МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И
МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ
Северо-Кавказский филиал ордена Трудового Красного Знамени
Федерального государственного бюджетного образовательного
учреждения высшего образования
«Московский технический университет связи и информатики»

Методическое пособие для практических занятий по дисциплине «Построение защищенных мультисервисных сетей связи»

(направление подготовки 11.03.02 «Инфокоммуникационные технологии и системы связи», профиль «Защищенные инфокоммуникационные системы»)

Ростов-на-Дону
2022

Методическое пособие для практических занятий

по дисциплине

«Построение защищенных мультисервисных сетей связи»

Составитель: Доц. кафедры ИТСС, к.т.н., доцент Манин А.А.

Рассмотрено и одобрено на заседании кафедры ИТСС

Протокол № 5 от 19.12.2022

Практическое занятие №1

Построение городской мультисервисной сети на принципах NGN

1.1 Цель работы: Изучить принципы модернизации ГТС для перевода её в городскую мультисервисную сеть с технологиями NGN. Освоить методику поэтапной модернизации ГТС.

1.2 Перечень оборудования:

- Компьютерный класс.

1.3 Задание:

- Провести модернизацию ГТС в последовательности, указанной для каждого варианта. Рекомендуется изменить расположение МС в соответствии с их группировкой по этапам.
- Результаты модернизации представить в схемах, состоящих из двух плоскостей: плоскость сигнализации и плоскость передачи информации.

1.4 Указания к проведению работы.

Функционально обобщённая структура NGN представляется четырьмя плоскостями (уровнями):

- плоскость услуг (приложений): её задача – обеспечение всего спектра услуг доступного на сетях следующего поколения. В большинстве случаев для реализации уровня приложений выделяются отдельные серверы и базы данных;

- плоскость управления (сеть сигнализации), обеспечивающая реализацию пользовательских запросов путём коммутации пакетов в транспортной сети. Всё то многообразие устройств, которые транслируют и коммутируют трафик данных, преобразуют информацию, заложенную в пакеты, в стандартную телефонную сигнализацию и соединения, сопрягают цифровые сети различной природы, терминируют на себе различные виды трафика, управляется из одного мощного ядра. Это ядро связывается с понятием Softswitch. Основная функция Softswitch – управление соединением

абонента А с абонентом Б с обеспечением параметров качества обслуживания;

- базовая плоскость – транспортная сеть пакетной передачи. От технологий, используемых на этом уровне, во многом зависит уровень работы всей сети следующего поколения и количество предоставляемых сервисов. В качестве транспорта могут быть использованы сети ATM, Ethernet и как наиболее перспективная - технология MPLS;

- плоскость абонентского доступа, реализующая все необходимые виды интерфейсов в каждой конкретной сети. Под термином «доступ» предполагается очень широкое понятие от цифровых абонентских линий до пограничных шлюзов и конвертеров сигнализации. Доступ в общем случае – это всё то оборудование, которое связывает сеть NGN с традиционными TDM-сетями и даже небольшими локальными сетями передачи данных. Природа подключения может быть разной: DSL-системы с медной кабельной парой, системы цифрового кабельного телевидения, беспроводные системы Wi-Fi и WiMAX, оптические технологии доступа (например PON). Объединяет их всех одно – в качестве конечного интерфейса абоненту предоставляется IP-подключение, что даёт возможность использовать интеллектуальный терминал с доступом к большому числу дополнительных сервисов.

Модернизация ГТС, изложенная в данной работе, в основном, описывает изменения, проводимые в базовой плоскости и плоскости управления.

Модель NGN, отражающая принципы построения сети, может быть представлена структурой, приведенной на рисунке 1.1. Одна из существенных особенностей NGN - разделение функций передачи IP-пакетов и управления этим процессом. Передача информации, в которой заинтересованы пользователи, осуществляется коммутаторами пакетов (КП). Вторая функция возложена на устройства управления (УУ), в качестве которых используются различные аппаратно-программные средства.

