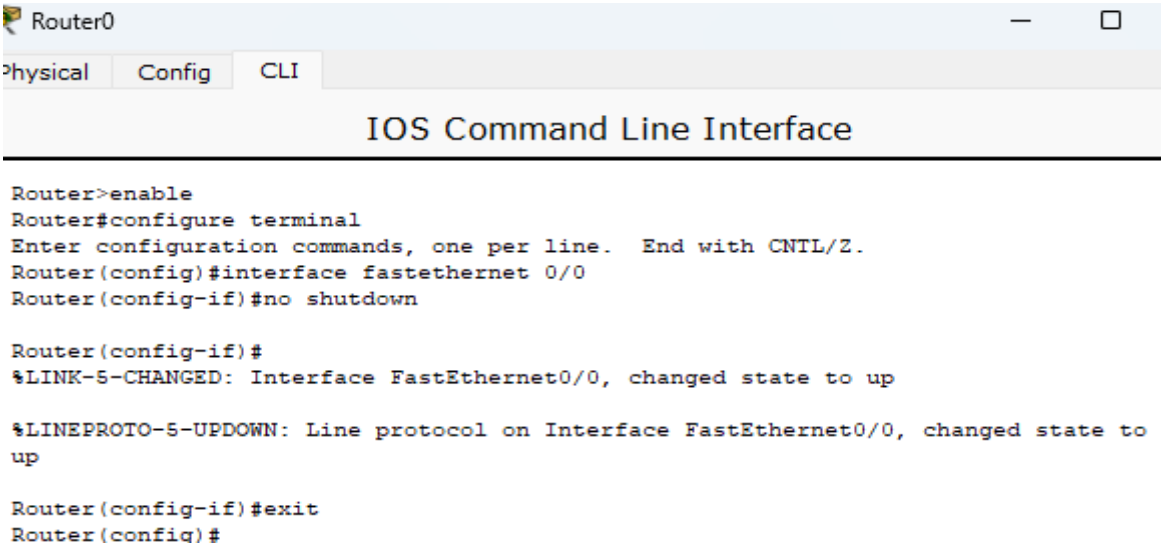

Рис. 4.5. Настройка магистрального порта маршрутизатора

Далее, так как маршрутизатор принимает два **VLAN**, возникает необходимость создания в нем двух **подынтерфейсов** (subinterfases). Каждому

подынтерфейсу, соответствующему определенному VLAN, необходимо сразу же назначить IP-адрес и маску сети. Сначала настроим подынтерфейс для VLAN2 (рис. 4.6).

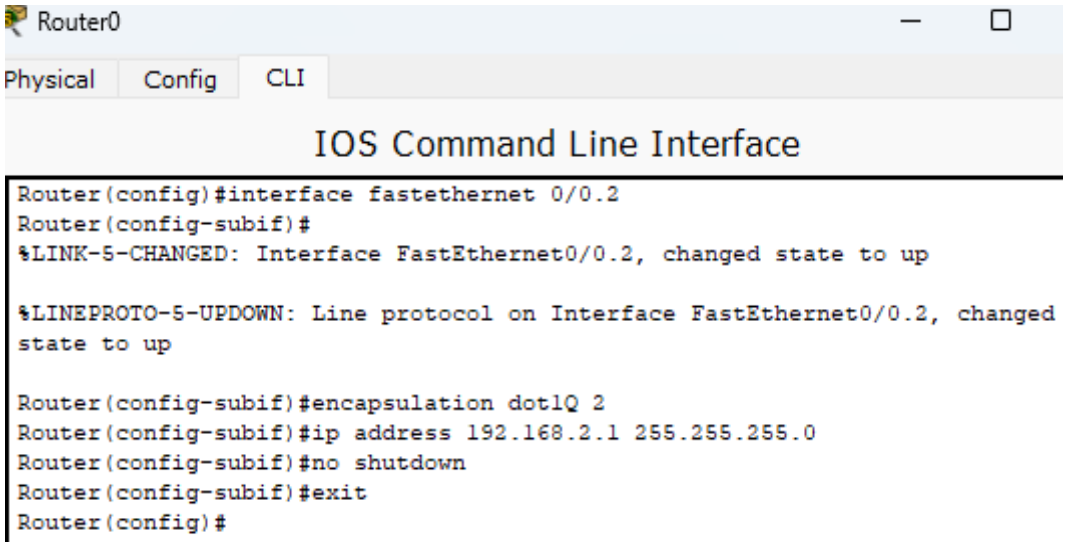

Рис. 4.6. Настройка подынтерфейса для VLAN2

Аналогично создадим подынтерфейс для **VLAN3** и определим IP-адрес и маску сети, где **IP-адресом** будет **192.168.3.1** и **маской – 255.255.255.0**. Следует учесть, что при использовании команды **encapsulation dot1Q** необходимо указать номер соответствующего VLAN, в данном случае – 3.

После создания необходимых подынтерфейсов заканчиваем работу в **IOS Command Line Interface** маршрутизатора:

Router(config)#**end**
Router#**write memory**
Router#

4.2.4. Выполнение конфигурации хостов и проведение контроля прохождения утилиты Ping в офисной сети

При назначении каждому хосту **IP-адреса** необходимо, кроме маски сети, еще прописывать и IP-адрес маршрутизатора (шлюза).

При помощи раздела **Desktop** и вкладки **Ip Configuration** хостов назначим каждому хосту IP-адреса, маски сети и IP-адреса шлюзов, как указано в табл. 4.1.

Таблица 4.1

Параметры IP-адресации хостов

Наименование хоста и VLAN	IP-адрес хоста	Маска сети	IP-адрес шлюза
Server0, VLAN2	192.168.2.2	255.255.255.0	192.168.2.1
ПК PC0, VLAN3	192.168.3.2	255.255.255.0	192.168.3.1
ПК PC1, VLAN3	192.168.3.3	255.255.255.0	192.168.3.1
ПК PC2, VLAN3	192.168.3.4	255.255.255.0	192.168.3.1

Нанесем на структурные схемы сети IP-адреса и наименования VLAN по образцу, указанному на рис. 4.7.

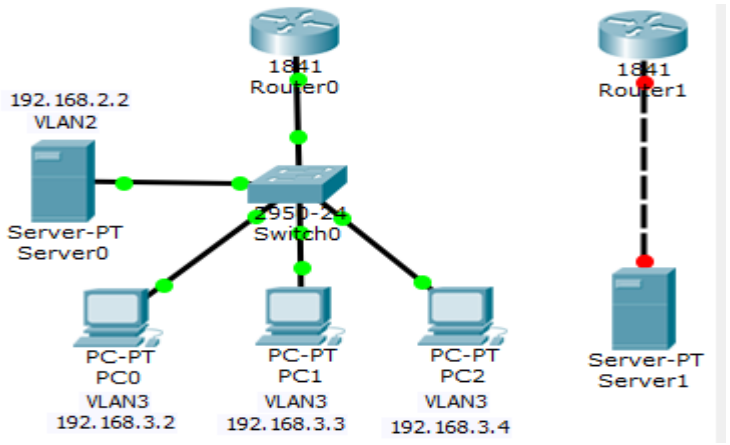

Рис. 4.7. Структурные схемы с обозначениями

После этого для проверки целостности и качества соединений в сети с каждого хоста направим на маршрутизатор и каждый хост утилиту **Ping**. Прохождение утилиты **Ping** завершено успешно, а следовательно, локальная сеть офиса настроена верно.

4.2.5. Соединение офисной сети с сетью провайдера, настройка соединения Router1 и Server1 сети провайдера

Теперь, допустим, в офисной локальной сети возникла потребность подключиться к Интернету. С этой целью необходимо обратиться к провайдеру, который осуществляет прокладку кабеля до офисной сети и выделяет для маршрутизатора **Router0** общедоступный (белый) **IP-адрес 213.234.10.2 с маской 255.255.255.252**.

Поскольку в **Cisco Packet Tracer** мы не можем подключиться к сети Интернет, мы симулируем сеть Интернет посредством **Router1** и **Server1**, у которых будут публичные белые адреса.

Определим для **Server1 IP-адрес 213.234.20.2 с маской 255.255.255.252** и проведем его настройку с учетом того, что сервер соединен с **Router1** через **интерфейс Fa0/0 с IP-адресом 213.234.20.1** и **маской 255.255.255.252** (рис. 4.8).

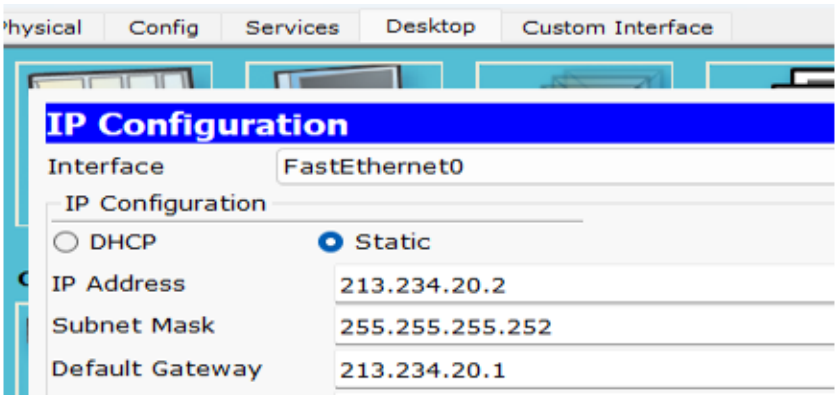

Рис. 4.8. Настройка Server1

Далее проверим наименование интерфейсов, соединяющих маршрутизаторы между собой. Так, на **Router1** интерфейс **Fa0/1**. Определим на этот интерфейс **Router1** белый **IP-адрес 213.234.10.1** с **маской 255.255.255.252**. Для присвоения IP-адреса проведем настройку **Router1** следующим образом (рис. 4.9).

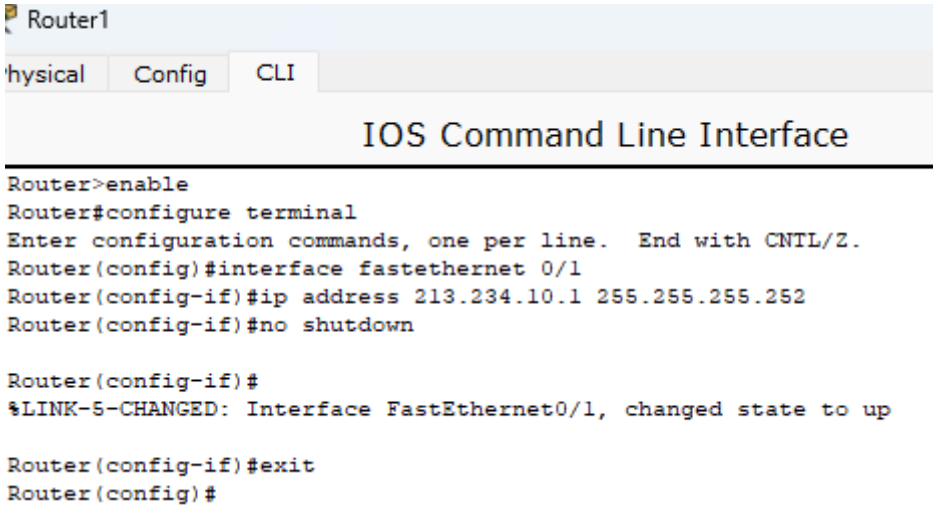

Рис. 4.9. Настройка интерфейса Router1 со стороны локальной сети

Для завершения настройки присвоим маршрутизатору **IP-адрес 213.234.20.1 с маской 255.255.255.252** для порта на **Server1** (рис. 4.10).

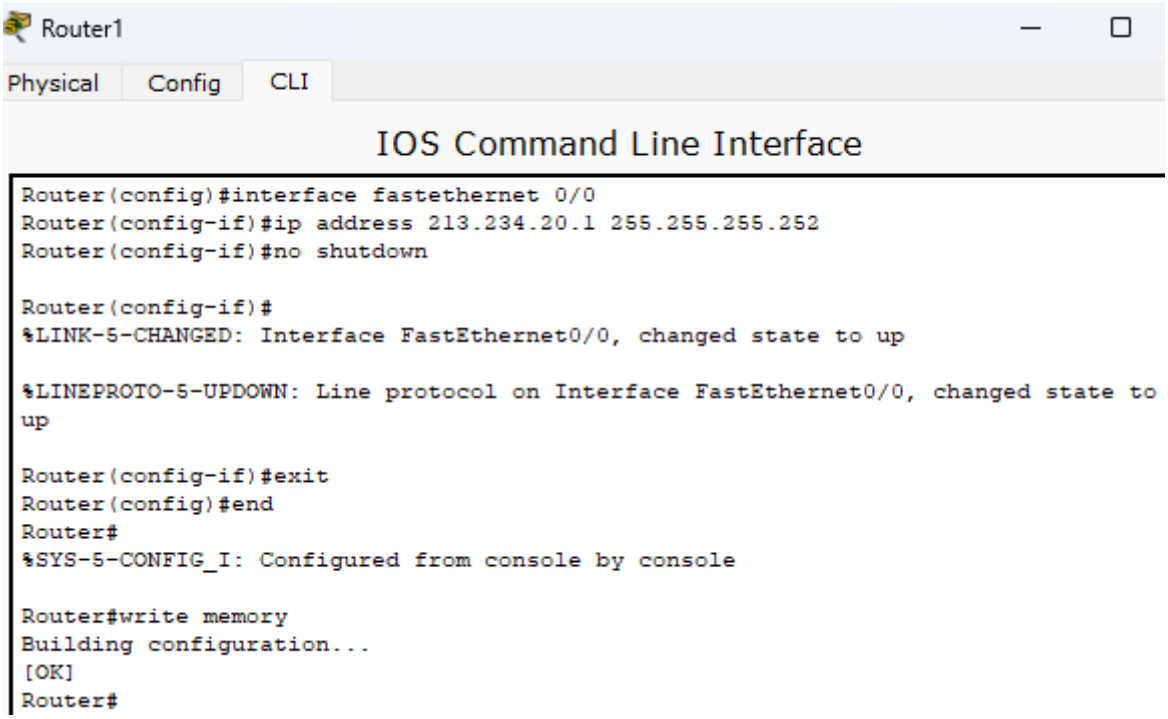

Рис. 4.10. Настройка интерфейса Router1 со стороны Server1

4.2.6. Настройка соединения Router0 и Router1

Для настройки офисного **Router0** пропишем **IP-адрес 213.234.10.2 с маской 255.255.255.252**, выданный для него провайдером. Кроме того, с целью

соединения двух маршрутизаторов (офисной сети и сети провайдера) создадим **шлюз по умолчанию** (рис. 4.11).

```
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastethernet 0/1
Router(config-if)#ip address 213.234.10.2 255.255.255.252
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastethernet 0/1
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up

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

Router(config-if)#exit
Router(config)#ip route 0.0.0.0 0.0.0.0 213.234.10.1
Router(config)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#write memory
```

Рис. 4.11. Настройка интерфейса Router0 со стороны Router1

Таким образом мы провели настройку соединения офисной сети и сети провайдера. Для лучшей наглядности нанесем на схему полезные IP-адреса всех устройств (рис. 4.12).

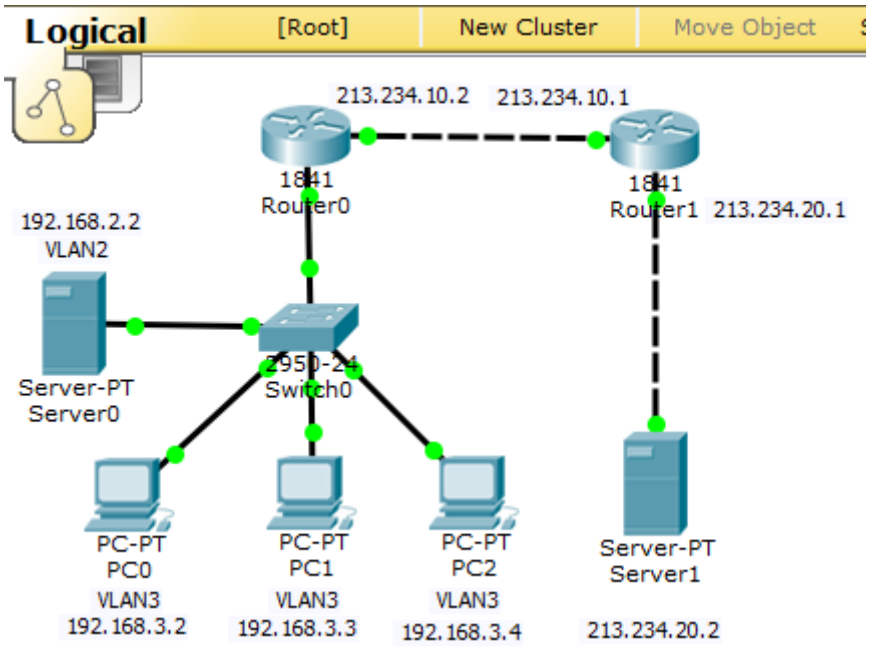

Рис. 4.12. Структурная схема с установленным соединением между сетями

Проверим правильность настройки соединения между маршрутизаторами **Router0** и **Router1** путем направления утилиты **Ping от маршрутизатора**

локальной сети до маршрутизатора провайдера. Ping прошел успешно, а следовательно, подтверждена правильность настройки указанного соединения (рис. 4.13).

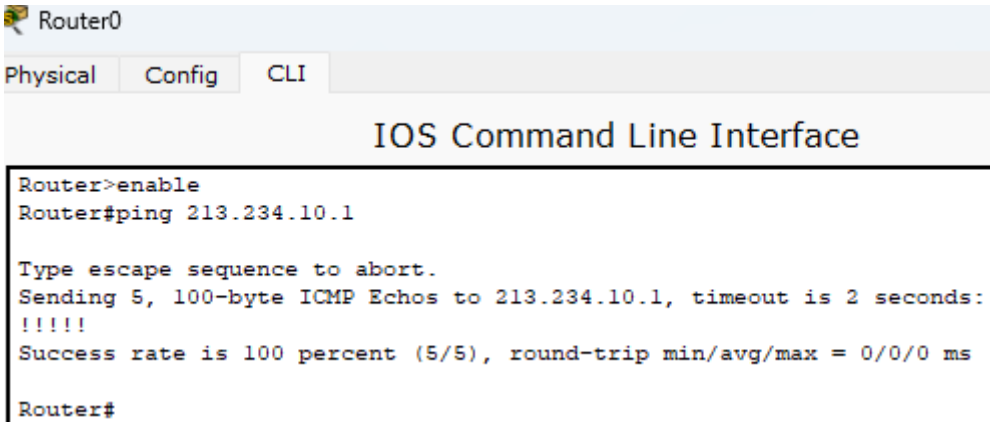

Рис. 4.13. Успешное прохождение утилиты Ping между маршрутизаторами

Направление утилиты **Ping** от одного из хостов офисной сети на IP-адрес маршрутизатора провайдера показало, что на этой стадии настройки еще нет соединения хостов локальной сети с внешними сетями. Связано это с тем, что хосты данной офисной сети имеют приватные адреса, что не позволяет без специальной настройки осуществлять выход в сеть Интернет и другие сети, имеющие общедоступные адреса.

4.2.7. Настройка технологии перегруженного NAT

Для решения вопроса обеспечения доступа от сети провайдера в локальную сеть с «серыми» адресами используют **технологию перегруженного NAT**. **NAT** имеет возможность перевода IPv4-адреса между «серыми» и «белыми» IPv4-адресами.

Учитывая изложенное, для доступа хостов офисной сети к тестовому серверу **Server1** применим **технологию NAT** в рассматриваемых сетях.

Для начала нужно определить, какой интерфейс **Router0** для **NAT** является внешним, а какой – внутренним.

В нашем случае интерфейс **Fa0/1** является внешним, настройка которого представлена на рис. 4.14.

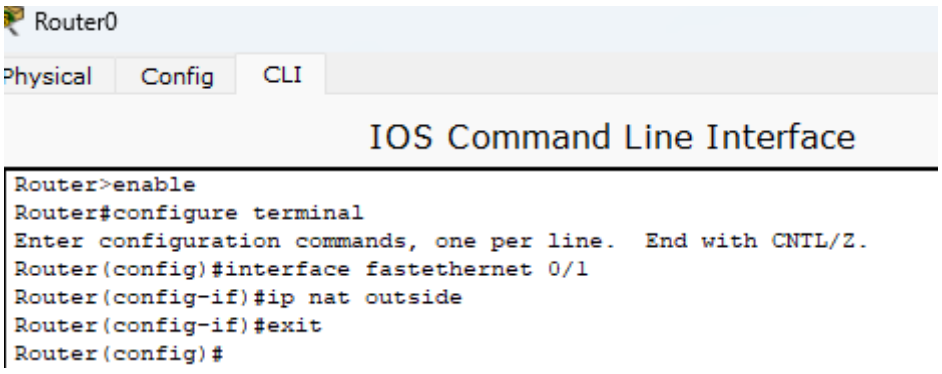

Рис. 4.14. Настройка внешнего интерфейса Router0

Для сервера офисной сети и хостов подынтерфейсы **Fa0/0.2** (сервера) и **Fa0/0.3** (компьютеров) являются внутренними (рис. 4.15).

Рис. 4.15. Настройка внутренних подынтерфейсов Router0

Кроме того, необходимо создать **списки контроля доступа** (ACL, **Access Control List**), представляющие набор текстовых выражений, которые в нашем случае установят трафик, подлежащий к выходу в сеть Интернет.

Для создания списков контроля допуска применяется команда:

ip access-list standard FOR-*название*

В этой команде название можно применить по своему усмотрению. В нашем случае за название примем **FOR-NATt**.

Также следует учесть, что если мы в продолжение строки введем знак вопроса, то на экране IOS **Command Line Interface** появится напоминание о необходимости введения **Wildcard bits** (Wildcard-маски).

Выполним указанные команды (рис. 4.16).

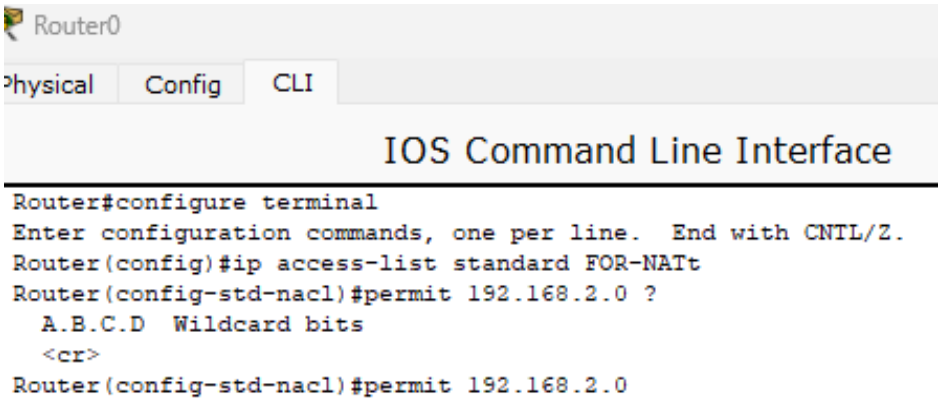

Рис. 4.16. Напоминание о необходимости введения Wildcard bits

Поэтому записываем 24-битную **Wildcard-маску** с точностью до наоборот от обычной маски, т. е. **0.0.0.255** (рис. 4.17).