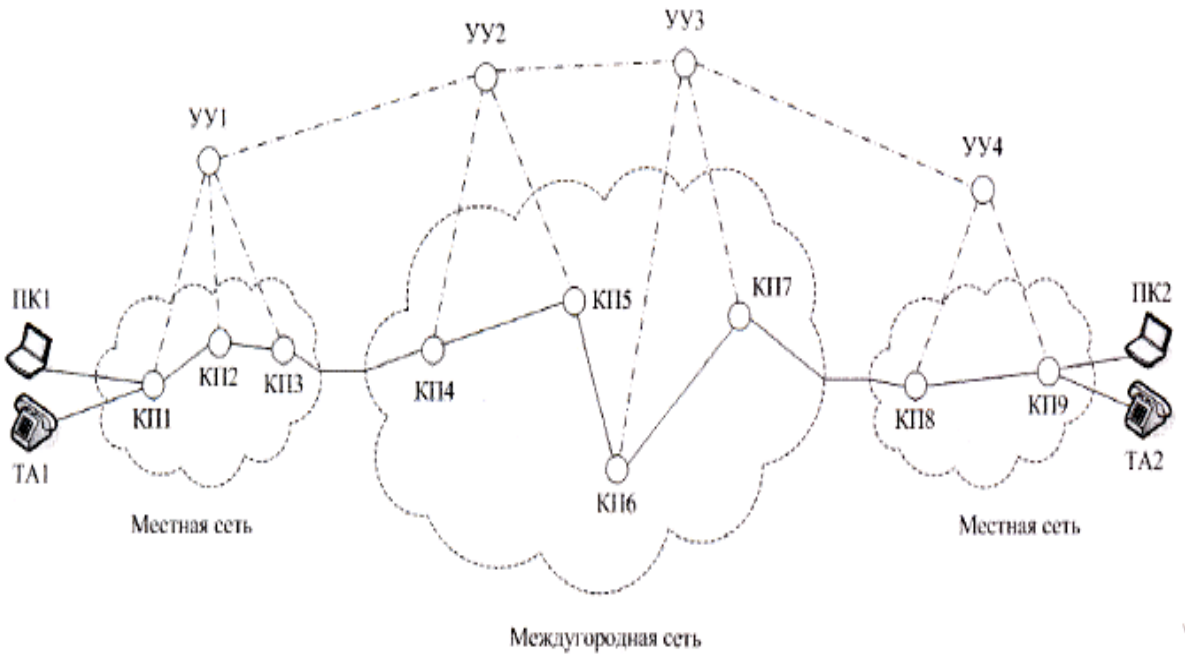

Рисунок 1.1. Модель сети следующего поколения.

Модель, представленная на рисунке 1.1, включает в себя три компонента: междугородную сеть и две местные сети. Количество КП в каждом компоненте сети было выбрано произвольно. Это справедливо также в отношении числа УУ, которые необходимы для определения основных атрибутов соединения. Предполагается, что оба пользователя располагают терминалами двух типов. Телефонный аппарат (ТА), необходимый для передачи речи. Персональный компьютер (ПК), обеспечивающий обмен данными и получение видеоинформации.

Три стратегии формирования NGN

У компании, решившей создать сеть следующего поколения, есть разные способы реализации поставленной задачи. Можно выделить три основных направления дальнейших действий (рисунок 1.2):

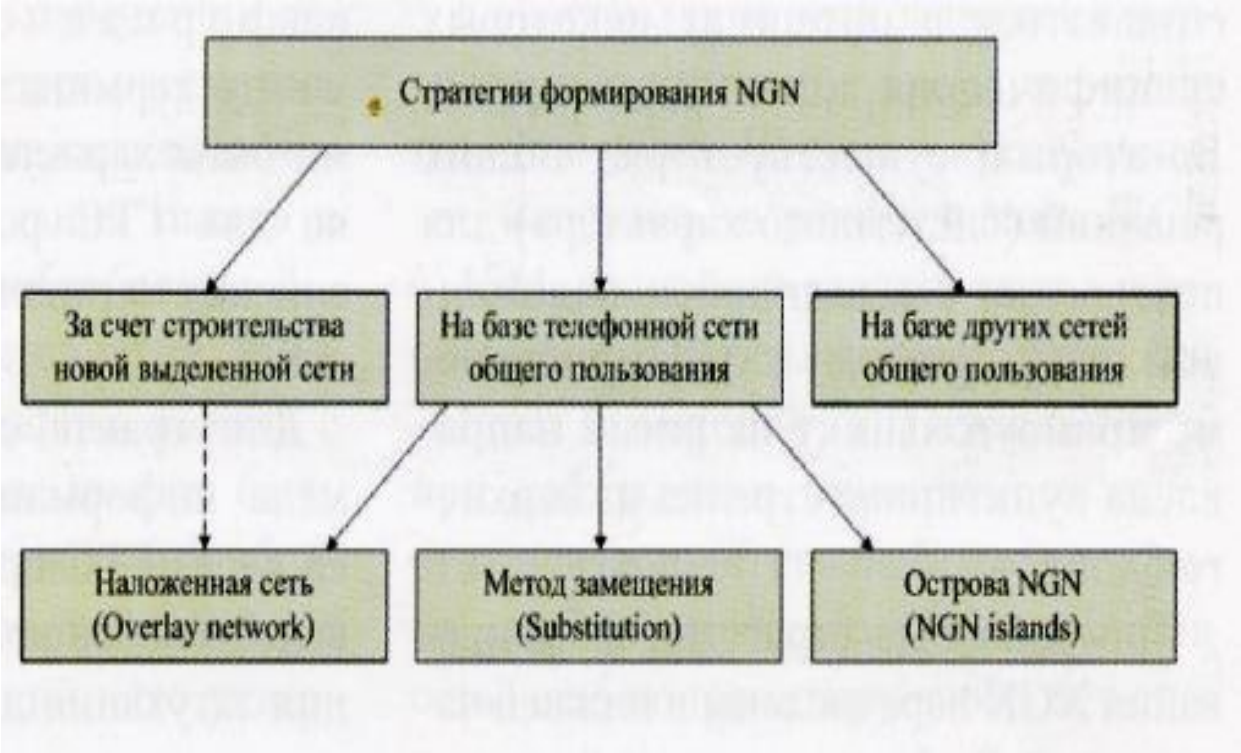

Рисунок 1.2. Классификация стратегий формирования NGN.

начать строительство новой выделенной сети, руководствуясь принципами, апробированными в начале девяностых годов прошлого века Операторами, которых стали называть альтернативными;

модернизировать телефонную сеть общего пользования (ТФОП) в полном соответствии со стандартами NGN;

создать NGN в результате реконструкции другой сети общего пользования (например, кабельного телевидения).

ТФОП по ряду причин представляется наиболее вероятной базой для построения NGN. Тем не менее организацию выделенной сети также следует рассматривать как одно из практически значимых решений. Тому есть две причины. Во-первых, выделенные сети NGN неизбежно будут создаваться в интересах некоторых специфических клиентских групп. Во-вторых, существует ряд общих решений (системного характера) для построения выделенной и наложенной сетей. Поэтому к левому нижнему прямоугольнику на рис. 2 направлена пунктирная стрелка из верхнего блока.

Три основные стратегии формирования NGN перечислены в нижней части рисунка 1.2. Выбор оптимальной стратегии может быть сделан путем анализа всех трех альтернатив.

Реализация стратегии "Наложенная сеть"

Сценарии реализации стратегии "Наложенная сеть" можно рассмотреть на примере ГТС, построенной по принципу связи коммутационных станций "каждая с каждой". Сети с такой структурой (рисунок 1.3) созданы во многих российских городах. Это районированные ГТС без узлообразования. Предполагается, что на местной станции (МС) под номером четыре в этой сети расположена транзитная станция (ТС), через которую осуществляется доступ к сети междугородной и международной телефонной связи. В данной работе аббревиатура МС распространяется только на районные АТС рассматриваемой ГТС, т.е. на РАТС.

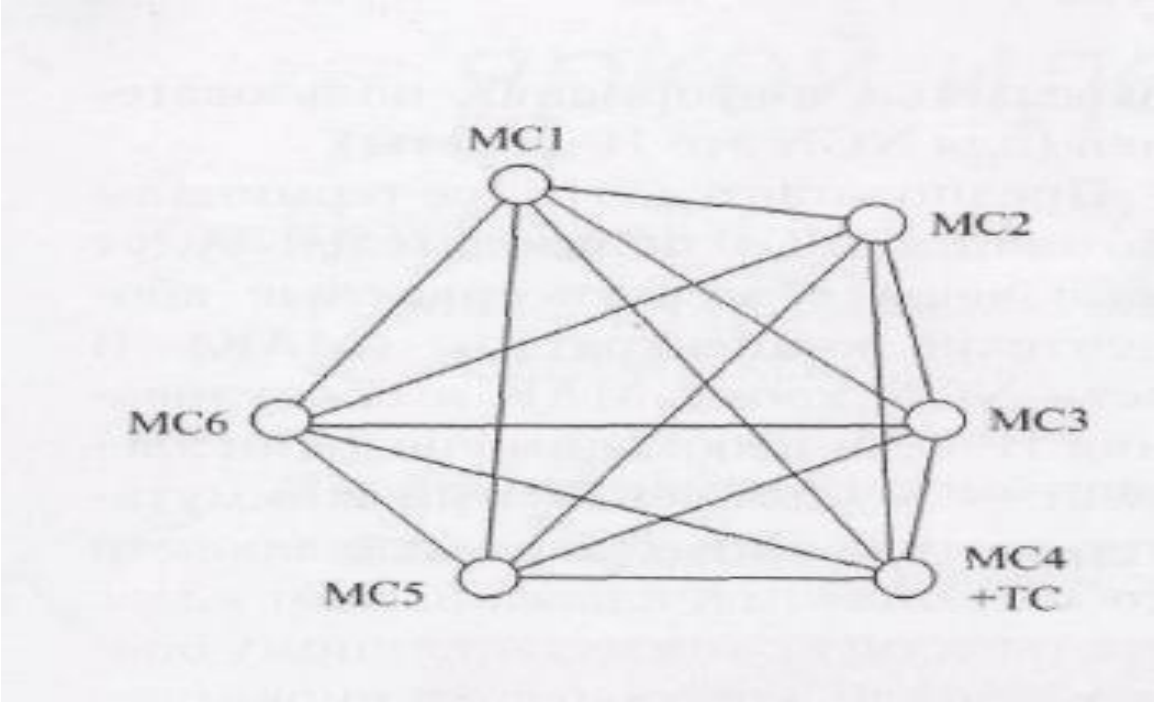

Рисунок 1.3. Структура ГТС без узлов.

Допустим, что заранее была определена оптимальная структура NGN (рисунок 1.4). Предположим также, что NGN начинает формироваться с уровня сетей международной и междугородной связи. Поэтому к моменту

модернизации нашей ГТС вместо автоматической междугородной телефонной станции (АМТС) уже установлен магистральный коммутатор (МК). Он обеспечивает транзит IP-пакетов, содержащих информацию любого вида (речь, данные, видео и их комбинация), в сетях междугородной и международной связи.