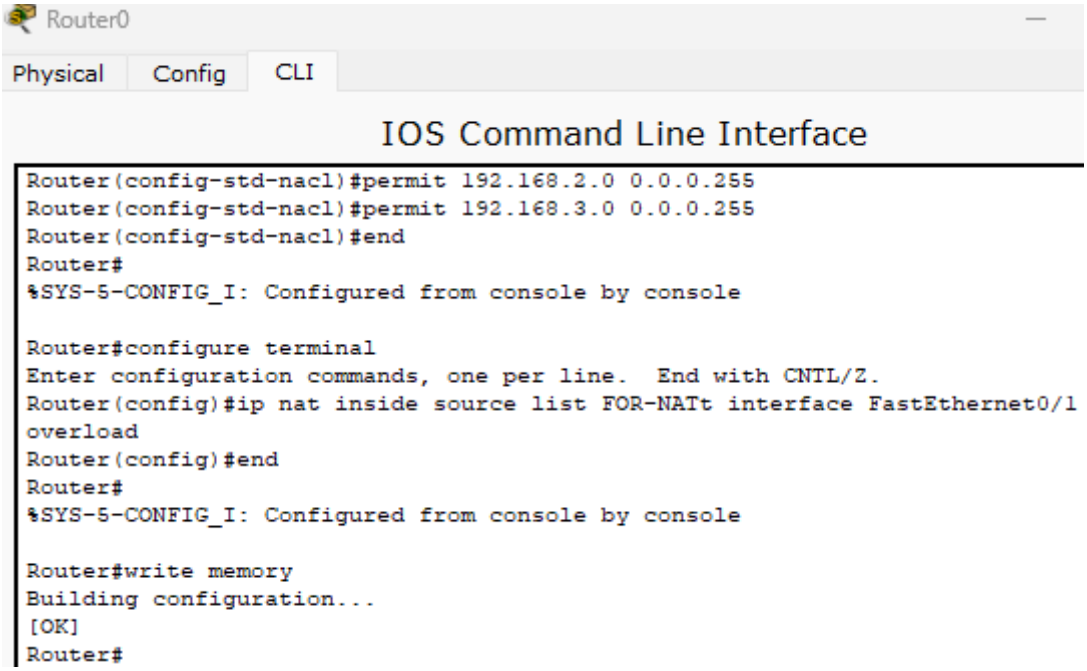

Рис. 4.17. Настройка списка контроля доступа внутренних подынтерфейсов Router0

Настройка использования технологии **NAT** для предоставления возможности преобразования приватных адресов офисной сети в общедоступный адрес провайдера завершена.

С целью проверки ее функционирования проверим прохождение утилиты **Ping** от компьютера **PC0** офисной сети до маршрутизатора провайдера (**Router1**), а также внешнего сервера (**Server1**). Утилита **Ping** прошла успешно.

Проконтролируем этапы прохождения **пакета ICMP** (**Internet Control Message Protocol –** протокол межсетевых управляющих сообщений), например от компьютера **PC0** офисной сети до общедоступного сервера (**Server1**) через коммутатор и роутер провайдера с использованием режима симуляции (рис. 4.18 и 4.19).

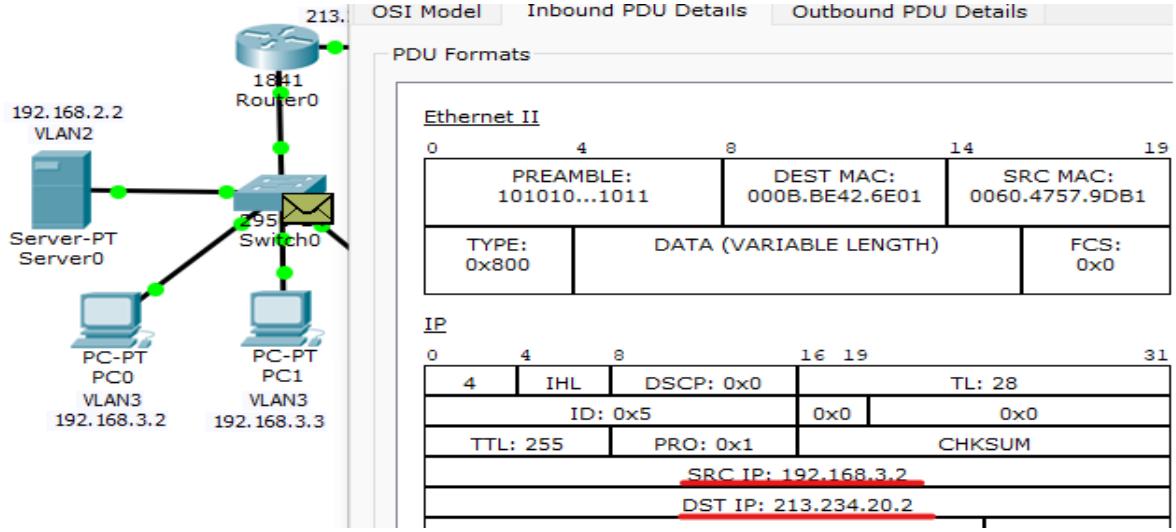

Рис. 4.18. Контроль прохождения пакета ICMP через Switch0

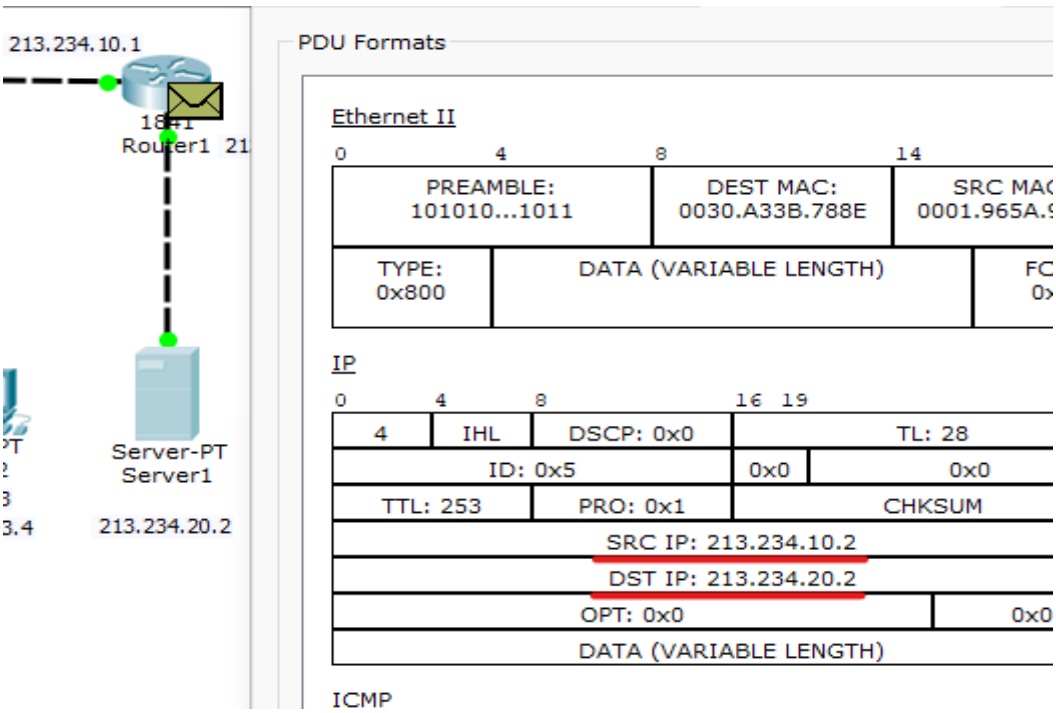

Рис. 4.19. Контроль прохождения пакета ICMP через Router1

Таким образом проведена настройка одного из видов технологий NAT – технологии перегруженного **NAT (PAT)**, которая позволяет осуществлять выход множеству хостов из офисной сети с частными адресами в сети с общественными адресами.

4.2.8. Настройка статического NAT

Рассмотрим далее возможности обращения из хоста сети с общественными адресами, например, к серверу офисной сети (**Server0**).

Для этого нам потребуется настройка статического **NAT**.

Проведем настройку маршрутизатора офисной сети (**Router0**) для предоставления доступа к серверу офисной сети (**Server0**) (рис. 4.20).

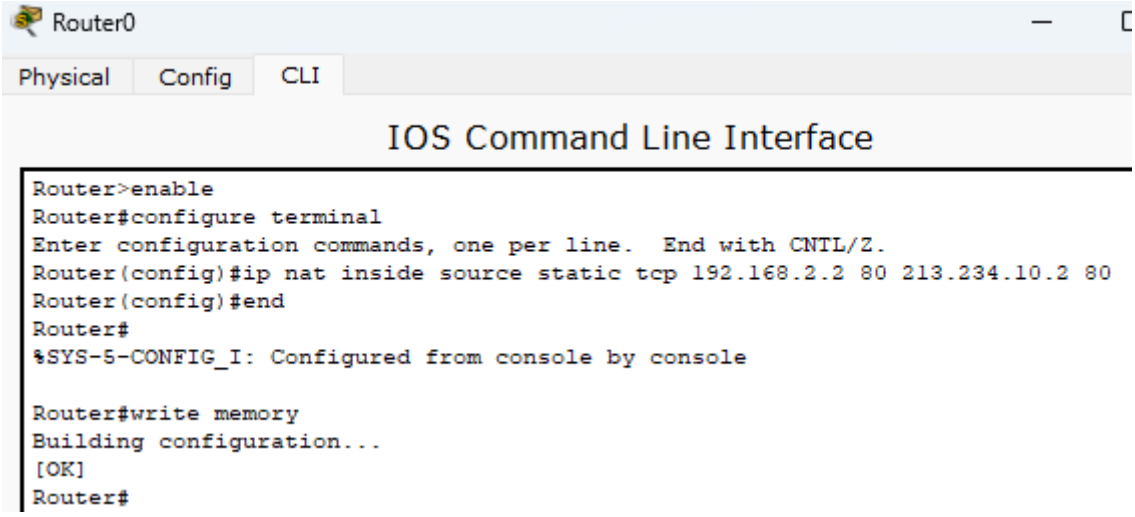

Рис. 4.20. Настройка доступа к серверу офисной сети из сети провайдера

Для проверки правильности настройки статического NAT в веб-браузере **Server1** создадим URL-запрос на сервер **Server0**. Выполнение запроса (ответ офисного сервера) подтверждено на рис. 4.21.

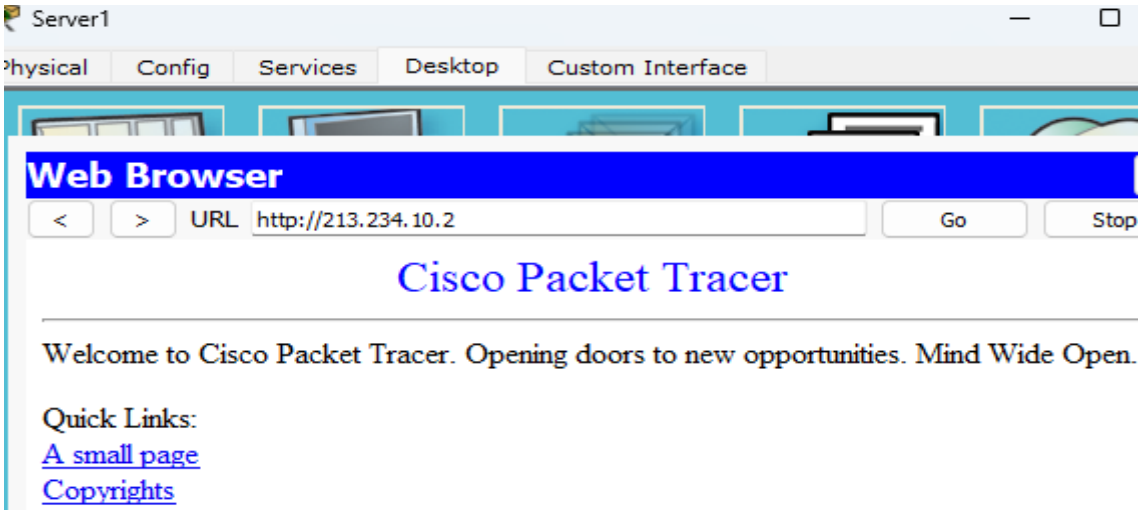

Рис. 4.21. Успешное прохождение ответа на URL-запрос с Server0

Таким образом, использование технологии NAT позволяет организовать устойчивое соединение между частными адресами офисной сети и общедоступными устройствами.

4.3. Практическое задание

4.3.1. Изучить сведения, изложенные в теоретических материалах данного практического занятия и, при необходимости, в дополнительных материалах.

4.3.2. С целью выполнения настройки взаимодействия составных частей сети и технологии **NAT** необходимо выполнить следующие действия.

4.3.2.1. Построить структурную схему офисной локальной сети, состоящую из сетевого оборудования и устройств, согласно указанному преподавателем номеру варианта задания, представленному в табл. 4.2.

4.3.2.2. Провести настройку коммутатора по количеству устройств и сегментов локальной сети в соответствии с вариантом.

4.3.2.3. Настроить порты и подынтерфейсы маршрутизатора локальной сети в соответствии с количеством сегментов, установленных вариантом задания.

4.3.2.4. Выполнить **IP-Configuration** всех хостов офисной сети и провести контроль прохождения утилиты **Ping** через маршрутизатор и остальные хосты.

4.3.2.5. Осуществить соединение офисной сети и сети провайдера и настройку соединения **Router** и **Server** сети провайдера.

4.3.2.6. Выполнить настройку соединения маршрутизаторов.

4.3.2.7. Последовательно провести настройку **PAT** и статического **NAT**.

4.3.2.8. Проконтролировать полноту и качество настройки соединения сетей.

4.3.3. Отчет о проделанной работе представить преподавателю непосредственно на компьютере или в виде сканирования ключевых эпизодов выполнения работы по п. 4.3.2.

4.3.4. Ответить на контрольные и дополнительные вопросы.

Варианты заданий

Таблица 4.2

Перечень вариантов заданий

Вариант	IP-адреса и количество сегментов сети	Вариант	IP-адреса и количество сегментов сети
1	1. ПК, два Laptop – первый сегмент. 2. ПК – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.10.0	8	1. ПК – первый сегмент. 2. Laptop, ПК – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.80.0
2	1. Два ПК, Laptop – первый сегмент. 2. ПК –второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.20.0	9	1. Три ПК – первый сегмент. 2. Laptop и сервер – второй сегмент. IP-адрес сети – 192.168.90.0
3	1. Два ПК и Laptop – первый сегмент. 2. Два ПК – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.30.0	10	1. Три Laptop – первый сегмент. 2. ПК – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.100.0
4	1. Два ПК – первый сегмент. 2. ПК, Laptop – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.40.0	11	1. Три ПК, Laptop – первый сегмент. 2. Сервер – второй сегмент. IP-адрес сети – 192.168.110.0
5	1. Три ПК – первый сегмент. 2. Laptop – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.50.0	12	1. Два ПК – первый сегмент. 2. ПК – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.120.0
6	1. ПК – первый сегмент. 2. Три Laptop – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.60.0	13	1. ПК – первый сегмент. 2. Три Laptop – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.130.0
7	1. Два ПК и Laptop – третий сегмент. 2. ПК – второй сегмент. 3. Сервер – первый сегмент. IP-адрес сети – 192.168.70.0	14	1. Три ПК – первый сегмент. 2. Laptop – второй сегмент. 3. Сервер – третий сегмент. IP-адрес сети – 192.168.140.0

Контрольные вопросы

1. Какие возможности предоставляет протокол IPv4?
2. Назовите причину медленного роста применения IPv6.
3. Какова роль сети Интернет в современном мире?

4. Перечислите возможные способы решения проблемы ограниченного количества существующих IPv4 адресов.

5. Каково назначение технологии NAT?

6. Приведите основные типы технологии NAT.

7. Каков порядок взаимодействия хоста с маршрутизатором (шлюзом) офисной сети при использовании технологии NAT?

8. Опишите взаимодействие маршрутизаторов локальной сети и сети провайдера с применением NAT.

9. Назовите основные причины применения технологии NAT.

10. Перечислите основные особенности доступа к серверу локальной сети из сети провайдера.

11. Какие последствия влечет за собой отключение NAT?

12. Перечислите основные типы технологии NAT.

13. Что такое Outside local address и для чего он предназначен?

14. Каково назначение Wildcard bits (mask)?

15. Каковы возможности повышения безопасности сети при использовании технологии NAT?

16. Приведите особенности технологии PAT.

17. Назовите преимущества технологии NAT.

18. Перечислите недостатки технологии NAT.

19. Каковы перспективы применения технологии NAT?

20. Что придет на замену технологии NAT?

Практическое занятие № 5
«Беспроводные сети Wi-Fi»

Цель занятия: изучить технологии беспроводных сетей Wi-Fi и порядок их настройки.

5.1. Краткие теоретические сведения

5.1.1. Беспроводные технологии передачи данных

С момента появления стандарта IEEE 802.11 в 1997 году беспроводные сети стали неотъемлемой частью современной коммуникационной инфраструктуры. Они обеспечивают мобильность и гибкость, позволяя пользователям взаимодействовать независимо от их местоположения и используемого оборудования.

Беспроводные технологии передачи данных охватывают широкий спектр электромагнитных волн, начиная от низкочастотных радиоволн и заканчивая видимым светом, что открывает множество возможностей для передачи информации (рис. 5.1).

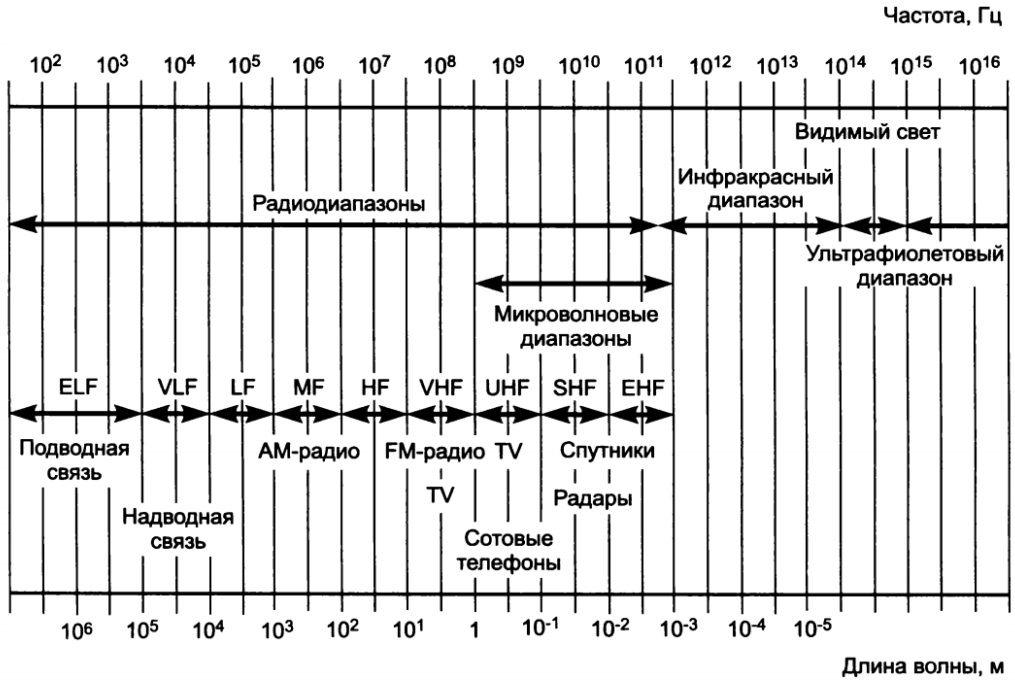

Рис. 5.1. Диапазон электромагнитного спектра

Электромагнитный спектр, используемый в беспроводной связи, варьируется от нескольких килогерц до сотен терагерц. Каждый диапазон частот имеет свои уникальные свойства, которые влияют на скорость передачи данных, дальность распространения сигнала и его способность преодолевать препятствия.

Низкочастотные волны (до 2 МГц) – сигналы, которые распространяются вдоль поверхности Земли, что позволяет им преодолевать

значительные расстояния. Например, радиоволны АМ-диапазона могут передаваться на сотни километров, что делает их идеальными для широковещательного радио.

Средние частоты (2–30 МГц) – сигналы, которые способны отражаться от ионосферы, что позволяет им распространяться на тысячи километров. Это свойство используется в коротковолновой радиосвязи.

Высокочастотные волны (свыше 30 МГц) – сигналы, которые распространяются только по прямой линии (прямая видимость – Line Of Sight, LOS). Они не могут огибать препятствия и требуют четкой линии между передатчиком и приемником. Например, микроволновые сигналы (выше 4 ГГц) подвержены поглощению водой, что делает их чувствительными к погодным условиям, таким как дождь или туман.

Одной из ключевых характеристик беспроводной связи является **скорость передачи данных**. Чем выше частота сигнала, тем больше информации можно передать за единицу времени. Однако с увеличением частоты ухудшается способность сигнала проникать через препятствия, такие как стены или перекрытия. Например, низкочастотные радиоволны легко проходят через стены, тогда как инфракрасный и видимый свет требует прямой видимости между передатчиком и приемником.

Механизмы распространения сигнала

При передаче сигналов в беспроводных сетях важно учитывать, как они взаимодействуют с окружающей средой. Основные механизмы распространения сигнала представлены на рис. 5.2 и включают отражение, дифракцию и рассеивание.

Отражение – когда сигнал встречает препятствие, размеры которого значительно превышают длину волны, часть энергии отражается. Это явление часто наблюдается в городских условиях, где сигналы отражаются от зданий.

Дифракция – если сигнал сталкивается с непроницаемым препятствием, он может огибать его, что позволяет принимать сигнал даже вне зоны прямой видимости.

Рассеивание – при встрече с препятствием, размеры которого сопоставимы с длиной волны, сигнал рассеивается в разных направлениях. Это приводит к многолучевому распространению, когда приемник получает несколько копий одного и того же сигнала.