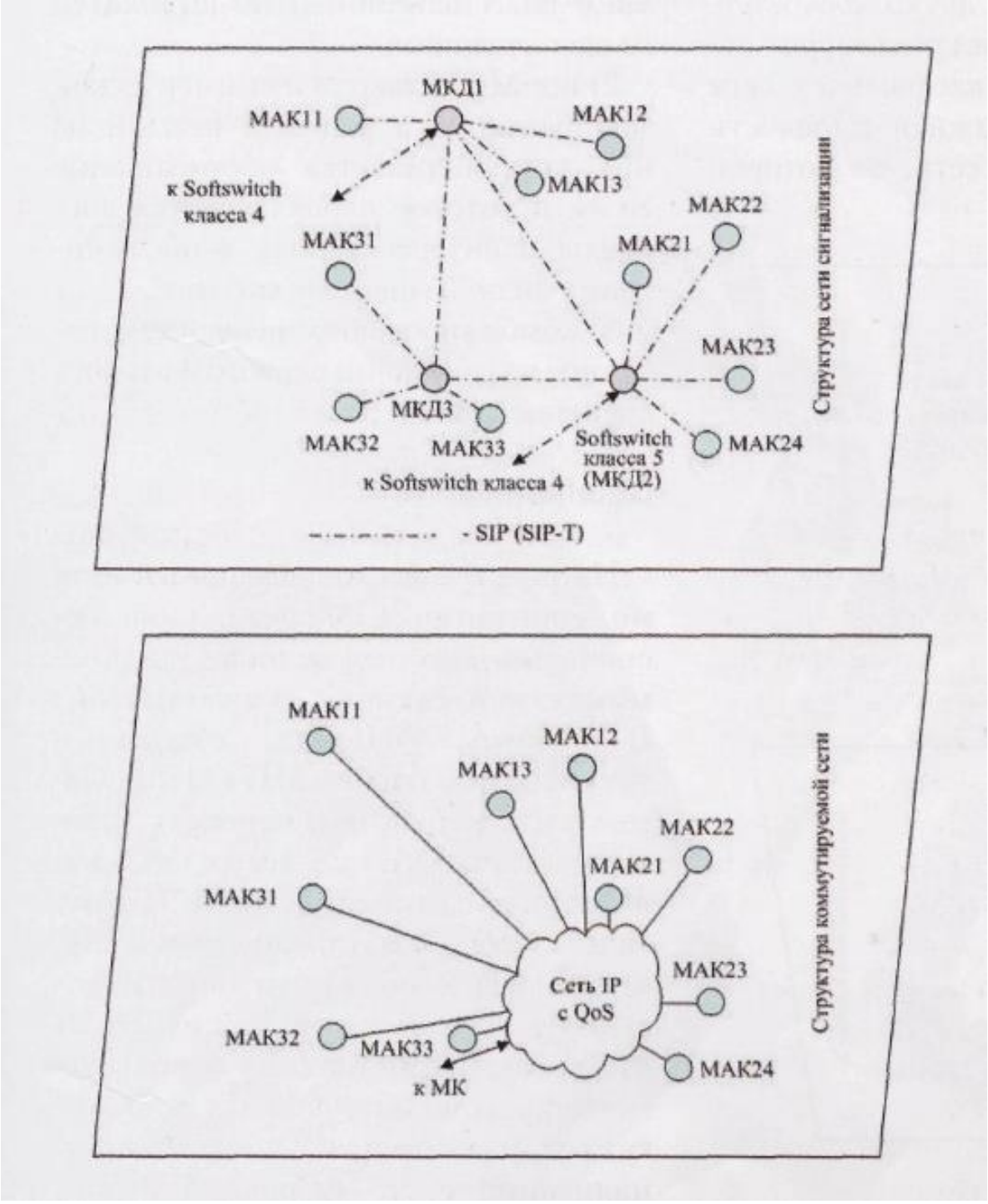

Рисунок 1.4. Оптимальная структура NGN.

Этот рисунок (как и несколько следующих) состоит из двух

плоскостей. Верхняя плоскость иллюстрирует основные изменения, касающиеся сети сигнализации. В нижней плоскости показана структура сети, по которой передается информация пользователей (для NGN это IP-пакеты).

Предполагается, что все терминалы потенциальных пользователей будут

включены в мультисервисные абонентские концентраторы (МАК). В сети NGN кроме МАК и оборудования IP-сети необходим еще один элемент - мультисервисный коммутатор доступа (МКД), представляющий собой Softswitch класса 5. Этот класс соответствует коммутационному оборудованию, которое функционирует на уровне местных станций. Для сигнализации на участках МАК - МКД, между МКД, а также между МКД и Softswitch класса 4 (который устанавливается на МК) предполагается использование протоколов SIP или SIP-T, но возможны и другие решения, если они соответствуют международным стандартам.