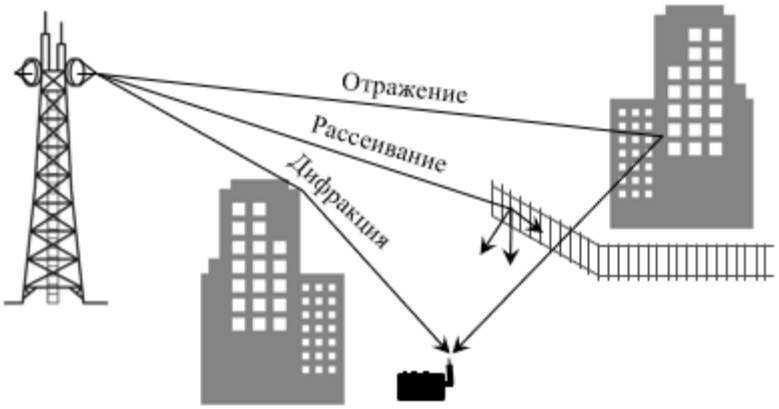

Рис. 5.2. Механизмы распространения сигнала после встречи с препятствием

Одной из основных проблем беспроводных сетей является высокий уровень помех и ошибок передачи данных. В отличие от проводных линий связи, где интенсивность битовых ошибок (BER) может достигать 10^{-9}–10^{-10}, в беспроводных сетях этот показатель часто составляет 10^{-3}. Это связано с многолучевым распространением, поглощением сигнала и другими факторами.

Для повышения надежности беспроводной связи используются различные методы:

• **широкополосное кодирование** – распределение энергии сигнала по широкому диапазону частот позволяет снизить влияние помех и повысить устойчивость связи;

• **оптимизация размещения оборудования** – установка передатчиков и приемников на высоких башнях или зданиях помогает минимизировать многократные отражения и улучшить качество сигнала;

• **протоколы с повторной передачей** – на канальном уровне используются протоколы, которые позволяют быстро обнаруживать и исправлять ошибки. Это особенно важно для минимизации задержек, которые могут возникать при использовании протоколов транспортного уровня, таких как TCP.

5.1.2. Стандарты и особенности применения технологии беспроводной локальной сети Wi-Fi

Технология Wi-Fi, основанная на стандартах IEEE 802.11, стала одной из самых популярных технологий для организации беспроводных локальных сетей (**WLAN**). Она обеспечивает высокоскоростную передачу данных без необходимости использования проводов, что делает ее незаменимой в современных домах, офисах и общественных пространствах

С момента своего появления Wi-Fi претерпел значительные изменения, и сегодня существует несколько поколений стандартов, каждое из которых предлагает улучшенные характеристики. Рассмотрим эволюцию стандартов Wi-Fi.

IEEE 802.11a – один из первых стандартов, который позволил достичь скорости передачи данных до 54 Мбит/с. Он работает в диапазоне 5 ГГц, что обеспечивает меньшую помеховую нагрузку по сравнению с более загруженным диапазоном 2,4 ГГц. Однако из-за ограниченной дальности распространения сигнала на высоких частотах его применение было ограничено.

IEEE 802.11n (Wi-Fi 4) – этот стандарт, представленный в 2009 году, стал значительным шагом вперед. Он поддерживает работу в диапазонах 2,4 ГГц и 5 ГГц, что позволяет достигать скоростей до 150 Мбит/с на одну антенну. Использование технологии MIMO (Multiple Input Multiple Output) с несколькими антеннами значительно повысило пропускную способность и устойчивость связи.

IEEE 802.11ac (Wi-Fi 5) – следующий этап развития Wi-Fi, который обеспечил скорость передачи данных до 6,77 Гбит/с при использовании восьми антенн MU-MIMO. Этот стандарт работает исключительно в диапазоне 5 ГГц и

предлагает улучшенную энергоэффективность, что особенно важно для мобильных устройств.

IEEE 802.11ax (Wi-Fi 6) – современный стандарт, утвержденный в 2021 году. Он использует технологию OFDMA (Orthogonal Frequency Division Multiple Access) для повышения спектральной эффективности и поддерживает модуляцию 1024-QAM, что увеличивает пропускную способность. Wi-Fi 6 работает в диапазонах 2,4 и 5 ГГц, а также может использовать дополнительные частоты в диапазоне от 1 до 7 ГГц. Ожидается, что этот стандарт повысит среднюю пропускную способность в четыре раза по сравнению с Wi-Fi 5.

IEEE 802.11be (Wi-Fi 7) был окончательно утвержден 2024 году. Утверждение документа означало фиксацию всех технических спецификаций, что дало производителям четкий ориентир для создания роутеров, чипсетов и клиентских устройств. Фокус сместился с разработки на внедрение и оптимизацию.

Одним из самых значимых нововведений, реализованных в стандарте, является **технология Multi-Link Operation (MLO)**. Ее суть выходит за рамки простого сложения скоростей. Раньше устройство подключалось к сети только через одну радиочастоту в один момент времени, даже если роутер был двух- или трехдиапазонным. MLO кардинально меняет эту логику. Теперь клиент, например смартфон или ноутбук, может устанавливать соединение одновременно через два или даже три разных диапазона, например через 5 и 6 ГГц. Это позволяет не просто увеличить общую пропускную способность, но и решить **две ключевые проблемы прошлых поколений**:

• **радикальное снижение задержки**, что критично для онлайн-игр, виртуальной реальности и удаленной работы;

• **значительное повышение стабильности**. Связь становится устойчивой к резким изменениям в эфире, что особенно актуально в многоквартирных домах с десятками соседних сетей. По сути, MLO создает для данных выделенную полосу с резервированием, что приближает Wi-Fi по надежности к проводным технологиям.

Другое важное усовершенствование – **расширение каналов до 320 МГц** в поддерживаемых диапазонах 6 ГГц и в некоторых конфигурациях 5 ГГц. Однако такая ширина требует очень большого и чистого частотного ресурса, который в полном объеме доступен именно в новом для Wi-Fi диапазоне 6 ГГц. Именно здесь раскрывается главный потенциал скорости, позволяя достичь теоретических пределов, близких к 40 Гбит/с.

Помимо этого, стандарт вводит **более совершенные методы координации между точками доступа в плотных средах**. Технологии координированного планирования и совместной обработки сигналов позволяют нескольким роутерам эффективнее делить эфирное пространство. Фактически новый стандарт закладывает основу для того, чтобы Wi-Fi стал полноценной заменой проводного Ethernet даже для самых требовательных профессиональных задач, предлагая при этом беспрецедентную мобильность.

Основные диапазоны частот Wi-Fi

Wi-Fi функционирует в нескольких выделенных частотных диапазонах, и с каждым новым поколением технологии их использование становится все более эффективным. Каждый диапазон обладает особыми характеристиками, определяющими его применение.

2,4 ГГц – наиболее распространенный диапазон, который обеспечивает хорошую дальность распространения сигнала, но подвержен помехам от других устройств, таких как микроволновые печи и Bluetooth-устройства.

5 ГГц – менее загруженный диапазон, который предлагает более высокие скорости передачи данных, но с меньшей дальностью распространения сигнала.

6 ГГц – новый диапазон, который стал доступен с появлением Wi-Fi 6E и ставший основным для Wi-Fi 7. Он обеспечивает еще более высокие скорости и меньшую задержку, что делает его идеальным для приложений, требующих высокой производительности. Диапазон 6 ГГц идеально подходит для самых требовательных приложений.

Wi-Fi широко используется для создания локальных сетей в домах, офисах и общественных местах. Он позволяет подключать к сети такие устройства, как компьютеры, смартфоны, планшеты, принтеры и другие гаджеты (рис. 5.3).

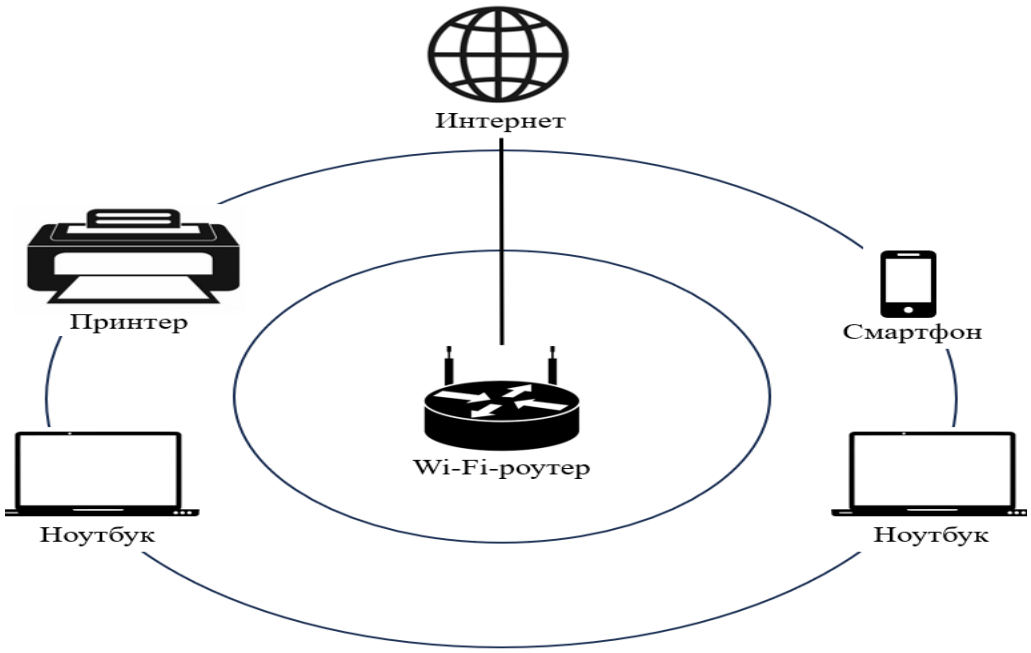

Рис. 5.3. Роутер с Wi-Fi объединяет устройства в локальную сеть

Кроме того, Wi-Fi играет ключевую роль в развитии интернета вещей (IoT), обеспечивая беспроводное подключение и управление умными устройствами, такими как термостаты, камеры видеонаблюдения, умные лампочки и замки.

5.1.3. Основы работы технологии Wi-Fi

Технология Wi-Fi стала неотъемлемой частью современной жизни, позволяя подключать устройства к сети без необходимости использования проводов. Для организации Wi-Fi-сети обычно используются роутеры или

маршрутизаторы, которые оснащены беспроводными адаптерами и антеннами для передачи и приема сигналов.

Рассмотрим **основные принципы работы** Wi-Fi.

1. Генерация и модуляция сигнала. Роутер или маршрутизатор преобразует электрический ток в радиоволны, которые используются для передачи данных. В процессе модуляции характеристики сигнала, такие как амплитуда, частота и фаза, изменяются в соответствии с передаваемой информацией. Это позволяет «упаковать» цифровые данные (последовательности нулей и единиц) в форму, пригодную для беспроводной передачи.

2. Передача сигнала. Радиоволны, созданные роутером, распространяются в пространстве и могут проходить через стены и другие препятствия. Однако качество сигнала ухудшается с увеличением расстояния и количества препятствий. Зона покрытия Wi-Fi обычно составляет несколько десятков метров внутри помещений и до 100 метров на открытых пространствах.

3. Прием и демодуляция сигнала. Устройства, такие как смартфоны, ноутбуки или планшеты, оснащены встроенными антеннами, которые принимают радиосигналы. Приемник демодулирует сигнал, преобразуя его обратно в цифровые данные, которые могут быть обработаны устройством.

Для защиты данных в беспроводных сетях используются различные методы шифрования. Наиболее распространенными являются WPA (Wi-Fi Protected Access) и WPA2, которые обеспечивают высокий уровень безопасности. Устаревший стандарт WEP (Wired Equivalent Privacy) больше не рекомендуется к использованию, так как он уязвим для взлома.

Преимущества Wi-Fi:

- **мобильность** – устройства могут свободно перемещаться в пределах зоны покрытия без потери соединения;
- **масштабируемость** – одна сеть может поддерживать подключение десятков устройств одновременно;
- **универсальность** – Wi-Fi позволяет подключать различные умные устройства, такие как камеры видеонаблюдения, термостаты и системы управления умным домом;
- **локальные сети** – устройства внутри дома или офиса могут обмениваться данными без необходимости подключения к Интернету;
- **меш-сети** – использование нескольких маршрутизаторов позволяет создавать «бесшовные» сети, обеспечивающие стабильное соединение даже в помещениях с большим количеством препятствий.

Недостатки Wi-Fi:

- **ограниченная зона покрытия** – сигнал Wi-Fi ослабевает с увеличением расстояния и при наличии препятствий, таких как стены или мебель. Для улучшения покрытия и повышения скорости передачи данных часто используются дополнительные устройства, такие как репитеры и мощные антенны;
- **помехи** – электромагнитные помехи от других устройств, работающих в том же частотном диапазоне, могут ухудшать качество соединения;

- **перегрузка каналов** – в многоквартирных домах или офисных зданиях одновременная работа множества сетей может приводить к перегрузке каналов, что снижает скорость и стабильность соединения.

Таким образом, беспроводные технологии передачи данных играют ключевую роль в современной коммуникационной инфраструктуре. Они обеспечивают мобильность и гибкость, но при этом сталкиваются с рядом вызовов, таких как ограниченная дальность распространения сигнала, чувствительность к помехам и погодным условиям. Благодаря использованию современных методов кодирования, оптимизации размещения оборудования и специализированных протоколов, беспроводные сети продолжают развиваться, обеспечивая высокую скорость и надежность передачи данных.

5.2. Порядок и основные правила выполнения заданий

Для рассмотрения сетей Wi-Fi и порядка их настройки будем использовать **симулятор Cisco Packet Tracer**. Порядок и основные правила выполнения заданий рассмотрим в ходе выполнения конкретного примера – соединения и настройки устройств беспроводной сети.

5.2.1. Построение структурных схем сети

Запустим **программу Cisco Packet Tracer**. В области **«Логическое пространство»** построим структурную схему сети в составе: компьютера – PC-PT (PC0), ноутбука – Laptop PT (Laptop0), беспроводного роутера –WRT300N (Wireless Router0) и маршрутизатора интернет-провайдера – 1841 (Router0). Подключение Laptop0 по беспроводной связи осуществим после настройки сети.

WRT300N является классическим **Wi-Fi-маршрутизатором**, который (или ему подобный) имеется практически в каждой квартире. На нем имеется один порт для подключения кабеля доступа **интернет-провайдера**, несколько портов для подключения локальных устройств и либо встроенная, либо внешняя **Wi-Fi-антенна**. В нашем случае – внешняя.

Маршрутизатор 1841 будем считать **маршрутизатором интернет-провайдера**.

Структурная схема указанной сети с запланированным соединением **Laptop0** к сети по технологии **Wi-Fi** представлена на рис. 5.4.

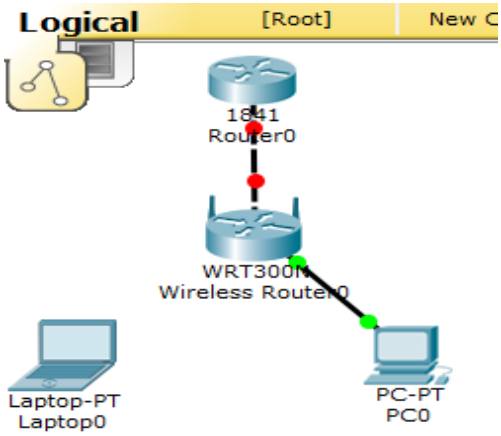

Рис. 5.4. Структурная схема

5.2.2. Настройка маршрутизатора интернет-провайдера (Router0)

Для решения этой задачи воспользуется **Cisco IOS Command Line Interface** маршрутизатора. Нажатием левой кнопки мыши открываем таблицу **Router0**, на которой подключаем вкладку **CLI**. На экране появится **IOS Command Line Interface**.

Далее осуществим настройку **IP-адреса 210.210.0.1** на интерфейсе **Router0** и **Trunk port**. Для этого выполняем ряд последовательных действий путем использования стандартных команд (рис. 5.5).

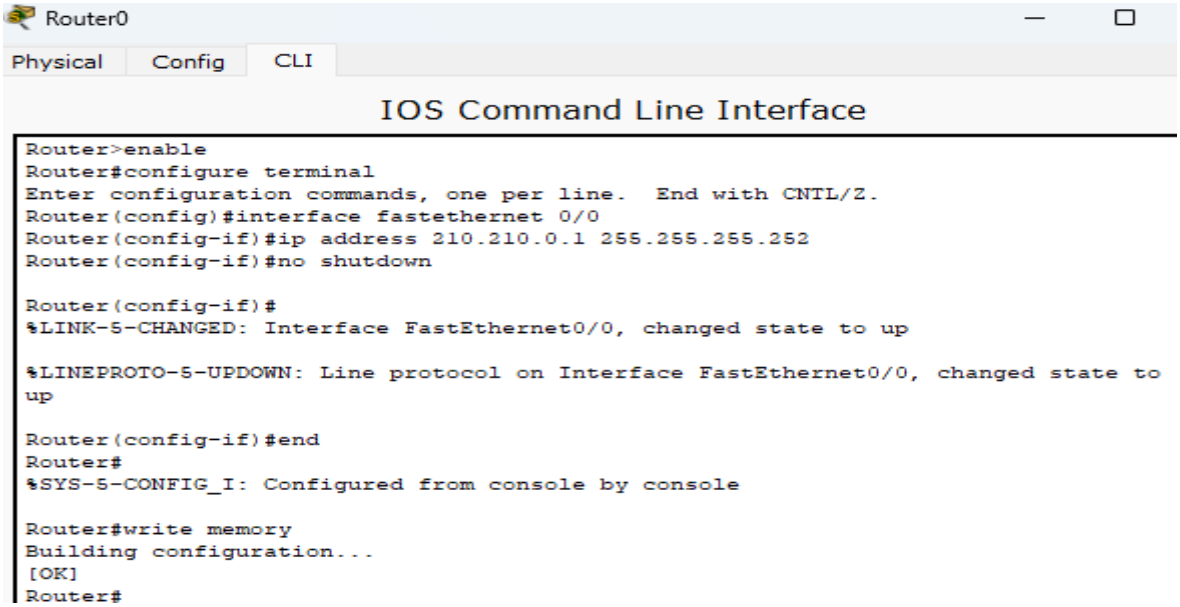

Рис. 5.5. Настройка IP-адреса на интерфейсе Router0 и магистрального порта

5.2.3. Настройка Wi-Fi-маршрутизатора (Wireless Router0)

Имеется два способа настройки **Wi-Fi-маршрутизатора**.

1. Через вкладку **Config**, где нам доступны настройки интерфейсов **Wi-Fi-маршрутизатора**.

2. Посредством симулирования графического интерфейса пользователя (**GUI**, Graphical User Interface).

Мы сосредоточимся на использовании интерфейса **GUI**.

На начальном этапе в разделе **Internet Setup** во вкладке **Setup** необходимо настроить IP-адрес для внешнего интерфейса маршрутизатора **Wireless Router0**, который предназначен для подключения к вашему и**нтернет-провайдеру**.

Существует три основных варианта настройки IP-адреса:

1. Автоматическое получение IP-адреса через **DHCP**.

2. Статический IP-адрес, который предоставляет интернет-провайдер.

3. Применение протокола **PPPoE** (Point-to-Point Protocol over Ethernet), который используется для передачи данных через Ethernet.

Кроме указанных в современных маршрутизаторах еще может использоваться **L2TP** (Layer 2 Tunnelling Protocol – протокол туннелирования второго уровня).

Для простоты выберем второй вариант – **Static IP (статический IP-адрес)**. Наведем курсор на **Wi-Fi-маршрутизатор** и нажмем левую кнопку мыши.

Появится таблица **Wireless Router0**. Выберем вкладку **GUI**. Появится таблица, верхнюю часть которой мы заполним, введя IP-адрес 210.210.0.2, как указано на рис. 5.6.

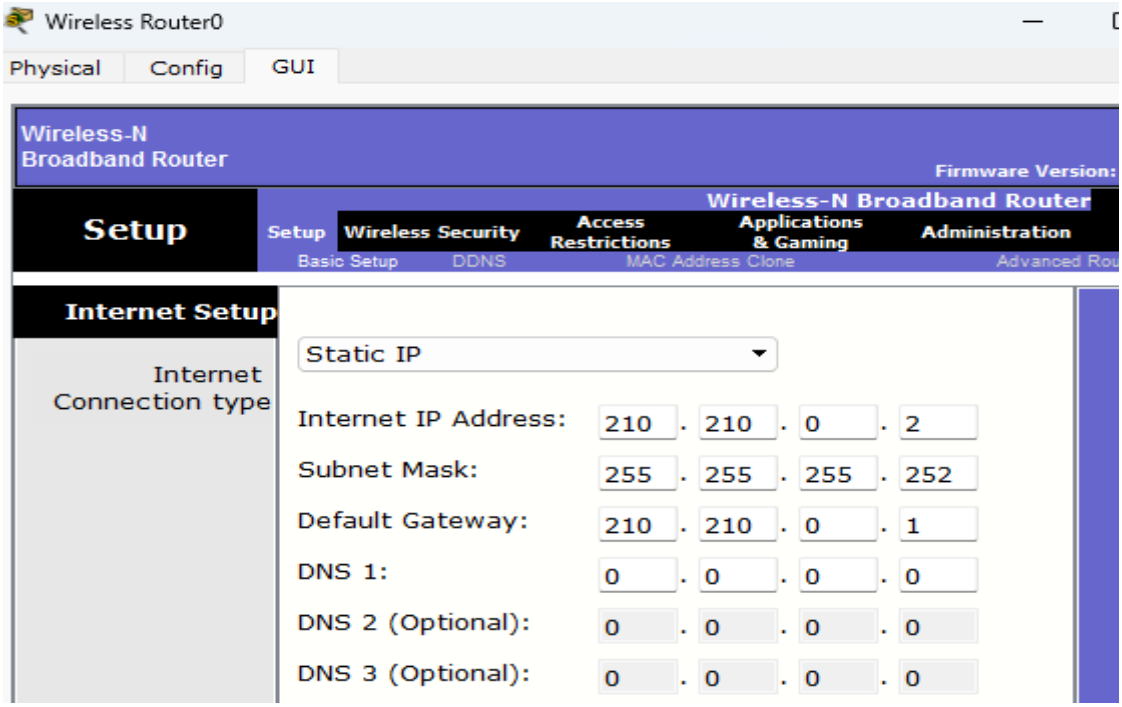

Рис. 5.6. Настройка IP-адреса Wireless Router0

По желанию можно задать **DNS-сервера**, но в нашем случае это не требуется. На этой же вкладке ниже в разделе настройки сети (**Network Setup**) по умолчанию предложены внутренние **IP-адреса**, т. е. те, которые будут раздаваться по **Wi-Fi** (рис. 5.7).