Создание NGN, структура которой показана на рисунке 1.4, может быть выполнено различными способами. С практической точки зрения следует выделить три сценария формирования NGN:

1) каждая МС после решения о необходимости ее замены оборудованием NGN одномоментно выводится из эксплуатации;
2) все МС остаются в коммерческой эксплуатации, а рядом с каждой из них устанавливается оборудование NGN, в которое переключаются абоненты, заинтересованные в обслуживании вида "Triple-play services";
3) комбинированное решение, основанное на сочетании первого и второго сценариев.

Сценарий 1

Первый этап

На рисунке 1.5 показан начальный этап модернизации ГТС без узлов

для сценария, который основан на одномоментной замене каждой МС. В данном примере замене подлежат РАТС МС3 и МС4. В границах IP-сети изображен транспортный шлюз MG (Media Gateway), который обеспечивает взаимодействие МАК со всеми МС, использующими технологию "коммутация каналов". Для анализа функций МКД необходимо обратиться к верхней плоскости рис. 5. Шесть МС, независимо от типа используемого оборудования коммутации, могут рассматриваться как пункты сигнализации - SP (Signaling Point). Такая трактовка предложена МСЭ при разработке спецификаций для обмена информацией по общему каналу сигнализации (ОКС). Номера SP и МС совпадают. Для нумерации пункта сигнализации, расположенного на ТС, выделена цифра "О".

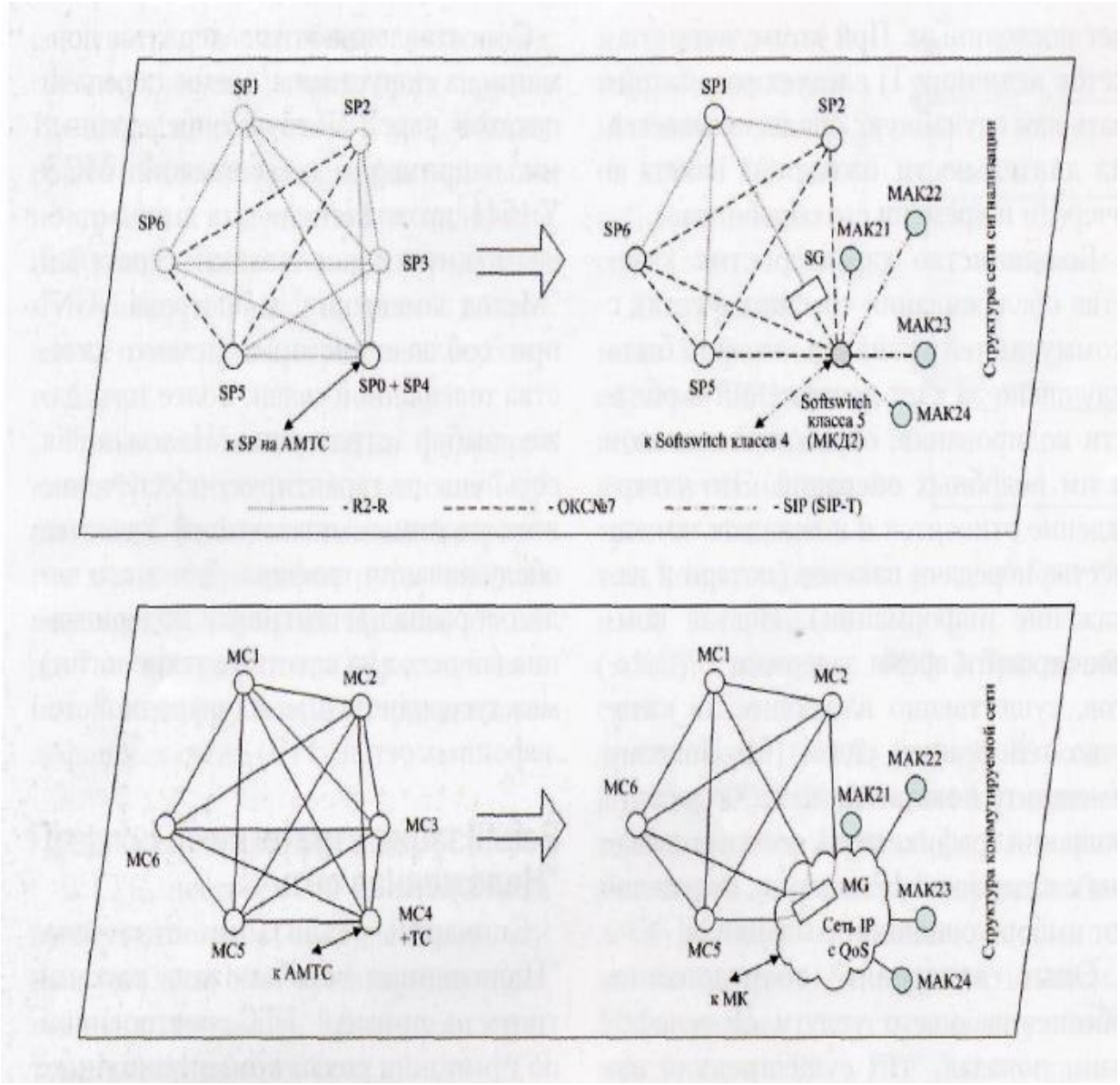

Рисунок 1.5. Первый этап модернизации ГТС без узлов. Сценарий I.