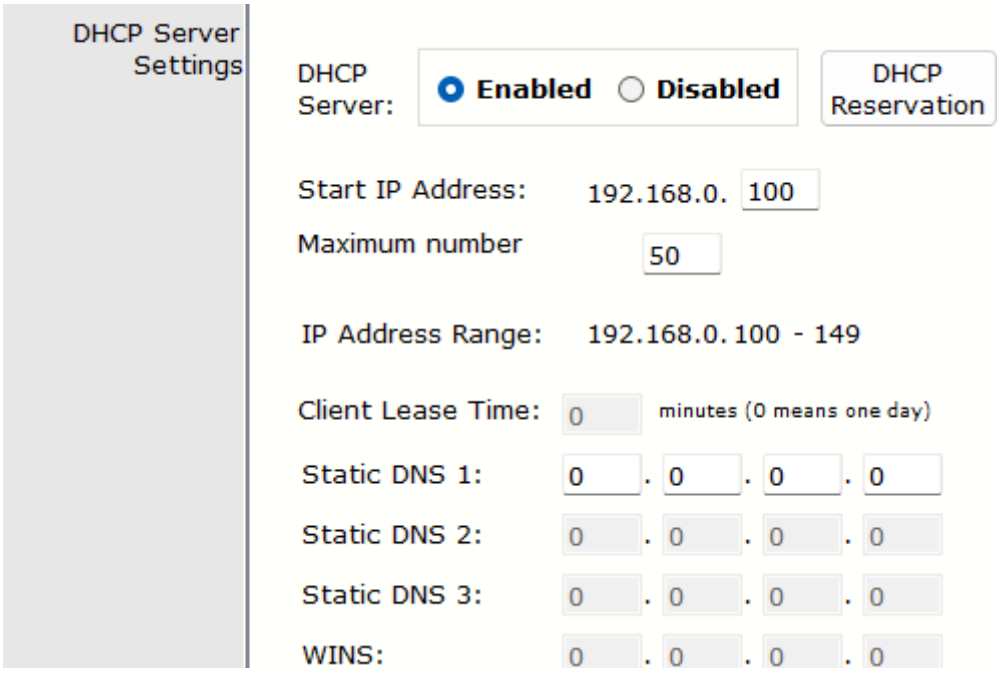

Рис. 5.7. Настройка раздела DHCP Server Settings

В разделе прописано, что **IP-адреса** начинаются с **192.168.0.100**. Их максимальное количество равно **50**. Можно убедиться, что частный **IP-адрес Wireless Router0 – 192.168.0.1.** Согласимся с этим диапазоном.

Также по умолчанию здесь **включен (Enabled)** протокол **DHCP**, что предопределяет автоматическое выделение IP-адреса и маски сети.

Нажатием кнопки **«Сохранить настройки» (Save Setting)** сохраняем заданные параметры вкладки **Setup**.

Далее переходим к настройкам в разделе **Basic Wireless Settings** вкладки **Wireless**.

Здесь мы можем выбрать режим сети (**Network Mode**). Выбираем смешанный режим (**Mixed**), что позволит подключать любой режим из трех имеющихся.

В графе **Network Name (SSID)** введем любой идентификатор сети, например **ipe168Bsuir**.

В графе **Radio Band** (ширина канала) оставим значение **Auto**.

В графе **Standart Channel** можно выбрать частоту. На данном **Wi-Fi-маршрутизаторе** представлена только **частота 2,4**, поэтому наиболее предпочтительно оставить **режим 1 – 2.412 Ghz**.

В графе **SSID Broadcast** (трансляция) задействовано **Enabled** (включено). Это означает, что все ближайшие устройства, включенные в сеть **Wi-Fi**, будут видеть нашу беспроводную сеть.

Общий вид настроенного раздела **Basic Wireless Settings** вкладки **Wireless** представлен на рис. 5.8.

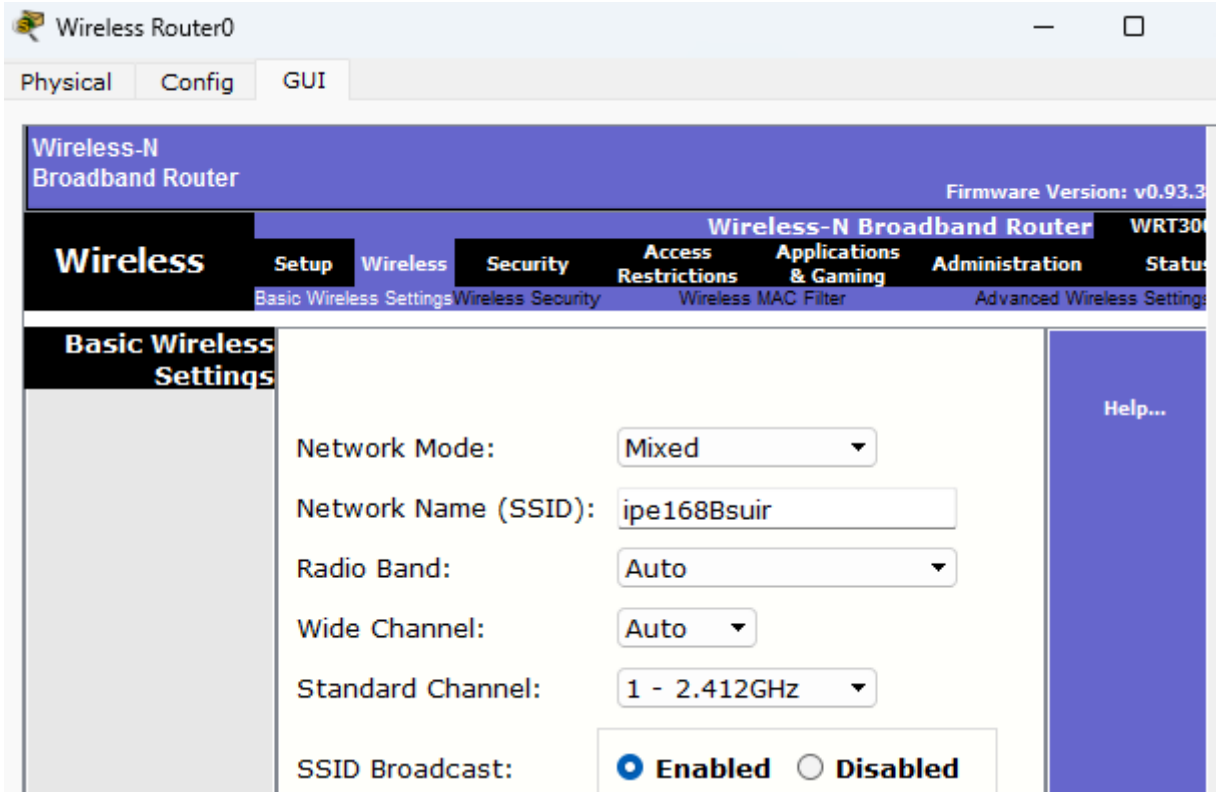

Рис. 5.8. Результат настройки раздела Basic Wireless Settings

Затем нажатием кнопки **Save Setting** (внизу таблицы) сохраним заданные настройки.

Следующим и завершающим настраиваемым нами разделом является **Wireless Security** вкладки **Wireless**.

Здесь в графе **«Безопасный режим»** (**Security Mode**) можно выбрать один из нескольких режимов. Первые по списку режимы недостаточно надежны. Поэтому выберем режим **WPA2 Personal**. Сразу же появляются дополнительные графы для настраивания.

В графе **Encryption** (шифрование) оставляем **AES** (Advanced Encryption Standard – симметричный алгоритм блочного шифрования).

В графе **Passphrase** (парольная фраза, не менее восьми символов для AES) введем парольное слово, например **Happiness14**.

Результат настройки раздела **Wireless Security** вкладки **Wireless** представлен на рис. 5.9.

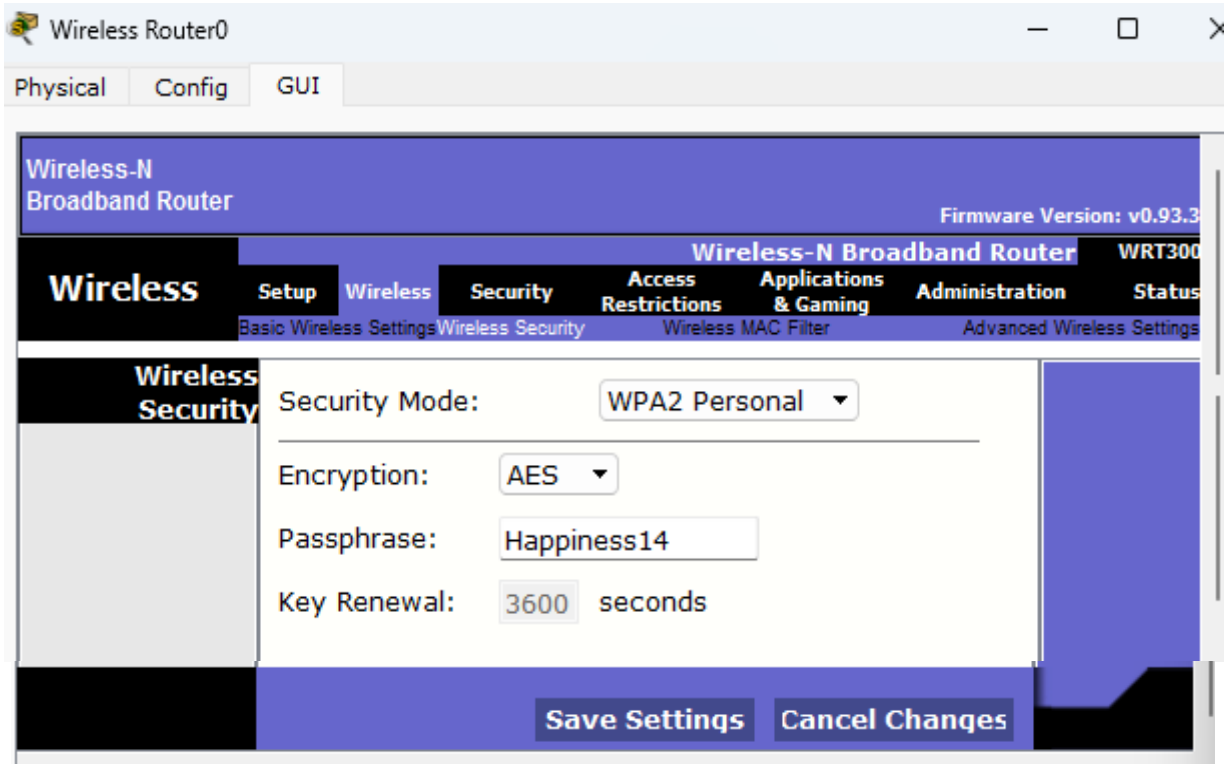

Рис. 5.9. Настройка раздела Wireless Security вкладки Wireless

Также нажатием кнопки **Save Setting** сохраним заданные настройки.

Здесь следует **добавить**, что **технологию NAT** мы не настраиваем, так как практически на всех **Wi-Fi-маршрутизаторах NAT** включена по умолчанию.

5.2.4. Настройка компьютера (PC0)

Подключение **PC0** к **Wi-Fi-маршрутизатору** выполнено при помощи витой пары. Откроем у **PC0** вкладку **Desktop → IP Configuration**. На вкладке **IP Configuration** подключим протокол **DHCP**. **PC0** сразу же получил **IP-адрес 192.168.0.100** (рис. 5.10).

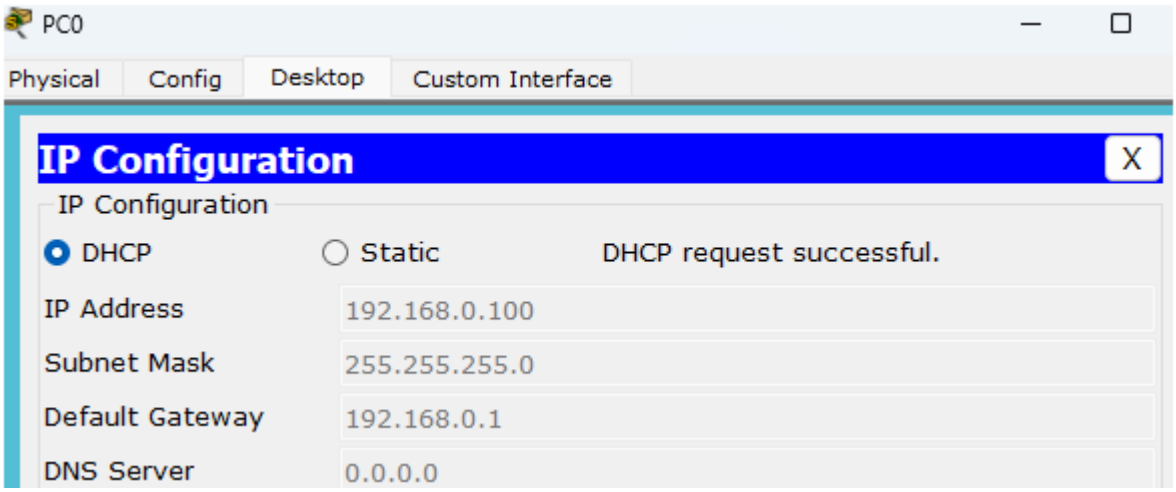

Рис. 5.10. Настройка PC0

С целью проверки качества соединения **PC0** откроем в нем вкладки **Desktop → Command Promt** и в появившемся окне введем команду-утилиту **Ping** по IP-адресу маршрутизатора интернет-провайдера – 210.210.0.1 и, например, частному IP-адресу Wi-Fi-маршрутизатора – 192.168.0.1. Появились сообщения об успешном прохождении утилиты **Ping** до мест назначения и обратно.

5.2.5. Установка Wi-Fi-модуля на ноутбук (компьютер)

Ноутбуки и компьютеры в **симуляторе Cisco Packet Tracer** не имеют в наличии **Wi-Fi-модуля**. Поэтому необходимо провести его установку (рис. 5.11).

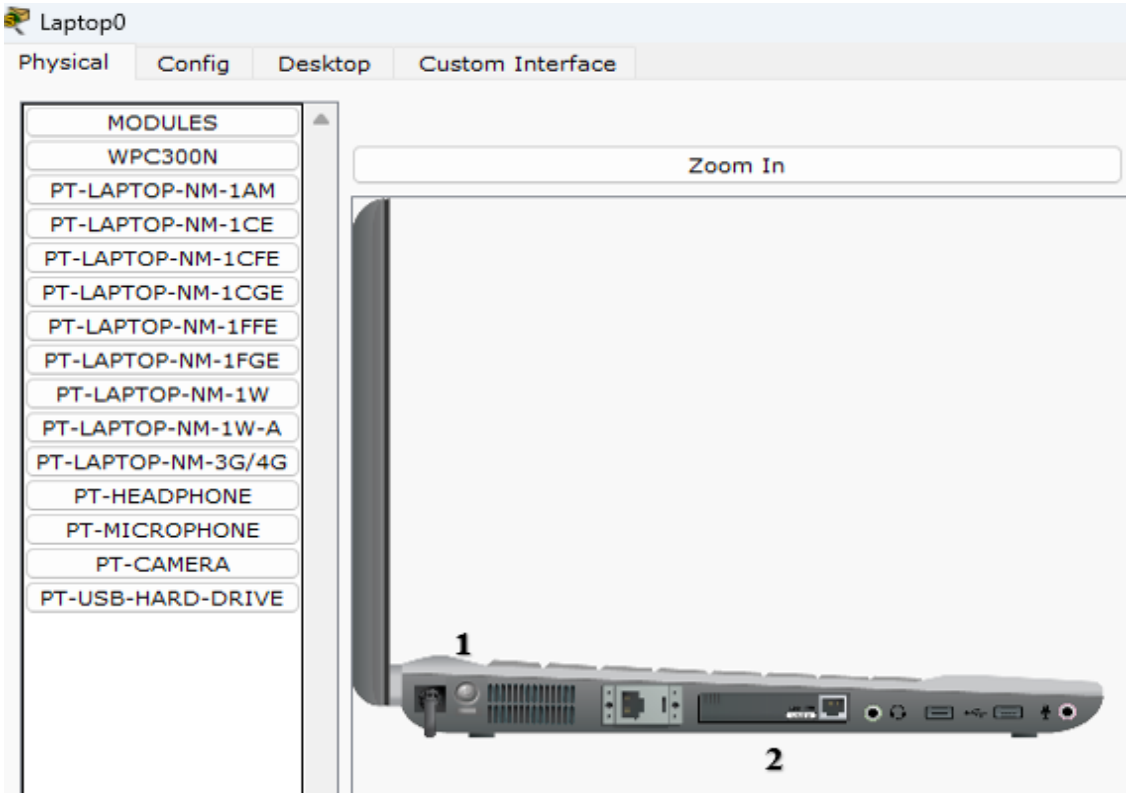

Рис. 5.11. Схема установки Wi-Fi-модуля на ноутбук

Для этого во вкладке **Physical** следует отключить **Laptop0**, нажав кнопку питания **1**, и вынуть имеющийся **модуль 2**. С этой целью необходимо мышью выбрать прямоугольник (модуль) 2 и потянуть его в направлении надписей серии **PT-Laptop-NM**. На его место поставить **Wi-Fi-модуль (WPC300N)**, протянув его мышью в освободившуюся нишу ноутбука **2**. После этого снова включаем **Laptop0,** нажав кнопку **1**.

Установка **Wi-Fi-модуля** на компьютер также проводится через вкладку **Physical**, аналогично установке на ноутбук (рис. 5.12).

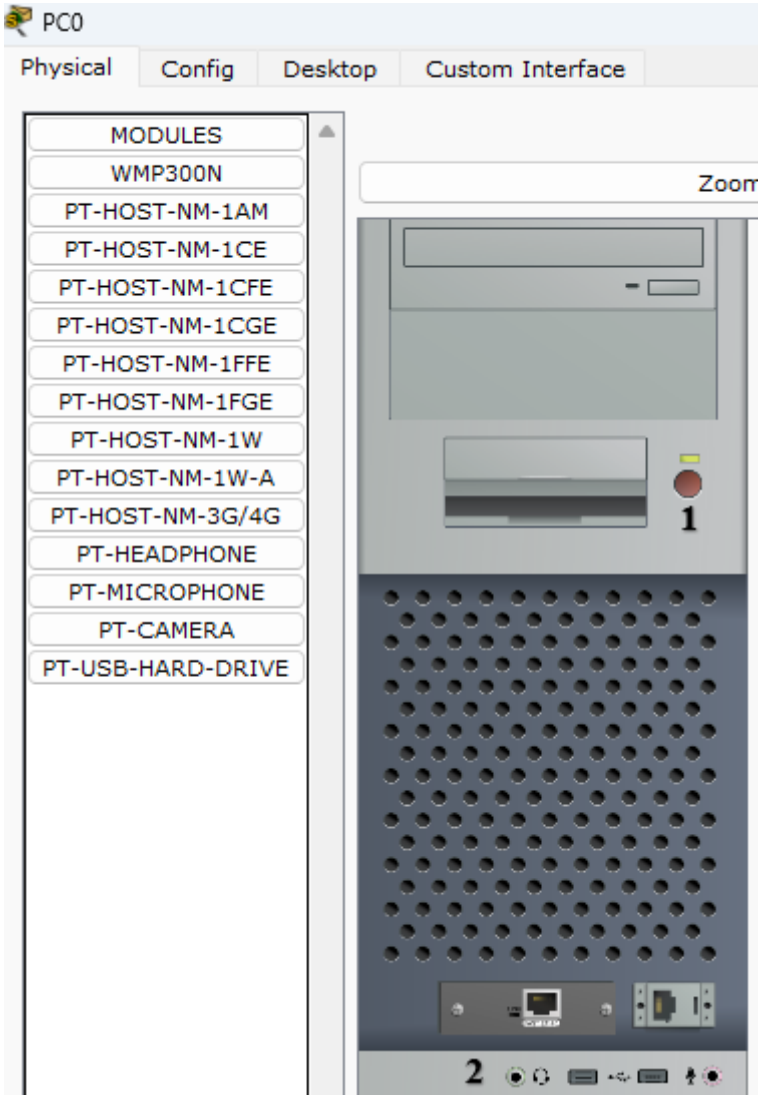

Рис. 5.12. Схема установки Wi-Fi-модуля на компьютер

5.2.6. Настройка ноутбука (Laptop0)

Переходим во вкладку **Desktop → PC Wireless**.

Выбираем вкладку **Connect** (вверху).

В разделе **Wireless Network Name** через короткое время мы видим наш идентификатор сети – **ipe168Bsuir**, заданный на **Wi-Fi-маршрутизаторе**, который и выбираем, наведя на него курсор и нажав левую кнопку мыши (рис. 5.13).

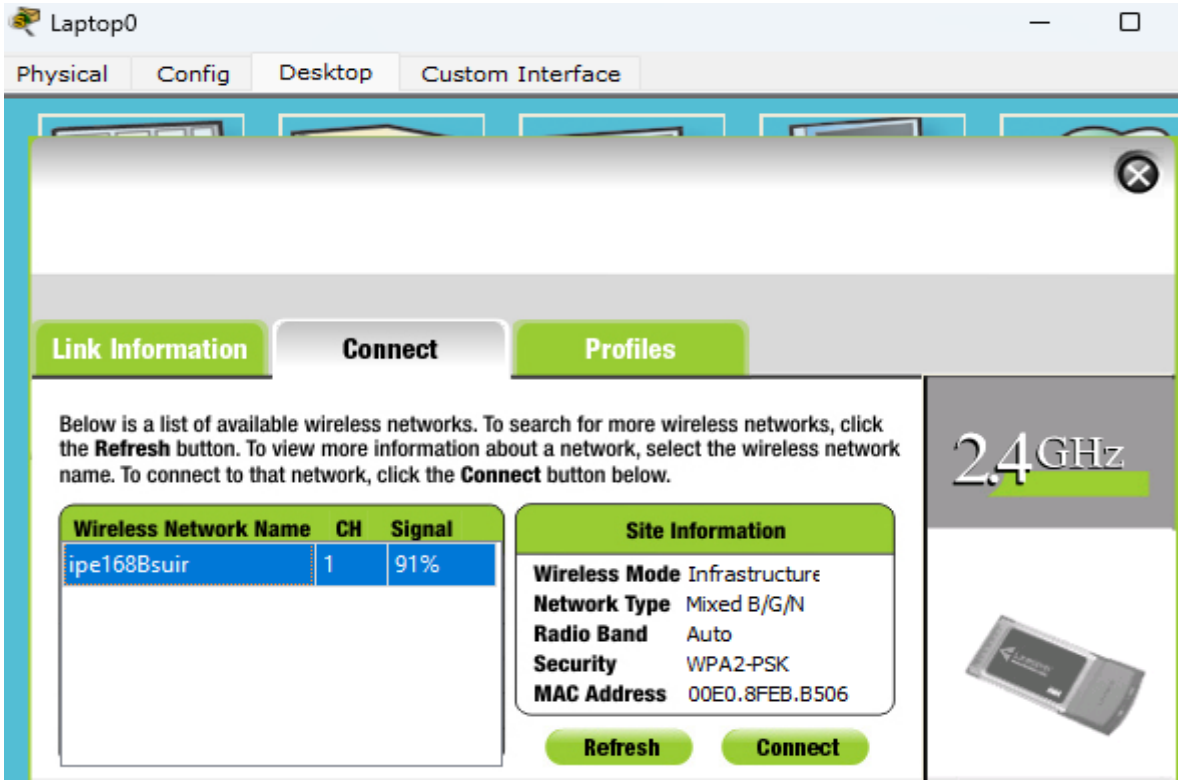

Рис. 5.13. Выбор назначенного идентификатора сети

Выбираем вкладку **Connect** (внизу) и в появившейся строке **Security** соглашаемся с типом подключения **WPA2-Personal**, в строку **Pre-shared Key** вписываем заданное на **Wi-Fi-маршрутизаторе** кодовое слово **Happiness14** (рис. 5.14).

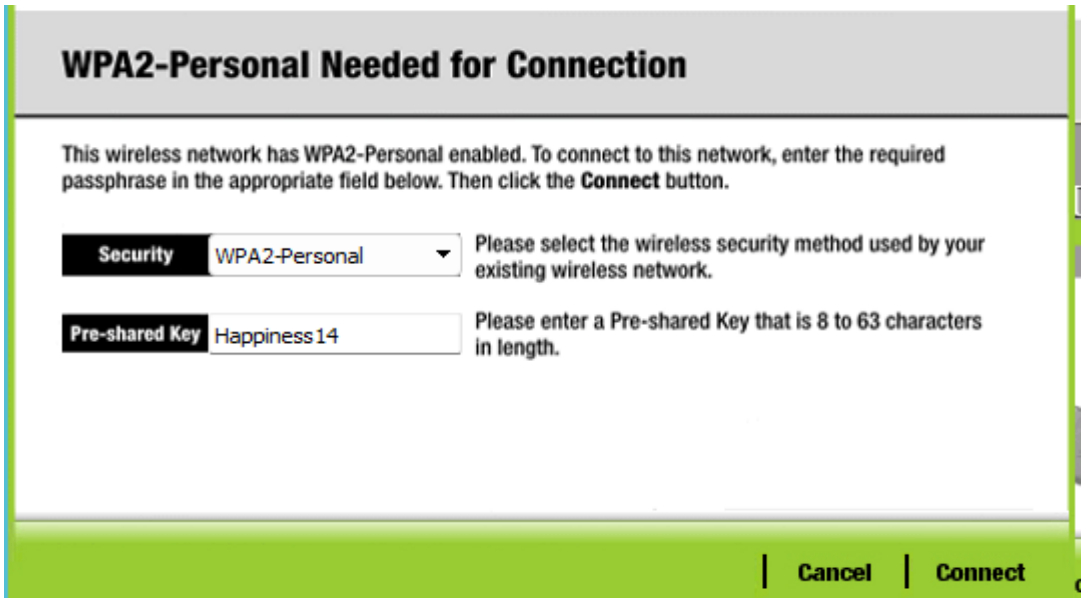

Рис. 5.14. Запись кодового слова

Выбираем вкладку **Connect**. По появившейся прерывистой линии можно сделать вывод, что Wi-Fi-подключение прошло успешно (рис. 5.15).

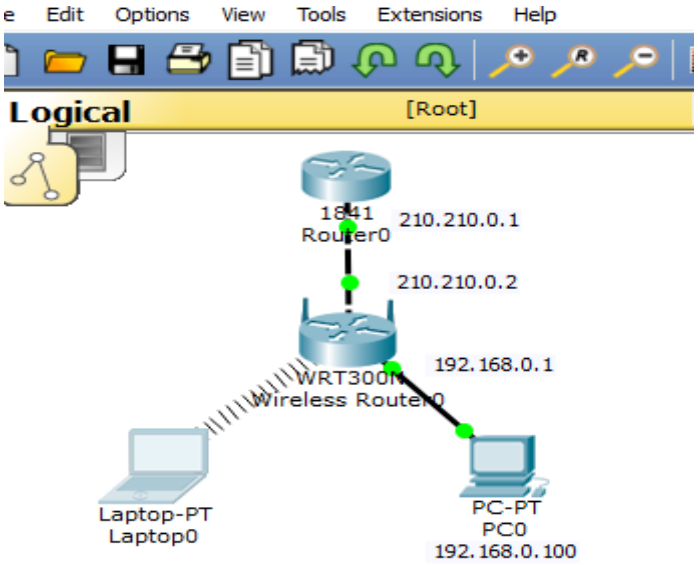

Рис. 5.15. Настроенная локальная сеть с беспроводным соединением

Посмотрим, какой **IP-адрес** присвоен **Laptop0** протоколом **DHCP**.

Для этого у Laptop0 откроем вкладку **Desktop → Command Promt** и введем команду **Ipconfig**. Или другой вариант: **Desktop → IP Configuration**. Удостоверимся, что **IP-адрес Laptop0 – 192.168.0.102** (рис. 5.16).

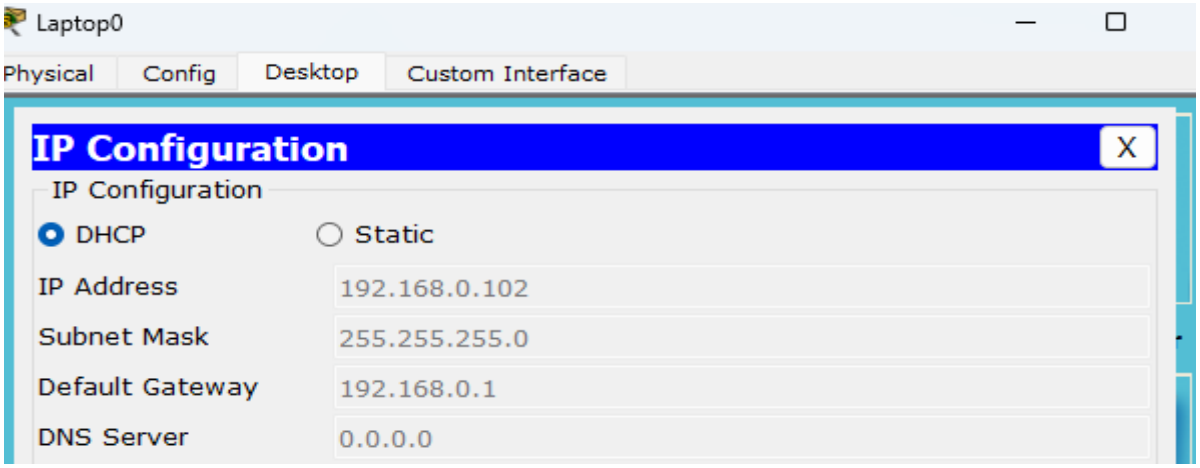

Рис. 5.16. Назначение IP-адреса Laptop0

5.2.7. Проверка функционирования беспроводной сети

Для проверки функционирования беспроводной сети во вкладке ноутбука **Command Promt** введем утилиту **Ping** и последовательно направим по IP-адресу маршрутизатора интернет-провайдера – 210.210.0.1, внешнему и частному IP-адресам Wi-Fi-маршрутизатора, соответственно, 210.210.0.2 и 192.168.0.100.

Таким образом, мы образовали локальную сеть с использованием Wi-Fi-подсети и классической проводной подсети.

5.3. Практическое задание

5.3.1. Изучить сведения, изложенные в теоретических материалах данного практического занятия и, при необходимости, в дополнительных материалах.

5.3.2. С целью изучения технологии беспроводных сетей Wi-Fi и порядка их настройки необходимо выполнить следующие действия.

5.3.2.1. Построить структурную схему локальной сети, состоящую для всех вариантов из беспроводного роутера и маршрутизатора интернет-провайдера, а также компьютеров и ноутбуков, согласно указанному преподавателем номеру варианта задания, приведенному в табл. 5.1.

5.3.2.2. Провести настройку маршрутизатора интернет-провайдера.

5.3.2.3. Настроить Wi-Fi-маршрутизатор.

5.3.2.4. Настроить проводные и беспроводные хосты согласно варианту. При этом два любых хоста должны быть соединены беспроводной связью.

5.3.2.5. Проконтролировать функционирование построенной локальной сети.

5.3.3. Отчет о проделанной работе представить преподавателю непосредственно на компьютере или в виде сканирования ключевых эпизодов выполнения работы в соответствии с п. 5.3.2.

5.3.4. Ответить на контрольные и дополнительные вопросы.

Варианты заданий

Таблица 5.1

Перечень вариантов заданий

Вариант	Исходные данные	Вариант	Исходные данные
1	1. ПК, три Laptop. 2. Белый IP-адрес сети – 222.210.0.0. 3. Серый IP-адрес сети – 192.168.120.0. 4. Название сети – group0101. 5. Пароль – Happiness1	8	1. Два ПК, два Laptop. 2. Белый IP-адрес сети – 218.210.0.0. 3. Серый IP-адрес сети – 192.168.80.0. 4. Название сети – group0108. 5. Пароль – Happiness8
2	1. Два ПК, два Laptop. 2. Белый IP-адрес сети – 212.210.0.0. 3. Серый IP-адрес сети – 192.168.20.0. 4. Название сети – group0102. 5. Пароль – Happiness2	9	1. ПК, три Laptop. 2. Белый IP-адрес сети – 219.210.0.0. 3. Серый IP-адрес сети – 192.168.90.0. 4. Название сети – group0109. 5. Пароль – Happiness9
3	1. Три ПК, Laptop. 2. Белый IP-адрес сети – 223.210.0.0. 3. Серый IP-адрес сети – 192.168.130.0. 4. Название сети – group0103. 5. Пароль – Happiness3	10	1. ПК, четыре Laptop. 2. Белый IP-адрес сети – 220.210.0.0. 3. Серый IP-адрес сети – 192.168.100.0. 4. Название сети – group0110. 5. Пароль – Happiness10
4	1. Два ПК, четыре Laptop. 2. Белый IP-адрес сети – 224.210.0.0. 3. Серый IP-адрес сети – 192.168.140.0. 4. Название сети – group0104. 5. Пароль – Happiness4	11	1. Три ПК, два Laptop. 2. Белый IP-адрес сети – 221.210.0.0. 3. Серый IP-адрес сети – 192.168.110.0. 4. Название сети – group0111. 5. Пароль – Happiness11

Окончание табл. 5.1

Вариант	Исходные данные	Вариант	Исходные данные
5	1. Три ПК, два Laptop. 2. Белый IP-адрес сети – 215.210.0.0. 3. Серый IP-адрес сети – 192.168.50.0. 4. Название сети – group0105. 5. Пароль – Happiness5	12	1. Два ПК, Laptop. 2. Белый IP-адрес сети – 217.210.0.0. 3. Серый IP-адрес сети – 192.168.70.0. 4. Название сети – group0112. 5. Пароль – Happiness12
6	1. Два ПК, Laptop. 2. Белый IP-адрес сети – 216.210.0.0. 3. Серый IP-адрес сети – 192.168.60.0. 4. Название сети – group0106. 5. Пароль – Happiness6	13	1. ПК, два Laptop. 2. Белый IP-адрес сети – 211.210.0.0. 3. Серый IP-адрес сети – 192.168.10.0. 4. Название сети – group0113. 5. Пароль – Happiness13
7	1. ПК, Три Laptop. 2. Белый IP-адрес сети – 213.210.0.0. 3. Серый IP-адрес сети – 192.168.30.0. 4. Название сети – group0107. 5. Пароль – Happiness7	14	1. ПК, четыре Laptop. 2. Белый IP-адрес сети – 214.210.0.0. 3. Серый IP-адрес сети – 192.168.40.0. 4. Название сети – group0114. 5. Пароль – Happiness14

Контрольные вопросы

1. Что является беспроводной сетью?
2. Что такое Wi-Fi-сети?
3. Каким стандартом регламентируется беспроводная среда передачи данных?
4. Чем характерен диапазон электромагнитного спектра?
5. Перечислите общие закономерности распространения электромагнитных волн.
6. Перечислите механизмы распространения сигнала после встречи с препятствиями.
7. В чем различие битовых ошибок проводной и беспроводной линий связи?
8. Как помехи влияют на беспроводную связь?
9. Как влияет на помехоустойчивость беспроводных линий связи передача на канальном уровне?
10. Назовите особенности применения беспроводной локальной сети Wi-Fi.
11. Назовите особенности применения стандарта для Wi-Fi 4.
12. Назовите особенности применения стандарта для Wi-Fi 5.
13. Назовите особенности применения стандарта для Wi-Fi 6.
14. Каковы перспективы стандарта IEEE 802.11be?
15. Приведите основные характеристики беспроводной локальной сети Wi-Fi.
16. Перечислите достоинства беспроводной локальной сети Wi-Fi.
17. Перечислите недостатки беспроводной локальной сети Wi-Fi.
18. Назовите наиболее распространенные частоты функционирования локальной сети Wi-Fi.
19. Каковы возможности применения в сети Wi-Fi технологии NAT?
20. В чем заключается особенность распространения сигнала по механизму дифракции?

Практическое занятие № 6
«Назначение, принципы работы и настройки межсетевого экрана»

Цель занятия: изучить назначения, принципы работы и настройки межсетевого экрана CISCO.

6.1. Краткие теоретические сведения

6.1.1. Определение и описание межсетевых экранов

Межсетевой экран (также известный как файрвол или брандмауэр) – это специализированное программное или программно-аппаратное решение, предназначенное для контроля и фильтрации сетевого трафика между различными сегментами сети. Его основная задача заключается в обеспечении безопасности путем ограничения несанкционированного доступа и предотвращения потенциальных угроз. Межсетевой экран действует как барьер, разделяющий сеть на зоны с разным уровнем доверия, и применяет заранее заданные правила для управления передачей данных (рис. 6.1).

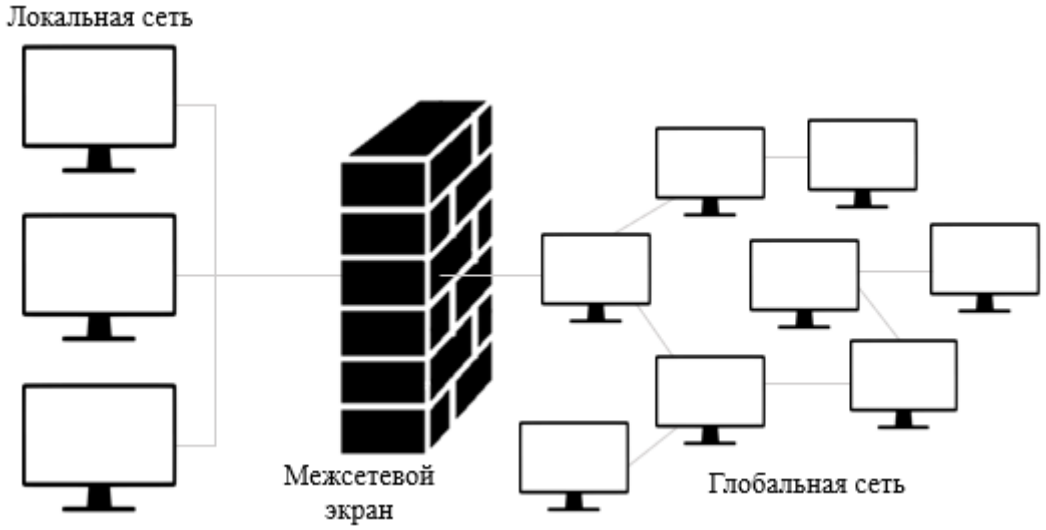

Рис. 6.1. Межсетевой экран на границе сетевого периметра

Первые межсетевые экраны появились в начале 1990-х годов, когда Интернет начал активно развиваться, и возникла необходимость в защите внутренних сетей от внешних угроз. Изначально их функции выполняли маршрутизаторы, которые анализировали заголовки пакетов на сетевом уровне. С течением времени технологии совершенствовались, и файрволы стали поддерживать фильтрацию на более высоких уровнях модели OSI, включая транспортный уровень. Это позволило повысить точность и гибкость управления трафиком.

К основным функциям межсетевых экранов относятся:

• **контроль доступа** – межсетевой экран регулирует доступ к ресурсам сети, разрешая или блокируя передачу данных на основе заданных правил. Это позволяет защитить внутреннюю сеть от несанкционированного доступа извне;

• **фильтрация трафика** – файрвол анализирует каждый пакет данных, проверяя его соответствие установленным критериям. Если пакет не соответствует правилам, он блокируется;

• **разделение сети** – межсетевой экран разделяет сеть на зоны с разным уровнем доверия, например внутреннюю сеть и Интернет. Это помогает минимизировать риски утечки данных и атак из внешней среды;

• **логирование и мониторинг** – файрволы регистрируют все попытки доступа и передачи данных, что позволяет администраторам отслеживать подозрительную активность и оперативно реагировать на угрозы.

Принципы работы межсетевых экранов

Межсетевой экран функционирует как система фильтров, которые последовательно обрабатывают входящий и исходящий трафик. Каждый фильтр соответствует определенному правилу, и порядок этих правил напрямую влияет на производительность и эффективность файрвола. Существует два основных подхода к обработке трафика.

1. «Разрешено все, что не запрещено» – в этом случае файрвол пропускает все пакеты, которые не подпадают под запрещающие правила. Такой подход менее строгий, но требует меньших усилий для настройки.

2. «Запрещено все, что не разрешено» – здесь файрвол блокирует весь трафик, кроме того, который явно разрешен правилами. Этот метод обеспечивает более высокий уровень безопасности, но требует тщательной настройки и постоянного обновления правил.

К операциям, выполняемым межсетевым экраном, относятся:

• **Allow (разрешить)** – пакет передается дальше по сети;

• **Deny (запретить)** – пакет блокируется без уведомления отправителя;

• **Reject (отклонить)** – пакет блокируется, но отправитель получает уведомление о недоступности сервиса. Этот метод менее безопасен, чем **Deny**, так как раскрывает информацию о сети.

Работу межсетевого экрана можно сравнить с системой безопасности в здании. Представьте, что ваша сеть – это здание, а файрвол – это охрана на входе. Все посетители должны пройти через контрольно-пропускной пункт, где их проверяют и идентифицируют. Однако если злоумышленнику удастся обойти охрану, он сможет свободно перемещаться внутри здания. Это подчеркивает важность обеспечения безопасности не только на границе сети, но и внутри нее.

В этой связи важное значение имеют ключевые принципы работы файрвола:

• **единая точка контроля** – весь трафик должен проходить через межсетевой экран, что позволяет централизованно управлять безопасностью;

• **полный контроль и регистрация** – файрвол должен отслеживать и записывать все попытки передачи данных, чтобы администраторы могли анализировать и предотвращать угрозы;

• **защита самого файрвола** – межсетевой экран должен быть надежно защищен от внешних атак, так как его взлом может привести к компрометации всей сети.

На сегодняшний день развертывание межсетевого экрана является стандартной и обязательной практикой для обеспечения безопасности любого компьютера или сети с выходом в Интернет. Пренебрежение этим элементарным средством защиты значительно повышает риск компрометации системы, что может привести к заражению вредоносным программным обеспечением, несанкционированному доступу и краже данных.

6.1.2. Сходство и различия межсетевого экрана и маршрутизатора

Функции межсетевого экрана и маршрутизатора во многом схожи. Оба устройства поддерживают:

- **динамическую маршрутизацию**, включая **RIP** (Routing Information Protocol – протокол маршрутной информации), **OSPF** (Open Shortest Path First – протокол динамической маршрутизации), **EIGRP** (Enhanced Interior Gateway Routing Protocol – протокол маршрутизации, разработанный компанией Cisco);
- **NAT**;
- **фильтрацию трафика посредством ACL** (Access Control List – список управления доступом);
- **VPN** (Virtual Private Network – виртуальная частная сеть), в том числе **Site-to-Site**, **RA VPN** (Remote Access VPN – шлюзы защищенного удаленного доступа).

Вместе с тем межсетевой экран – это в первую очередь устройство безопасности. И в этой связи многие функции защиты сети на межсетевом экране включены по умолчанию, в то время как на маршрутизаторе их необходимо настраивать. К примеру, в семействе межсетевых экранов **CISCO ASA** (Adaptive Security Appline) весь трафик по умолчанию запрещен, а на маршрутизаторах – разрешен. Такая функция, как межсетевое экранирование **VPN** на **CISCO ASA** выполнена лучше. Обеспечение удаленного доступа – это одна из функций межсетевых экранов **CISCO ASA** и она не имеет аналогов реализации на маршрутизаторах. Однако есть далеко не полный перечень **функций**, которые, как правило, отсутствуют на межсетевых экранах и **присутствуют на маршрутизаторах**:

- **BGP** – протокол динамической маршрутизации;
- **MPLS** (Multiprotocol Label Switching – многопротокольная коммутация по меткам) – механизм в высокопроизводительной телекоммуникационной сети;
- **DMVPN** (Dynamic Multipoint Virtual Private Network – динамическая многоточечная виртуальная частная сеть) – технология для создания виртуальных частных сетей;
- **GRE** (Generic Routing Encapsulation) – технология осуществления туннелирования пакетов сети;
- **WLAN** (Wireless Local Area Network – беспроводная локальная сеть) **Conroller**.

6.1.3. Классификация межсетевых экранов

Межсетевые экраны представляют собой важный инструмент для обеспечения безопасности сетей, и их классификация может быть проведена на

основе уровня сетевой модели OSI, на котором они функционируют. В зависимости от этого выделяют несколько типов межсетевых экранов, каждый из которых имеет свои особенности, преимущества и недостатки (рис. 6.2).

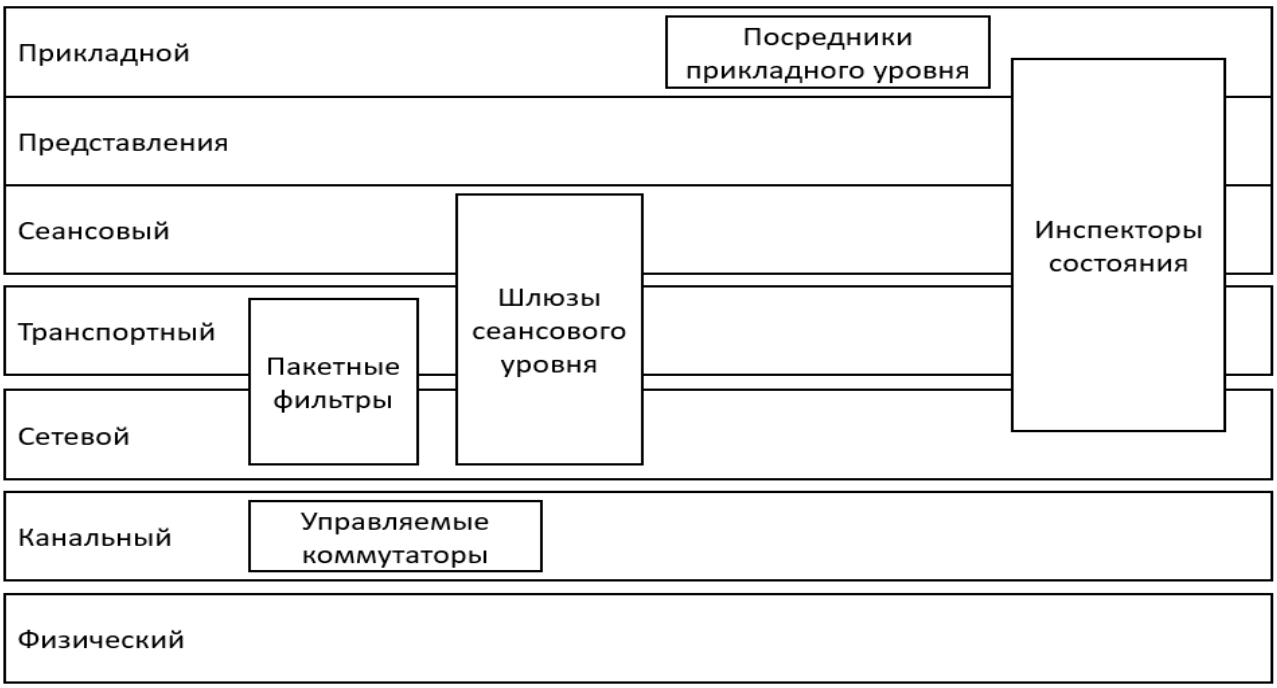

Рис. 6.2. Классификация межсетевых экранов

Рассмотрим более подробно основные категории.