В городе начинает формироваться сеть IP, поддерживающая показатели качества обслуживания (Quality of Service, QoS), которые определены для NGN. Перечень таких показателей устанавливает Администрация связи. Основанием для нормирования этих показателей может служить, например, рекомендация МСЭ Y.1541. На начальном этапе создания NGN в сети IP может использоваться всего один коммутатор. В рассматриваемом примере четыре концентратора МАК обеспечивают обслуживание всех абонентов, ранее включенных в МС3 и МС4.

Следует подчеркнуть, что для взаимодействия с аналоговыми МС необходим шлюз сигнализации SG (Signalling Gateway). Дело в том, что коммутаторы Softswitch не поддерживают процессы обмена сигналами управления и взаимодействия, которые используются в отечественных аналоговых коммутационных станциях. Предполагается, что только МС1 построена на аналоговом коммутационном оборудовании. Система сигнализации, принятая для российских аналоговых МС, названа здесь R2-R (или R1,5). Такое обозначение расшифровывается как российская версия системы сигнализации R2, принятой МСЭ.

Второй этап

На рисунке 1.6 показан один из возможных вариантов дальнейшего построения NGN. Он рассматривается как второй этап модернизации ГТС и основан на замене двух коммутационных станций: МС1 и МС2. Одновременная замена двух МС - один из возможных вариантов развития городской инфокоммуникационной системы в соответствии с выбранным сценарием. Решения такого рода интересны с точки зрения минимизации затрат на сеть доступа. Отправной точкой для выбора рационального решения служит вариант, предусмотренный программой модернизации ГТС.

Установка МКД1 подразумевает реконструкцию сети доступа, в которой появляются еще три МАК. Между абонентами семи эксплуатируемых

МАК информация всех типов передается в виде IP-пакетов. Управляют всеми соединениями два МКД. Переход к технологии "коммутация каналов" необходим только для соединений, которые устанавливаются с терминалами, включенными в МС5 или в МС6.

Радикальные изменения свойственны сети сигнализации. Только для МС5 и МС6 используются системы сигнализации, реализованные для телефонной связи. Все остальные элементы городской сети (МАК и МКД) уже взаимодействуют между собой по единой системе сигнализации, принятой для NGN.