1. Управляемые коммутаторы. Управляемые коммутаторы иногда относят к классу межсетевых экранов, так как они способны фильтровать трафик на канальном уровне модели OSI. Они работают с MAC-адресами и могут разделять трафик внутри локальной сети, используя технологии, такие как VLAN (Virtual Local Area Network). Однако их функциональность ограничена, так как они не способны анализировать трафик на более высоких уровнях, например, на сетевом или транспортном. Это делает их непригодными для защиты от внешних угроз, таких как атаки из Интернета.

Преимущества:

- высокая скорость обработки трафика;
- эффективность для внутренней сегментации сети.

Недостатки:

- неспособность анализировать протоколы выше канального уровня;
- ограниченная защита от сложных атак.

2. Пакетные фильтры. Пакетные фильтры работают на сетевом уровне и анализируют заголовки пакетов, такие как IP-адреса, порты и тип протокола (TCP, UDP). Они являются одним из самых распространенных типов межсетевых экранов и часто встроены в маршрутизаторы. Пакетные фильтры могут блокировать фрагментированные пакеты, которые часто используются в атаках, но их основная слабость заключается в уязвимости к IP-спуфингу (подделке IP-адресов).

Преимущества:

- низкая стоимость;
- минимальная задержка при обработке трафика;
- гибкость в настройке правил фильтрации.

Недостатки:

- неспособность анализировать содержимое пакетов;
- уязвимость к атакам, использующим подделку адресов;
- отсутствие аутентификации на уровне пользователя.

3. Шлюзы сеансового уровня. Шлюзы сеансового уровня действуют как посредники между внутренней и внешней сетями. Они отслеживают состояние активных соединений и пропускают только те пакеты, которые принадлежат установленным сессиям. Это делает их эффективными для предотвращения атак, таких как DoS (отказ в обслуживании). Однако они не анализируют содержимое пакетов.

Преимущества:

- низкая стоимость;
- высокая скорость обработки трафика;
- скрытие топологии внутренней сети.

Недостатки:

- неспособность фильтровать содержимое пакетов;
- ограниченная функциональность для анализа прикладного уровня.

4. Межсетевые экраны прикладного уровня (прокси-брандмауэры). Эти экраны работают на прикладном уровне и анализируют трафик, связанный с конкретными протоколами, такими как HTTP, FTP или SMTP. Они создают два отдельных соединения: одно между клиентом и прокси, другое – между прокси и сервером. Это позволяет им проверять содержимое пакетов, блокировать подозрительные данные и выполнять аутентификацию пользователей.

Преимущества:

- высокий уровень безопасности благодаря анализу содержимого пакетов;
- возможность аутентификации на уровне пользователя;
- скрытие внутренней сети от внешних угроз.

Недостатки:

- более высокая стоимость по сравнению с другими типами;
- ограниченная поддержка протоколов, таких как RPC и UDP;
- снижение производительности из-за глубокого анализа трафика.

5. Инспекторы состояния. Инспекторы состояния сочетают в себе функции пакетных фильтров, шлюзов сеансового уровня и прокси-брандмауэров. Они анализируют трафик на всех уровнях модели OSI, отслеживают состояние соединений и проверяют содержимое пакетов. Это делает их наиболее универсальными и эффективными для защиты сетей.

Преимущества:

- высокая производительность;
- возможность анализа трафика на всех уровнях;
- прозрачность для пользователей;
- гибкость в поддержке новых протоколов и служб.

Недостатки:

- более сложная настройка и управление;
- меньшая защищенность по сравнению с прокси-брандмауэрами на прикладном уровне.

Сравнение типов межсетевых экранов представлено в табл. 6.1.

Таблица 6.1

Сравнение типов межсетевых экранов

Тип экрана	Уровень OSI	Преимущества	Недостатки
Управляемые коммутаторы	Канальный	Высокая скорость, эффективность для внутренней сегментации	Неспособность анализировать протоколы выше канального уровня
Пакетные фильтры	Сетевой	Низкая стоимость, минимальная задержка, гибкость	Уязвимость к IP-спуфингу, отсутствие анализа содержимого пакетов
Шлюзы сеансового уровня	Сеансовый	Низкая стоимость, высокая скорость, скрытие топологии сети	Неспособность анализировать содержимое пакетов
Прокси-брандмауэры	Прикладной	Высокий уровень безопасности, аутентификация пользователя, анализ содержимого	Высокая стоимость, снижение производительности, ограниченная поддержка UDP
Инспекторы состояния	Все уровни	Универсальность, высокая производительность, прозрачность для пользователей	Сложность настройки, меньшая защищенность на прикладном уровне

6.1.4. Варианты реализации сетевых экранов

Межсетевые экраны могут быть реализованы в двух основных формах: **программной** и **программно-аппаратной**. Каждый из этих вариантов имеет свои особенности, преимущества и недостатки, которые определяют их применение в различных сценариях. Рассмотрим их более подробно.

Программные межсетевые экраны

Программные решения представляют собой специализированное программное обеспечение, которое устанавливается на существующие компьютеры или серверы. Такие экраны часто используются в небольших организациях или домашних сетях, где бюджет ограничен, а требования к безопасности не столь высоки.

Процесс внедрения программного экрана представляет собой следующие этапы:

1. Выбор устройства – необходимо выбрать выделенный сервер или обычный компьютер.

2. Установка операционной системы – на выбранный компьютер устанавливается операционная система, например Windows, Linux или FreeBSD.

3. Установка и настройка программного обеспечения – после установки операционной системы настраивается программное обеспечение межсетевого экрана, такое как iptables (для Linux), Windows Firewall или сторонние решения, например pfSense.

Преимущества программных решений:

- **низкая стоимость** – нет необходимости приобретать специализированное оборудование;
- **гибкость** – возможность настройки под конкретные задачи и интеграции с другими программными решениями;
- **простота масштабирования** – при необходимости можно перенести программное обеспечение на более мощное оборудование.

Недостатки программных решений:

- **зависимость от аппаратного обеспечения** – производительность экрана ограничена возможностями компьютера, на котором он установлен;
- **сложность настройки** – требует глубоких знаний сетевых технологий;
- **низкая отказоустойчивость** – в случае сбоя в работе компьютера вся сеть остается без защиты.

Программно-аппаратные межсетевые экраны

Программно-аппаратные решения представляют собой специализированные устройства, которые объединяют в себе аппаратные компоненты и предустановленное программное обеспечение. Такие устройства часто называют **security appliance** (устройства безопасности). Они предназначены для выполнения конкретных задач и обеспечивают высокую производительность и надежность.

Разновидности программно-аппаратных решений:

- **модули в коммутаторах или маршрутизаторах** – некоторые производители сетевого оборудования, такие как Cisco или Juniper, предлагают встраиваемые модули, которые добавляют функциональность межсетевого экрана к уже существующим устройствам;
- **специализированные устройства** – это отдельные устройства, которые предназначены исключительно для выполнения функций межсетевого экрана. Они работают на базе оптимизированных операционных систем, таких как FreeBSD или Linux, и имеют предустановленное программное обеспечение.

Преимущества программно-аппаратных решений:

- **простота внедрения** – устройства поставляются с предустановленной и настроенной операционной системой, что значительно упрощает процесс установки;
- **высокая производительность** – специализированные устройства оптимизированы для выполнения конкретных задач, что позволяет им обрабатывать трафик с минимальными задержками;

• **простота управления** – управление такими устройствами может осуществляться через стандартные протоколы, такие как SNMP, Telnet, SSH или TLS, что позволяет администраторам контролировать их удаленно;

• **отказоустойчивость и высокая доступность** – многие устройства поддерживают функции резервирования и автоматического восстановления, что обеспечивает непрерывную защиту сети.

Недостатки программно-аппаратных решений:

• **высокая стоимость** – специализированные устройства обычно дороже, чем программные решения;

• **ограниченная гибкость** – такие устройства менее гибки в настройке по сравнению с программными решениями;

• **зависимость от производителя** – обновления и поддержка зависят от производителя устройства.

Сравнение программных и программно-аппаратных решений представлено в табл. 6.2.

Таблица 6.2

Сравнение программных и программно-аппаратных решений

Характеристика	Программные экраны	Программно-аппаратные экраны
Стоимость	Низкая	Высокая
Простота внедрения	Требует настройки ОС и ПО	Предустановленная ОС, минимальная настройка
Производительность	Зависит от аппаратного обеспечения	Высокая, оптимизирована для задач
Управление	Требует глубоких знаний	Упрощенное, через стандартные протоколы
Отказоустойчивость	Низкая	Высокая
Гибкость	Высокая	Ограниченная

6.1.5. Межсетевой экран Cisco ASA 5505: особенности и возможности

Межсетевой экран Cisco ASA 5505 – это компактное, но мощное устройство, разработанное для обеспечения безопасности локальных сетей (рис. 6.3 и 6.4).

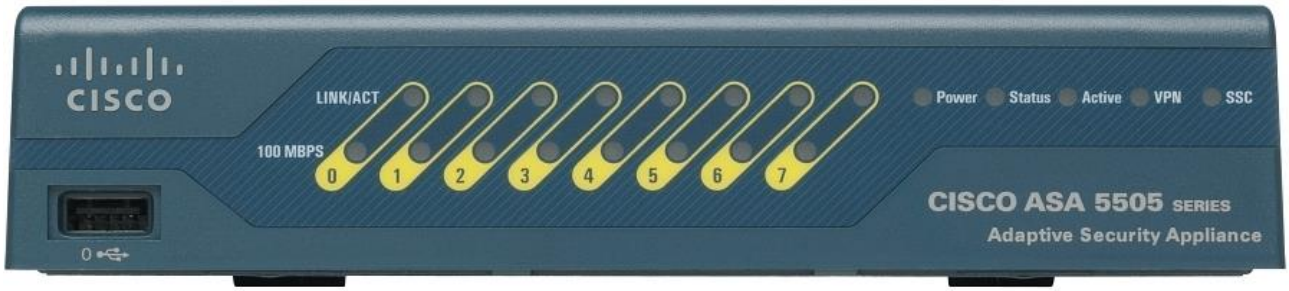

Рис. 6.3. Внешний вид межсетевого экрана Cisco ASA 5505 (вид спереди)

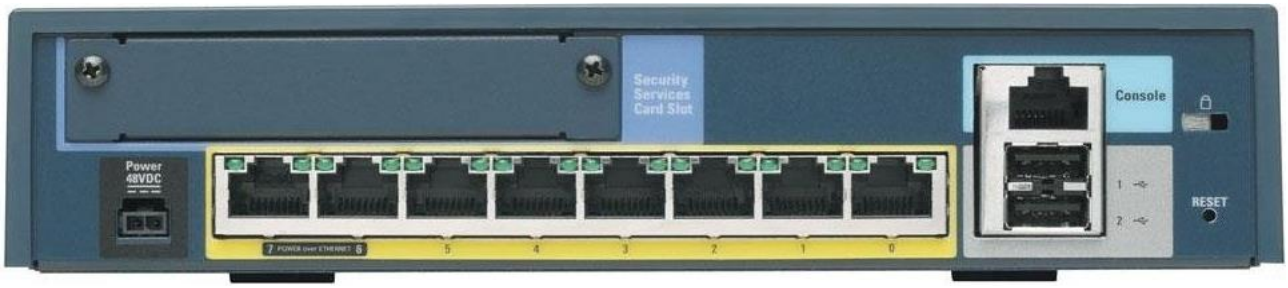

Рис. 6.4. Внешний вид межсетевого экрана Cisco ASA 5505 (вид сзади)

Оно сочетает в себе высокую производительность, гибкость и простоту настройки, что делает его идеальным решением для малого бизнеса, филиалов компаний и удаленных сотрудников. Cisco ASA 5505 предоставляет комплексную защиту от внешних угроз, таких как вирусы, спам, шпионские программы, а также обеспечивает фильтрацию трафика и предотвращение вторжений.

Cisco ASA 5505 представляет собой многофункциональное устройство, которое выполняет следующие задачи:

- **защита сети** – предотвращение атак, блокирование вредоносного трафика и фильтрация контента;
- **организация VPN** – поддержка SSL и IPsec VPN для безопасного удаленного доступа;
- **сегментация сети** – возможность создания виртуальных сетей (VLAN) для повышения безопасности и управления трафиком;
- **поддержка VoIP и беспроводных технологий** – наличие портов Power over Ethernet (PoE) для подключения IP-телефонов и точек доступа.

Cisco ASA 5505 выполнен в компактном корпусе, который не предназначен для установки в 19-дюймовые стойки. Это делает устройство удобным для использования в небольших офисах или удаленных локациях.

Основные аппаратные характеристики:

- **сетевые интерфейсы** – восемь портов 10/100 Мбит/с, которые могут быть сгруппированы для создания до трех виртуальных сетей (VLAN);
- **порты PoE** – два порта с поддержкой Power over Ethernet, что упрощает подключение IP-телефонов и беспроводных точек доступа;
- **USB-порты** – три порта USB 2.0 для подключения дополнительных устройств и расширения функциональности;
- **производительность** – поддержка до 10 000 одновременных сессий, скорость межсетевого экранирования до 150 Мбит/с и VPN-шифрование до 100 Мбит/с.

Cisco ASA 5505 оснащен встроенным программным обеспечением, которое обеспечивает простоту настройки и управления. Ключевые программные функции:

- **Cisco ASDM (Adaptive Security Device Manager)** – графический интерфейс для настройки и управления устройством. ASDM позволяет

администраторам быстро развертывать и настраивать межсетевой экран, минимизируя затраты на внедрение;

- **поддержка VPN** – устройство поддерживает SSL и IPsec VPN, что обеспечивает безопасный доступ удаленных пользователей к корпоративным ресурсам;
- **система предотвращения вторжений (IPS)** – после установки дополнительного модуля AIP SSC (Advanced Inspection and Prevention Security Services Card) устройство получает возможность блокировать сетевые черви и предотвращать атаки;
- **гибкость настройки** – возможность динамической группировки портов для создания VLAN, что позволяет сегментировать сеть и повышать ее безопасность.

6.1.6. Уровни безопасности в межсетевых экранах Cisco ASA

Межсетевые экраны Cisco ASA используют концепцию уровней безопасности (Security Level) для управления доступом между различными зонами сети. Уровень безопасности представляет собой числовое значение от 0 до 100, которое определяет степень доверия к интерфейсу или сети, подключенной к этому интерфейсу. Чем выше уровень безопасности, тем более защищенной считается зона. Эта система позволяет администраторам гибко настраивать правила доступа и обеспечивать безопасность сети.

Каждому интерфейсу межсетевого экрана Cisco ASA, будь то физический или логический (субинтерфейс), может быть назначен определенный уровень безопасности. Основное правило заключается в том, что трафик может свободно проходить от интерфейса с более высоким уровнем безопасности к интерфейсу с более низким уровнем. Однако обратный доступ (от низкого уровня к высокому) требует явного разрешения через правила безопасности, такие как ACL (Access Control List).

Уровни безопасности и их назначение:

1. Уровень безопасности 0 (Security Level 0). Это минимальный уровень безопасности, который обычно назначается интерфейсу, подключенному к внешней сети, например к Интернету. Такой интерфейс называется Outside. Устройства, подключенные к этому интерфейсу, не имеют доступа к внутренним сетям, если это не разрешено явно через правила ACL (интерфейс, подключенный к провайдеру Интернета).

2. Уровни безопасности от 1 до 99 (Security Level 1–99). Эти уровни назначаются промежуточным зонам, таким как DMZ (демилитаризованная зона), зона управления и серверная зона. DMZ, например, часто используется для размещения общедоступных сервисов, таких как веб-серверы или почтовые серверы, которые должны быть доступны из внешней сети, но при этом изолированы от внутренней корпоративной сети (интерфейс, подключенный к DMZ, может иметь уровень безопасности 50).

3. Уровень безопасности 100 (Security Level 100). Это максимальный уровень безопасности, который назначается интерфейсу, подключенному к внутренней корпоративной сети. Такой интерфейс называется Inside. Он считается наиболее доверенным, и трафик из этой зоны может свободно проходить в зоны с более низкими уровнями безопасности (интерфейс, подключенный к локальной сети офиса).

Рассмотрим типичный сценарий, в котором межсетевой экран Cisco ASA разделяет сеть на три зоны (рис. 6.5):

- **внутренняя (локальная) сеть (Inside)** – уровень безопасности 100. Это наиболее защищенная зона, где находятся корпоративные данные и ресурсы. Трафик из этой зоны может свободно проходить в DMZ и внешнюю сеть;
- **DMZ (демилитаризованная зона)** – уровень безопасности 50. В этой зоне размещаются общедоступные сервисы, такие как веб-серверы. Трафик из DMZ может проходить только во внешнюю сеть, но не во внутреннюю;
- **внешняя сеть (Outside)** – уровень безопасности 0. Это зона с наименьшим уровнем доверия, например Интернет. Трафик из этой зоны не может проникать во внутреннюю сеть или DMZ без явного разрешения.

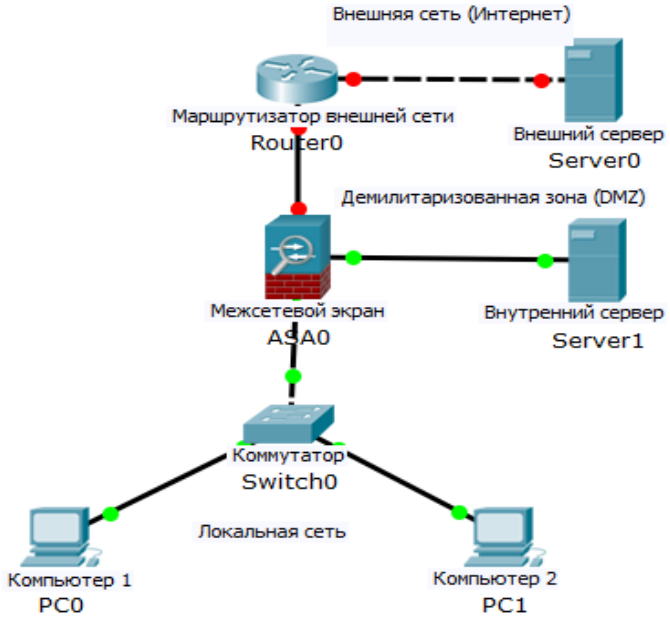

Рис. 6.5. Пример назначения уровней безопасности в сети

Таким образом, межсетевой экран на сегодняшний день является практически обязательным элементом обеспечения компьютера, подключенного к Интернету. Без него вероятность несанкционированного проникновения на компьютер, попадания на него вирусов, троянов, кражи конфиденциальной информации не просто велика, а близка к 100 %.

6.2. Порядок и основные правила выполнения заданий

Для изучения принципов работы и настройки межсетевого экрана **Cisco** будем использовать **симулятор Cisco Packet Tracer**.

Порядок и основные правила выполнения заданий рассмотрим в ходе выполнения конкретного примера – создания простой локальной сети с использованием сетевого экрана и ее соединения с сетью интернет-провайдера.

6.2.1. Построение структурной схемы сети

Запустим **программу Cisco Packet Tracer**. В области **«Логическое пространство»** построим структурную схему общей сети в составе: два компьютера (PC-PT), сервер (Server-PT), маршрутизатор (1841) и межсетевой экран ASA 5505 (ASA0).

Структурная схема сети представлена на рис. 6.6.

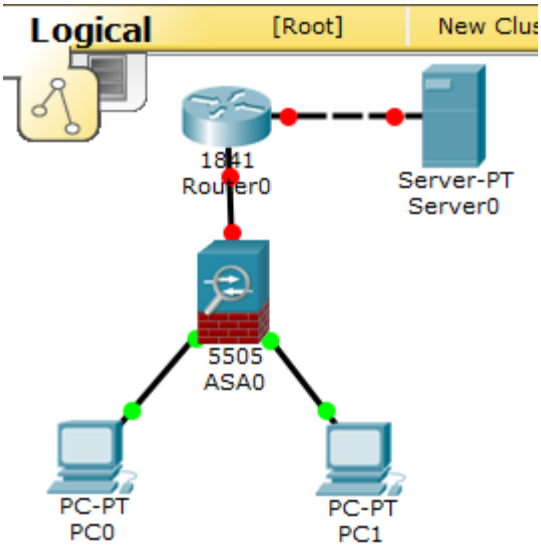

Рис. 6.6. Структурная схема сети

Следует отметить, что выбор межсетевого экрана **Cisco ASA 5505** связан с отсутствием других вариантов межсетевых экранов.

6.2.2. Рассмотрение технических и лицензионных возможностей и предустановок межсетевого экрана Cisco ASA 5505 (ASA0)

Для решения этой задачи воспользуется **Cisco ASA Command Line Interface межсетевого экрана ASA0**. С этой целью путем нажатия левой кнопки мыши открываем **ASA IOS Command Line Interface**.

Войдем в привилегированный режим. При этом, когда в интерфейсе командной строки потребуется использование пароля, необходимо нажать клавишу **Enter**.

ciscoasa>**enable**

Password: (нажатие клавиши Enter)

ciscoasa#

Для начала при помощи команды **Show Version** целесообразно посмотреть возможности **Cisco ASA 5505**, которые представлены в **Cisco Packet Tracer** (рис. 6.7).

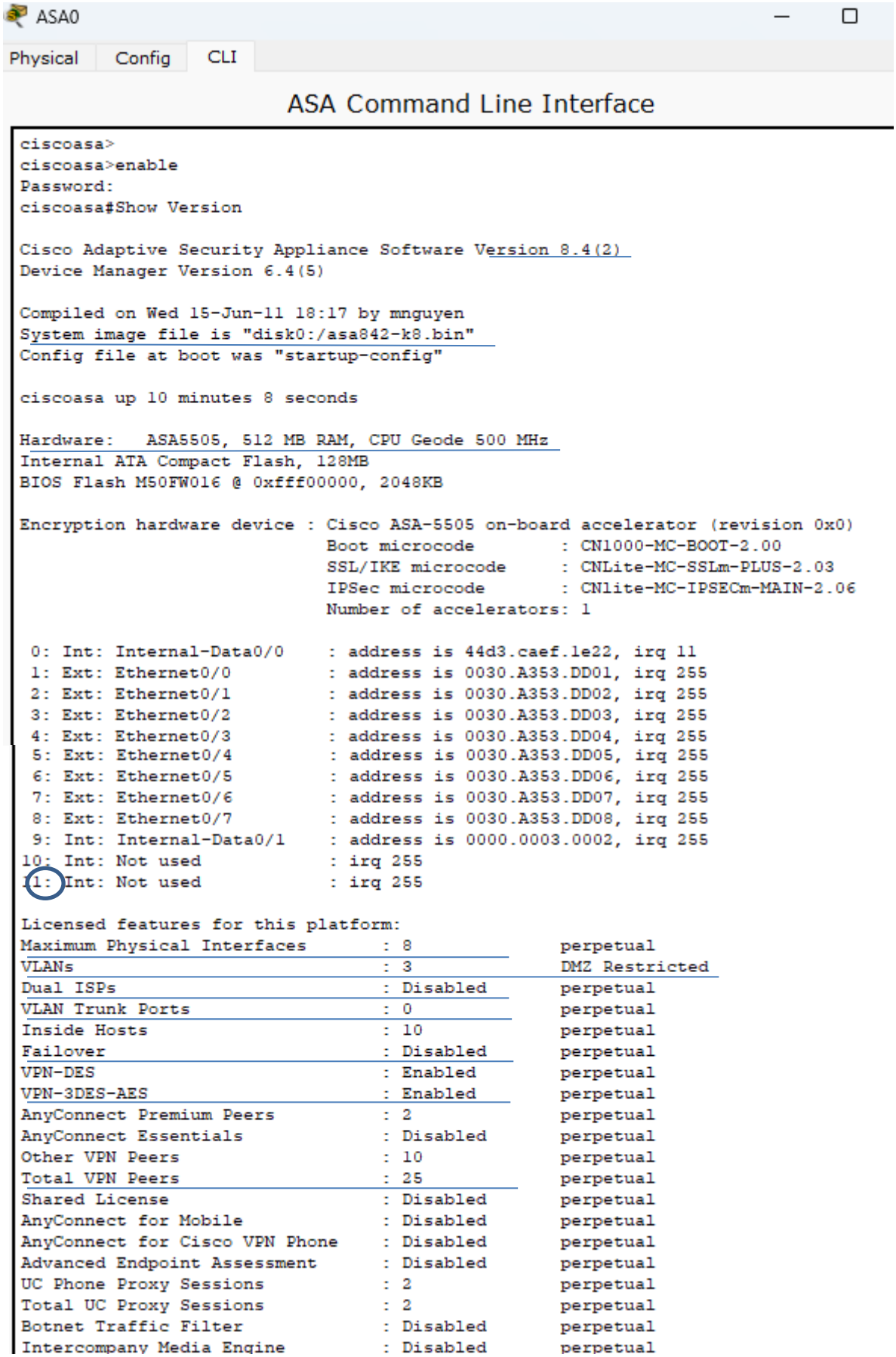

Рис. 6.7. Технические и лицензионные возможности Cisco ASA 5505

На рис. 6.7 мы можем отметить ряд сведений, которые дают нам представление о важных технических и лицензионных возможностях **Cisco ASA 5505**:

- **версия диспетчера устройств** программного обеспечения межсетевого экрана – 8.4 (2);
- **файл прошивки** системы – disk0:/asa842-k8.bin;
- **аппаратное обеспечение** – ASA5505, 512 MB RAM, CPU Geoge 500 MHz;
- **количество портов** – 11;
- максимальное **количество физических интерфейсов** – 8;
- максимальное **количество VLAN** – 3, из них третий DMZ;
- **Dual ISPs** (взаимодействие с двумя провайдерами) – неработоспособно;
- **VLAN магистрального порта** – 0;
- **аварийное переподключение** (Failover) – неработоспособно;
- **алгоритм шифрования** – VPN-3DES-AES;
- общее **количество VPN-узлов** – 25.

Из изложенного следует, что имеющийся в **Cisco Packet Tracer** межсетевой экран **Cisco ASA 5505** обладает достаточно ограниченными возможностями.

Тем не менее для поставленной цели занятия имеющихся функций межсетевого экрана **Cisco ASA 5505** вполне достаточно. Поэтому продолжим его настройку. Целесообразно при помощи команды **Show Run** посмотреть имеющиеся предустановки на межсетевом экране (рис. 6.8).

ASA Command Line Interface

```
ciscoasa#show run
: Saved
:
ASA Version 8.4(2)
!
hostname ciscoasa
names
!
interface Ethernet0/0
 switchport access vlan 2
!
interface Ethernet0/1
!
interface Ethernet0/2
!
interface Ethernet0/3
!
interface Ethernet0/4
!
interface Ethernet0/5
!
interface Ethernet0/6
!
interface Ethernet0/7
!
interface Vlan1
 nameif inside
 security-level 100
 ip address 192.168.1.1 255.255.255.0
interface Vlan2
 nameif outside
 security-level 0
 ip address dhcp
```

Рис. 6.8. Просмотр предварительных настроек ASA0

На рис. 6.8 мы видим, что по умолчанию порт Ethernet0/0 (интернет-провайдера) уже настроен на VLAN2. Остальные порты, соответственно, настроены на VLAN1.

Кроме того, необходимо внимательно рассмотреть необходимые нам настройки **Nameif inside** (название внутреннего интерфейса) и **Nameif outside** (название внешнего интерфейса).

Также мы видим, что здесь по умолчанию настроен **DHCP-сервер**, который будет назначать адреса компьютерам сети.

Подобная преднастройка межсетевого экрана по своим возможностям соответствует коммутатору третьего уровня.

6.2.3. Проверка настройки IP-адресов компьютеров

Откроем на компьютере **PC0** вкладки **Desktop → IP-Configuration** и включим **DHCP**. При помощи этого протокола **PC0** получил **IP-адрес – 192.168.1.5** (рис. 6.9).

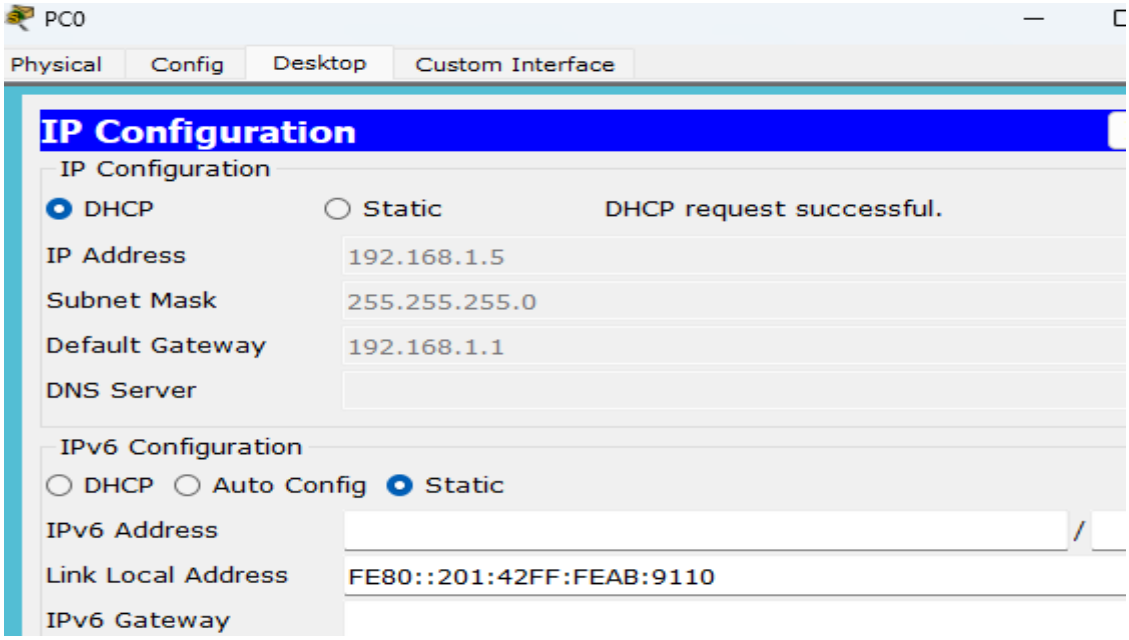

Рис. 6.9. Назначение IP-адреса компьютеру PC0 протоколом DHCP

Аналогичные действия проведем для компьютера PC1. В результате мы получаем для PC1 IP-адрес – 192.168.1.6.

6.2.4. Настройка удаленного доступа к Cisco ASA 5505

Предположим, что на одном из компьютеров администратор сети хочет удаленно администрировать ASA0. Для этого на ASA0 в режиме глобального конфигурирования выполним ряд последовательных действий, а затем посмотрим, что у нас получилось при помощи команды **Show run**.

В отличие от коммутатора и маршрутизатора в межсетевом экране автоматическая зашифровка паролей – это одна из многих функций безопасности по умолчанию (рис. 6.10).

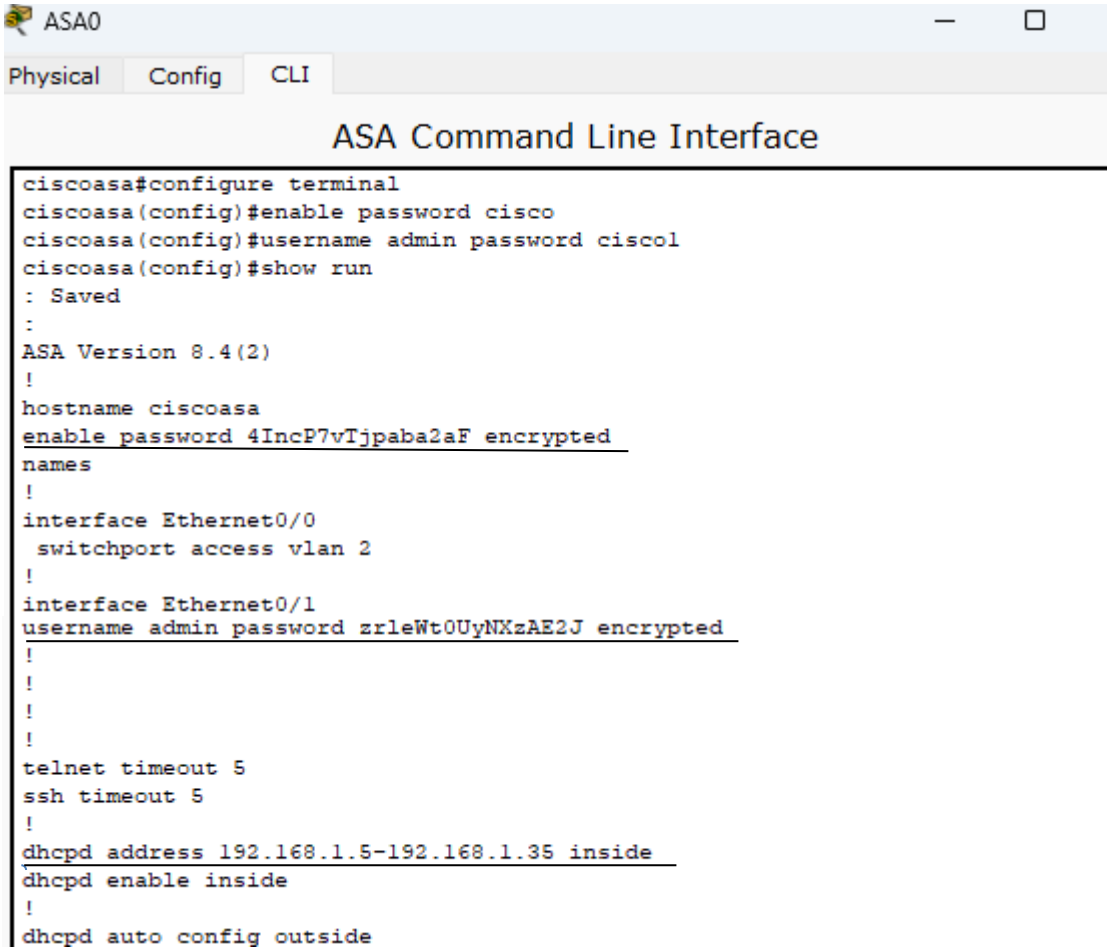

Рис. 6.10. Зашифрованные пароли ASA0 и администратора

Для удаленного подключения нам нужно выбрать прикладной протокол, который позволит управлять операционной системой удаленно, а также осуществлять скрытие информации путем туннелирования с применением протокола TCP. Как правило, в этом случае имеется возможность использования протокола **SSH** (Secure Shell – безопасная оболочка). Этот протокол вполне подходит нам с точки зрения обеспечения безопасности (рис. 6.11).

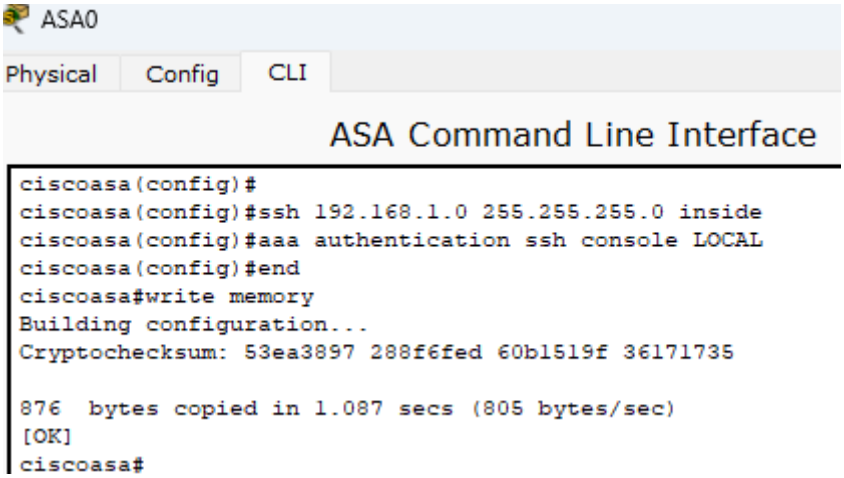

Рис. 6.11. Настройка удаленного доступа к ASA0

Мы применили его для локальной сети с **IP-адресом 192.168.1.0**. Также задали параметры аутентификации пользователя (администратора), где подтвердили использование локальной базы.

Проверим доступ с компьютера PC0. Для этого выберем вкладку **Command Promt** и там наберем команду **ssh -l admin 192.168.1.1** (здесь после **ssh** и дефиса стоит английское эль малое).

PC>**ssh -l admin 192.168.1.1**
Password:**cisco1**
ciscoasa>**enable**
Password:**cisco**
ciscoasa#

При наборе пароля администратора – **cisco1** – и пароля входа в компьютер – **cisco** – буквы на экране не высвечиваются. После ввода паролей установился удаленный доступ к межсетевому экрану. Наберем команду **Show run**, например с компьютера PC1. Ее работа подтверждает установление удаленного соединения.

6.2.5. Настройка уровня безопасности (Security Level) межсетевого экрана

Уровни безопасности при необходимости можно изменять. Так, например, уменьшим уровень безопасности **VLAN1** (внутренней локальной сети) со 100 до 95, набрав ряд команд (рис 6.12), в чем убедимся при помощи команды **Show run**.

ASA Command Line Interface

```
ciscoasa#configure terminal
ciscoasa(config)#interface vlan1
ciscoasa(config-if)#security-level 95
ciscoasa(config-if)#end
ciscoasa#show run
: Saved
:
ASA Version 8.4(2)
!
hostname ciscoasa
enable password 4IncP7vTjpaba2aF encrypted
names
!
interface Ethernet0/0
 switchport access vlan 2
!
interface Ethernet0/1
!
interface Ethernet0/2
!
interface Ethernet0/3
!
interface Vlan1
 nameif inside
 security-level 95
 ip address 192.168.1.1 255.255.255.0
!
interface Vlan2
 nameif outside
 security-level 0
 ip address dhcp
```

Рис. 6.12. Результат настройки на снижение уровня безопасности VLAN1

Одновременно на рис. 6.12 мы можем видеть, что внешний интерфейс межсетевого экрана не настроен. Выполним данную настройку с учетом выданного интернет-провайдером общедоступного **IP-адреса – 210.210.0.2** и после этого наберем команду **Show run** (рис. 6.13).

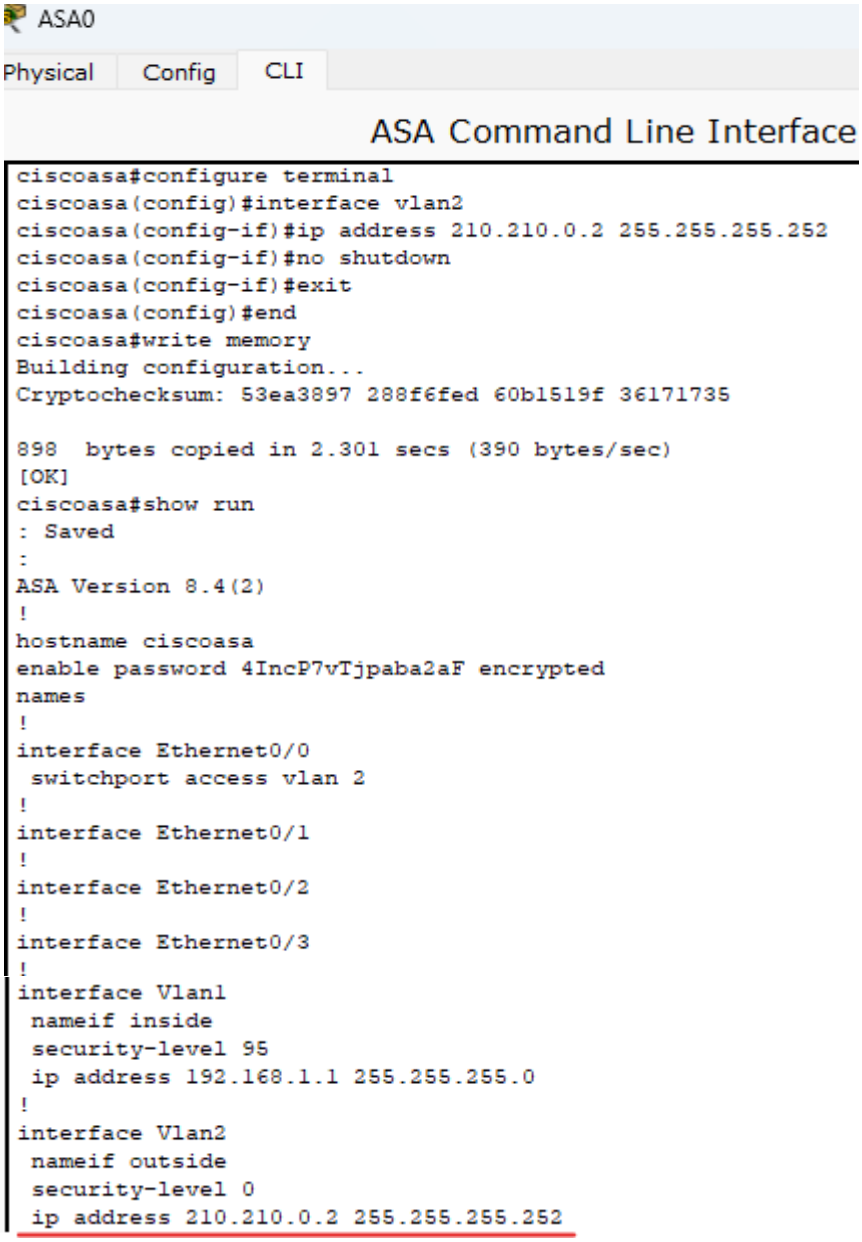

Рис. 6.13. Настройка внешнего интерфейса межсетевого экрана

На рис. 6.13 уже представлена информация с **IP-адресом VLAN2**.

6.2.6. Настройка интерфейсов маршрутизатора (Router0)

Как мы уже знаем, у маршрутизаторов, так же как и у межсетевых экранов, порты по умолчанию находятся в режиме **down**. Поэтому необходимо в первую очередь поднять физический порт маршрутизатора, которым в нашем случае является **Fa0/0**. С этой целью при помощи команд выполняем ряд последовательных действий (рис. 6.14).

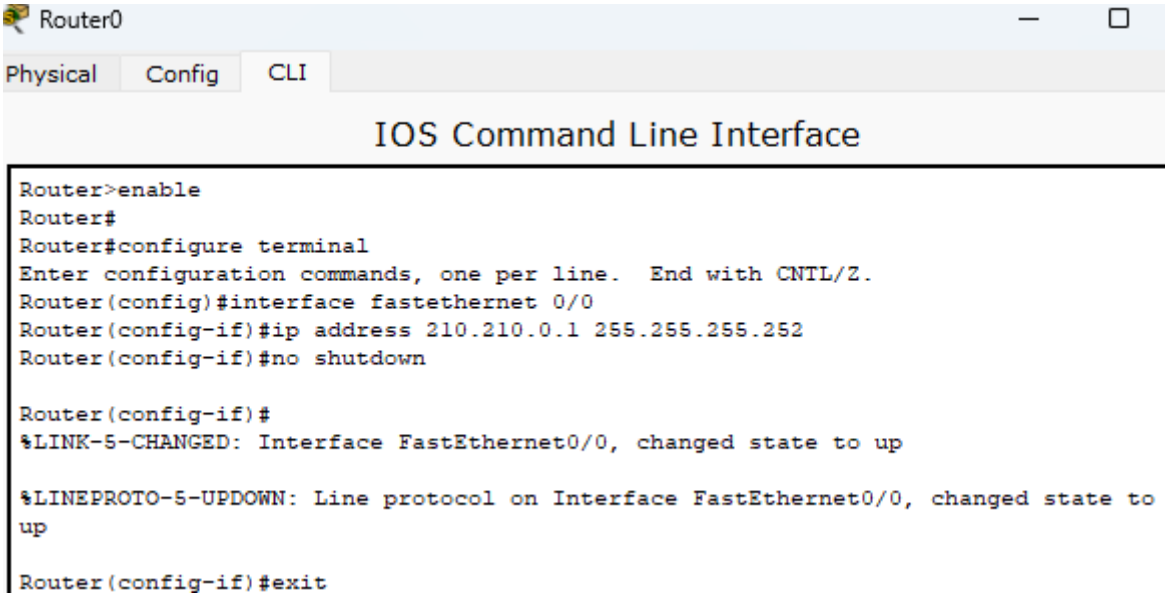

Рис. 6.14. Настройка интерфейса маршрутизатора
со стороны межсетевого экрана

И сразу настроим интерфейс маршрутизатора **Fa0/1** со стороны сервера (рис. 6.15).