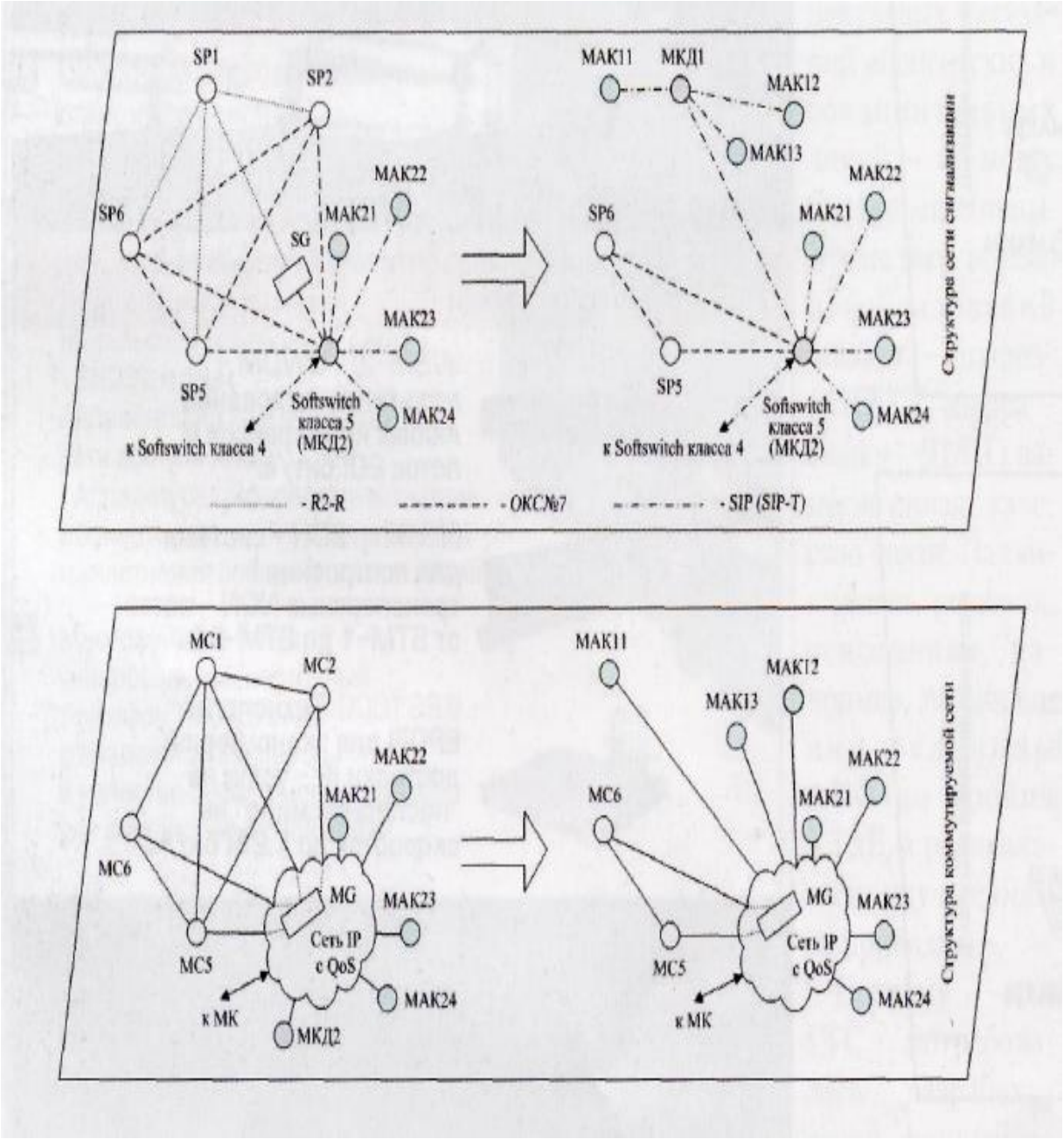

Рисунок 1.6. Второй этап модернизации ГТС без узлов. Сценарий I.

Третий этап

Структура сети становится все более похожей на структуру NGN, формирование которой завершается на третьем - заключительном - этапе. Этот этап (рисунок 1.7) приводит к созданию сети со структурой, которая была выбрана в качестве оптимального решения (см. рисунок 1.4). Варианты модернизации ГТС в соответствии с рассматриваемым сценарием могут различаться темпами замены эксплуатируемого оборудования коммутации,

численностью МКД и МАК в IP-сети, а также другими атрибутами, не влияющими на принципы поэтапного создания NGN.