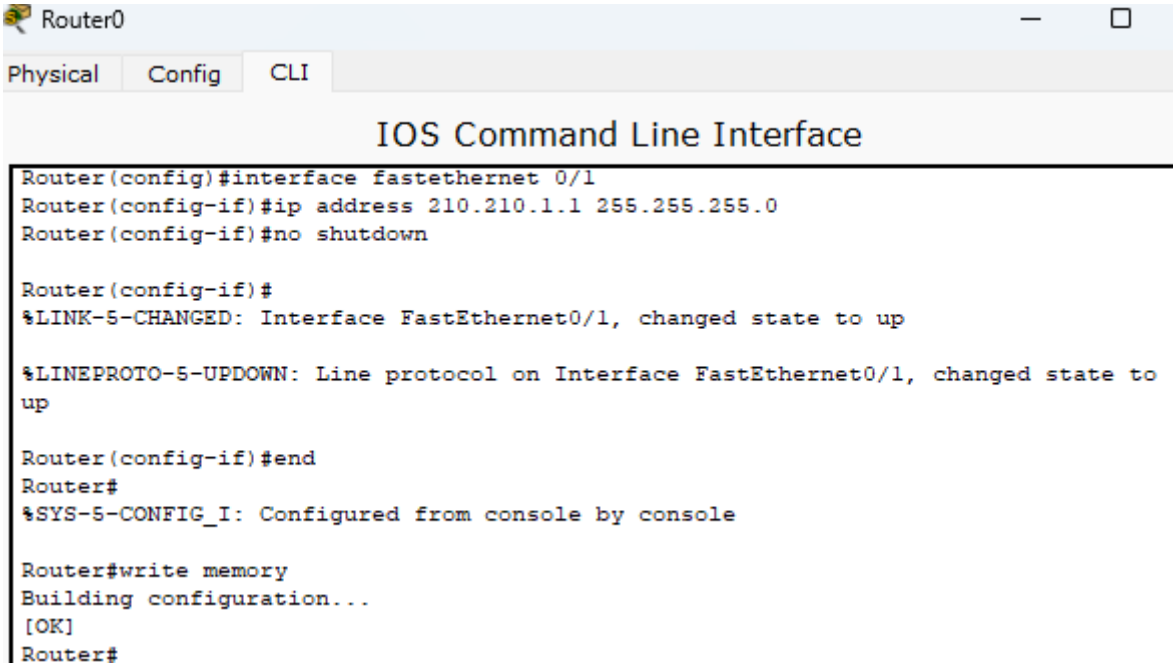

Рис. 6.15. Настройка интерфейса маршрутизатора со стороны сервера

Как видим на рис. 6.15, настройка порта **Fa0/1** маршрутизатора прошла успешно.

Для того чтобы убедиться в правильности настройки соединения межсетевого экрана с маршрутизатором провайдера, проверим прохождение утилиты **Ping** между этими двумя устройствами. Утилита прошла успешно (рис. 6.16).

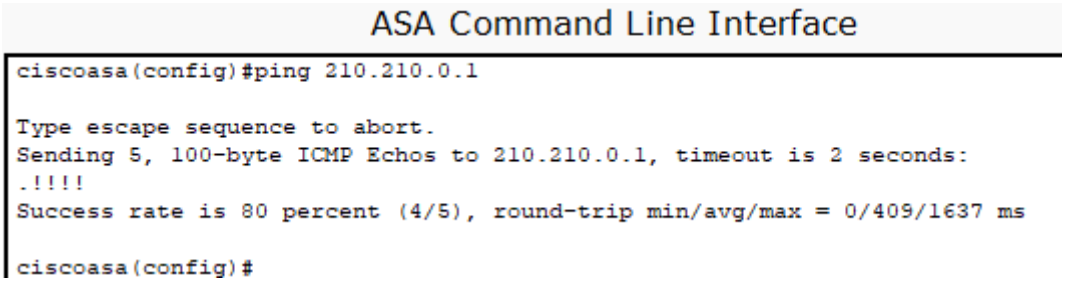

Рис. 6.16. Результаты прохождения утилиты Ping к маршрутизатору

Кроме всего прочего, на маршрутизаторе должен быть прописан маршрут в локальную сеть (6.17).

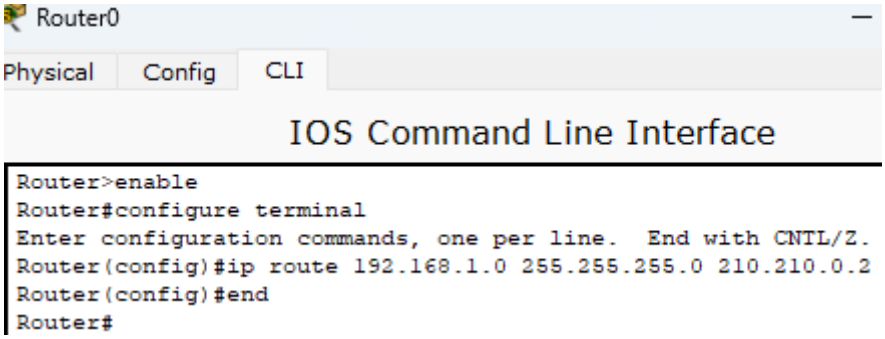

Рис. 6.17. Определение маршрута от маршрутизатора в локальную сеть

6.2.7. Настройка сервера (Server0)

Откроем на сервере **Server0** вкладки **Desktop → IP-Configuration** и настроим IP-адрес – 210.210.1.2 с маской 255.255.255.0 и IP-адрес Default Gateway – 210.210.1.1 (рис. 6.18).

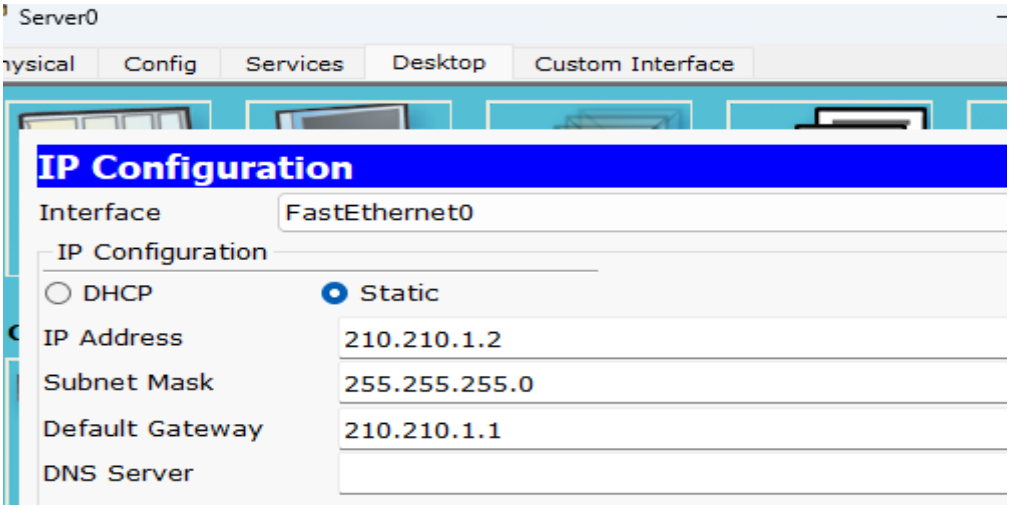

Рис. 6.18. Конфигурирование сервера

6.2.8. Настройка маршрута из локальной сети к серверу по умолчанию

Поскольку маршрутизатор является устройством интернет-провайдера, то маршрут по умолчанию определим через него. С этой целью наберем последовательный ряд команд (рис. 6.19).

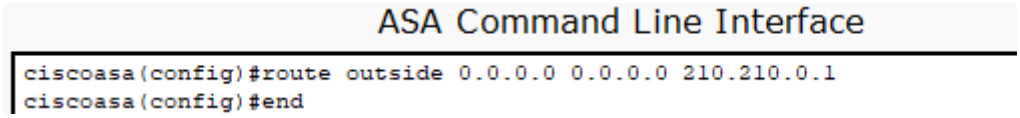

Рис. 6.19. Настройка маршрута из локальной сети к серверу

Проверим прохождение утилиты **Ping** от межсетевого экрана к серверу. Утилита прошла успешно (рис. 6.20).

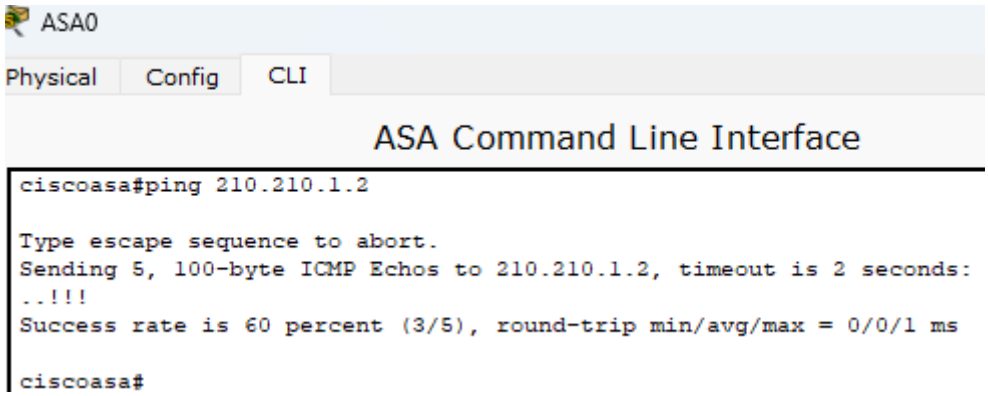

Рис. 6.20. Результаты прохождения утилиты Ping к серверу

6.2.9. Настройка межсетевого экрана на инспектирование трафика (Stateful Packet Inspection)

Кроме отслеживания адреса источника и получателя, межсетевой экран с функцией **SPI** (Stateful Packet Inspection – инспектирование трафика) проверяет контекст пакетов данных. Он работает с обоими протоколами – TCP и UDP. В случае UDP, который передает пакеты данных без получения подтверждения, межсетевой экран следует конфигурации (например, допустимая продолжительность сеанса).

Межсетевой экран с SPI использует собственную память для хранения данных о предыдущих пакетах и **таблицу состояний** соединений. Это позволяет ему определять, принадлежит ли входящий пакет к установленному сеансу связи. Например, он отслеживает, какое приложение инициировало подключение к Интернету, и ожидает ответ от веб-сервера. Если пакет не соответствует ожидаемому состоянию сеанса, **межсетевой экран с SPI** его отбрасывает.

У сетевого экрана Cisco ASA 5505 инспектирование трафика по умолчанию не настроено, поэтому нам предстоит это выполнить. Для этого необходимо войти в режим глобального конфигурирования и определить так называемый **Class Map**, т. е. тип трафика, который мы хотим исследовать. И здесь укажем, что будем использовать весь трафик, который проходит через межсетевой экран.

Дальше мы создаем политику – это наши действия над трафиком. Причем это действие применяется к созданному нами классу. Нас, конечно же, интересуют протоколы, которые подлежат инспектированию.

И в заключение определим, в каком направлении будем делать инспектирование. Остановимся на **Global Policy**, что означает **во всех направлениях**.

Для реализации всех указанных действий применим ряд команд, которые представлены на рис. 6.21.

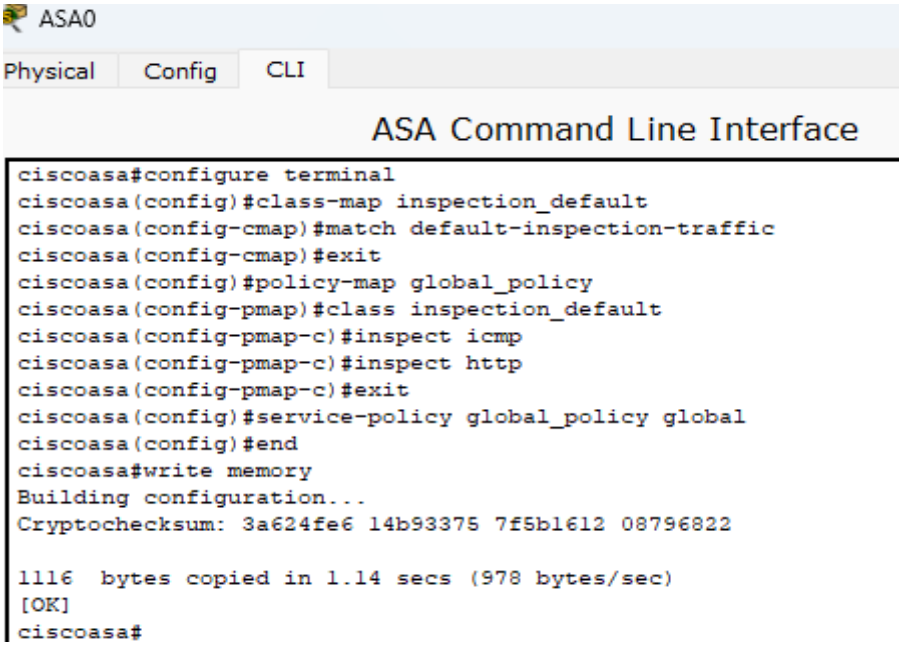

Рис. 6.21. Настройка межсетевого экрана с функцией SPI

Проверка прохождения утилиты **Ping** от компьютера PC0 к серверу Server0 подтвердила правильность построения и настройки сети.

Создадим также в веб-браузере URL-запрос на сервер **Server0**. Выполнение запроса подтверждено (рис. 6.22).

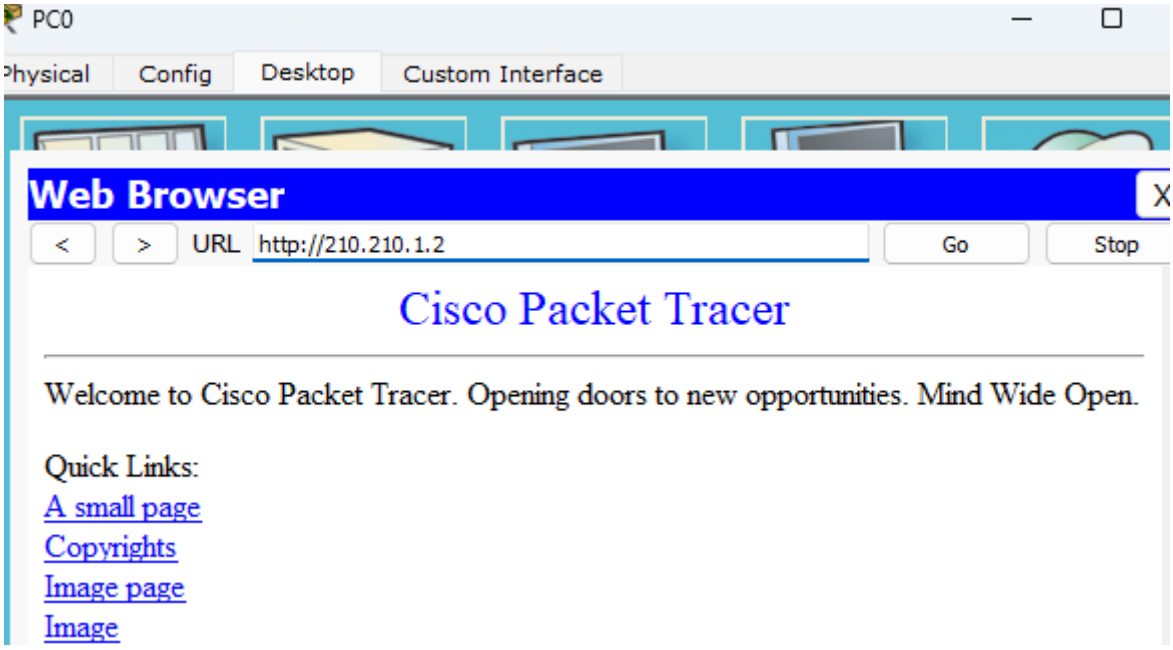

Рис. 6.22. Успешное прохождение URL-запроса

Прохождение утилиты **Ping** от сервера **Server0** к компьютеру **PC0** не представляется возможным, так как на внешнем интерфейсе **Security Level** равен нулю и межсетевой экран ее не пропускает, а следовательно, сеть настроена правильно.

6.3. Практическое задание

6.3.1. Изучить сведения, изложенные в теоретических материалах данного практического занятия и, при необходимости, в дополнительных материалах.

6.3.2. С целью выполнения настройки устройств и организации их взаимодействия с учетом требований межсетевого экрана необходимо выполнить следующие действия.

6.3.2.1. Построить структурную схему сети, состоящую из сетевого оборудования и устройств с исходными данными, согласно указанному преподавателем номеру варианта задания, приведенному в табл. 6.3.

6.3.2.2. Рассмотреть и проанализировать технические и лицензионные возможности и предустановки межсетевого экрана Cisco ASA 5505.

6.3.2.3. Настроить удаленный доступ к Cisco ASA 5505.

6.3.2.4. Настроить IP-адреса компьютеров и проверить установление удаленного доступа к межсетевому экрану с компьютеров сети.

6.3.2.5. Установить уровни безопасности (Security Level) межсетевого экрана.

6.3.2.6. Настроить маршрутизатор интернет-провайдера.

6.3.2.7. Провести конфигурирование общедоступного сервера.

6.3.2.8. Определить маршрут от локальной сети к серверу по умолчанию.

6.3.2.9. Настроить межсетевой экран на инспектирование трафика (Stateful Packet Inspection).

6.3.2.10. Проконтролировать настройку соединения в сети.

6.3.3. Отчет о проделанной работе представить преподавателю в действии непосредственно на компьютере или в виде сканирования ключевых эпизодов выполнения работы по п. 6.3.2.

6.3.4. Ответить на контрольные и дополнительные вопросы.

Варианты заданий

Таблица 6.3

Перечень вариантов заданий

Вариант	IP-адреса и пароли	Вариант	IP-адреса и пароли
1	1. IP-адрес общедоступной сети – 210.210.10.0. 2. Пароль пользователя – akipe1. 3. Пароль входа в ASA – fire1	8	1. IP-адрес общедоступной сети – 210.210.80.0. 2. Пароль пользователя – akipe8. 3. Пароль входа в ASA – fire8
2	1. IP-адрес общедоступной сети – 210.210.20.0. 2. Пароль пользователя – akipe2. 3. Пароль входа в ASA – fire2	9	1. IP-адрес общедоступной сети – 210.210.90.0. 2. Пароль пользователя – akipe9. 3. Пароль входа в ASA – fire9

Окончание табл. 6.3

Вариант	IP-адреса и пароли	Вариант	IP-адреса и пароли
3	1. IP-адрес общедоступной сети – 210.210.30.0. 2. Пароль пользователя – akipe3. 3. Пароль входа в ASA – fire3	10	1. IP-адрес общедоступной сети – 210.210.100.0. 2. Пароль пользователя – akipe10. 3. Пароль входа в ASA – fire10
4	1. IP-адрес общедоступной сети – 210.210.40.0. 2. Пароль пользователя – akipe4. 3. Пароль входа в ASA – fire4	11	1. IP-адрес общедоступной сети – 210.210.110.0. 2. Пароль пользователя – akipe11. 3. Пароль входа в ASA – fire11
5	1. IP-адрес общедоступной сети – 210.210.50.0. 2. Пароль пользователя – akipe5. 3. Пароль входа в ASA – fire5	12	1. IP-адрес общедоступной сети – 210.210.120.0. 2. Пароль пользователя – akipe12. 3. Пароль входа в ASA – fire12
6	1. IP-адрес общедоступной сети – 210.210.60.0. 2. Пароль пользователя – akipe6. 3. Пароль входа в ASA – fire6	13	1. IP-адрес общедоступной сети – 210.210.130.0. 2. Пароль пользователя – akipe13. 3. Пароль входа в ASA – fire13
7	1. IP-адрес общедоступной сети – 210.210.70.0. 2. Пароль пользователя – akipe7. 3. Пароль входа в ASA – fire7	14	1. IP-адрес общедоступной сети – 210.210.140.0. 2. Пароль пользователя – akipe14. 3. Пароль входа в ASA – fire14

Контрольные вопросы

1. Что представляет собой межсетевой экран?
2. Для чего предназначен межсетевой экран?
3. Как работает межсетевой экран?
4. Чем отличается межсетевой экран от маршрутизатора?
5. Чем отличается межсетевой экран от коммутатора?
6. Приведите классификацию межсетевых экранов.
7. Для чего нужны уровни безопасности межсетевого экрана?
8. Как работают уровни безопасности межсетевого экрана?
9. По каким параметрам осуществляется фильтрация трафика межсетевым экраном?
10. Что представляют собой пакетные фильтры?
11. Назовите основные характеристики межсетевого экрана Cisco ASA 5505.
12. Что такое демилитаризованная зона?
13. Каковы варианты реализации межсетевых экранов?
14. Назовите основные правила уровней безопасности межсетевых экранов.
15. Опишите шлюзы сетевого уровня.
16. Какие бывают управляемые коммутаторы?
17. Опишите посредников прикладного уровня.
18. Какие бывают инспекторы состояния?
19. С чем связана ограниченность анализа межсетевого экрана?
20. Сравните межсетевой экран с маршрутизатором.

Список рекомендованной литературы

1. Олифер, В. Г. Компьютерные сети: принципы, технологии, протоколы : учеб. пособие для студентов вузов / В. Г. Олифер, Н. А. Олифер. – Юбилейное изд. доп., испр. – СПб. : Питер, 2024. – 1008 с. : ил.

2. Таненбаум, Э. Компьютерные сети / Э. Таненбаум, Д. Уэзеролл, Н. Фимстер. – 6-е изд. – СПб. : Питер, 2024. – 992 с. : ил.

3. Левашов, П. Ю. Киберкрепость : всестороннее руководство по компьютерной безопасности / П. Ю. Левашов. – СПб. : Питер, 2024. – 544 с.

4. Букатов, А. А. Компьютерные сети: расширенный начальный курс : учеб. пособие / А. А. Букатов, С. А. Гуда. – СПб. : Питер, 2020. – 496 с.

5. Глецевич, И. И. Администрирование компьютерных систем и сетей : учеб.-метод. пособие / И. И. Глецевич. – Минск : БГУИР, 2021. – 80 с.

Св. план 2025, поз. 23

Учебное издание

Балтрукович Петр Иванович
Тумилович Мирослав Викторович
Медведев Олег Сергеевич

ПРИКЛАДНЫЕ ЗАДАЧИ ПРИМЕНЕНИЯ СЕТЕВЫХ ТЕХНОЛОГИЙ. ПРАКТИКУМ

УЧЕБНО-МЕТОДИЧЕСКОЕ ПОСОБИЕ

Редактор *А. Ю. Шурко*
Корректор *Е. Н. Батурчик*
Компьютерная правка, оригинал-макет *В. А. Долгая*

Подписано в печать 26.01.2026. Формат 60×84 1/16. Бумага офсетная. Гарнитура «Таймс».
Отпечатано на ризографе. Усл. печ. л. 7,21. Уч.-изд. л. 7,5. Тираж 90 экз. Заказ 138.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники».
Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий №1/238 от 24.03.2014,
№2/113 от 07.04.2014, №3/615 от 07.04.2014.
Ул. П. Бровки, 6, 220013, г. Минск