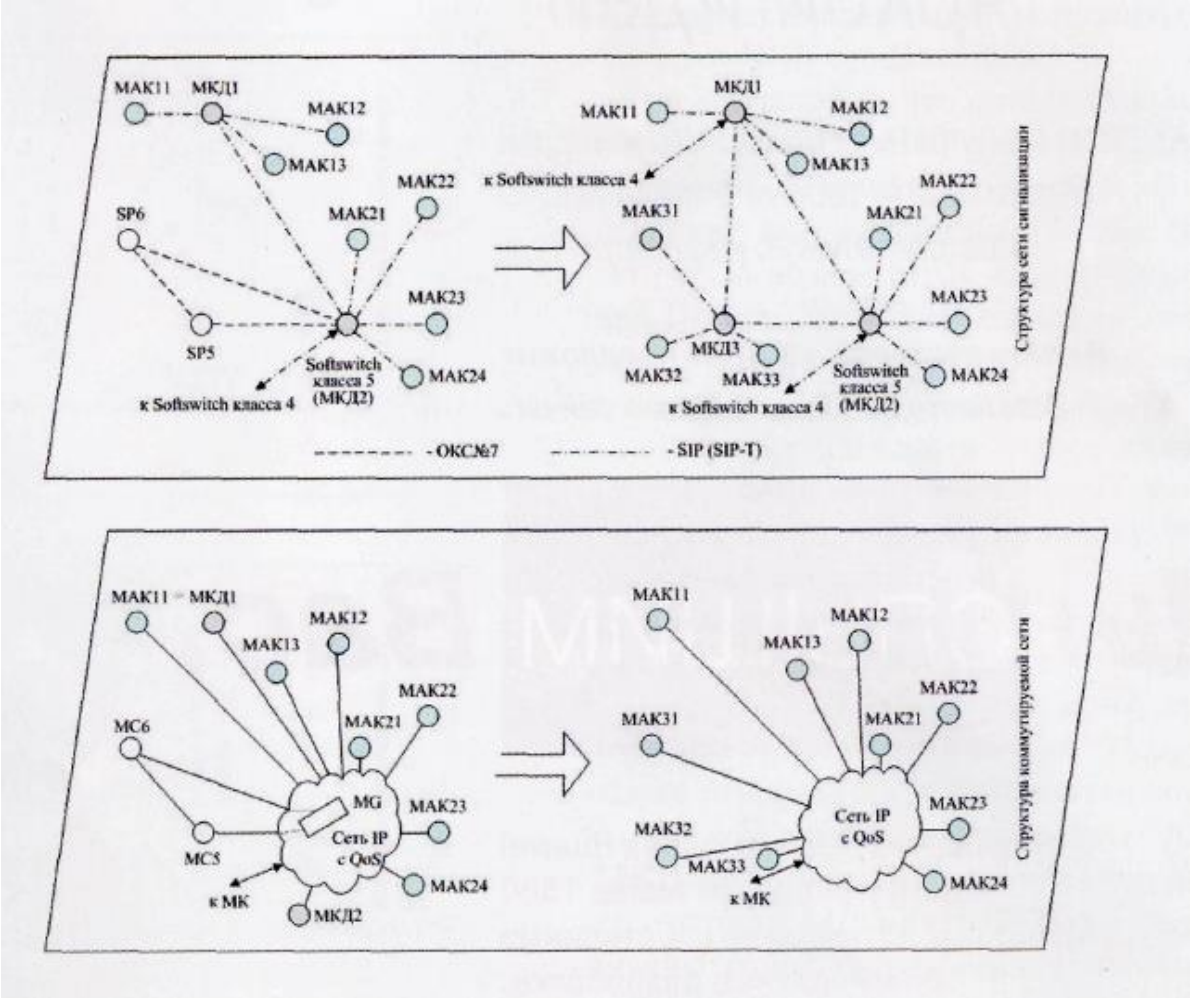

Рисунок 1.7. Третий этап модернизации ГТС без узлов. Сценарий I.

Следует упомянуть еще одну проблему - выбор технологий, необходимых для поддержки показателей QoS. Эта задача требует дополнительного исследования. Следует, правда, отметить, что затраты Оператора на создание сети IP с поддержкой QoS существенно меньше тех инвестиций, которые потребуются для замены всех МС и реализации современной сети доступа.

Сценарий 2

Второй сценарий может быть представлен с помощью модели, показанной на рисунке 1.8. Он не содержит верхнюю плоскость, так как принципы

построения системы сигнализации не изменяются. Результат реализации этого сценария можно рассматривать как оперативное создание "наложенной сети", пользователям которой доступны все виды обслуживания, входящие в набор "Triple-play services". Численность таких пользователей ограничена: она определяется уровнем платежеспособного спроса на обслуживание вида "Triple-play services".

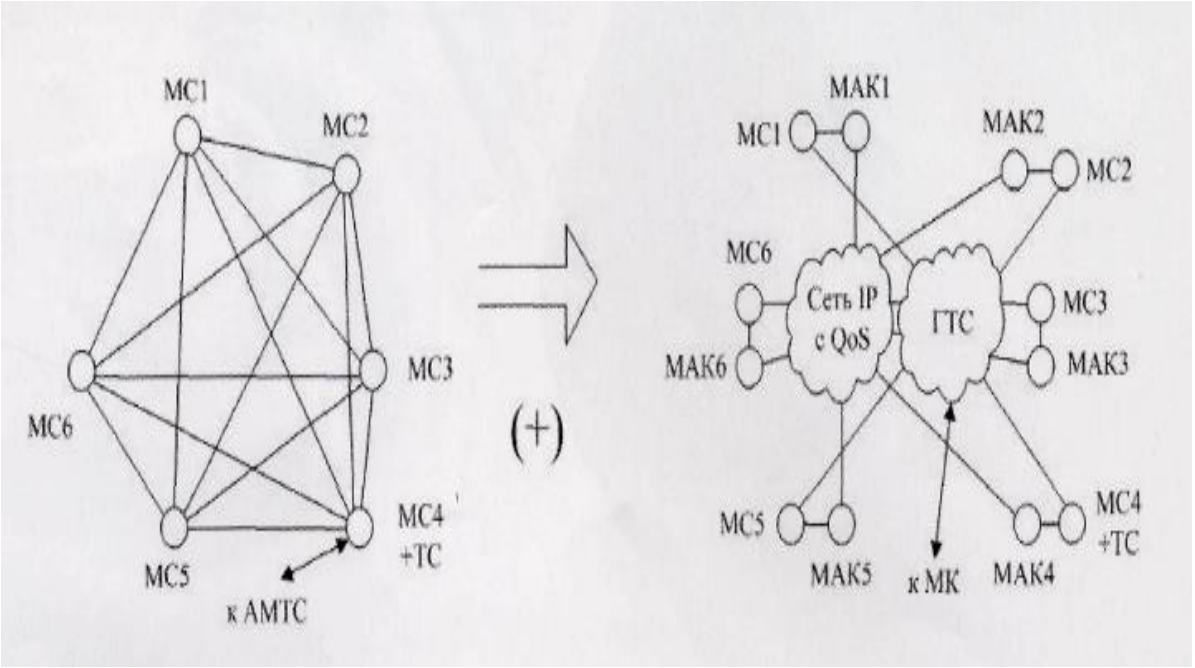

Рисунок 1.8. Второй сценарий модернизации ГТС без узлов.

Существенное отличие от первого сценария создания NGN состоит в том, что параллельно формируется "вторая" сеть. Это обстоятельство подчеркивает знак "+" на рисунке 1.8, расположенный под стрелкой, которая разделяет этапы развития ГТС. Очевидно, что сеть сигнализации должна быть создана сразу и почти в полном объеме. Между каждой МС и тем МАК, который установлен рядом с ней, должны быть организованы тракты Е1. Фактически каждый МАК становится выносным модулем одной из МС, которая обеспечивает ему выход в ГТС.

Сценарий 3

Третий сценарий реализации стратегии "Наложенная сеть"

представляет собой сочетание двух решений, которые были рассмотрены выше. Любой из трех сценариев обеспечивает формирование NGN, которая должна соответствовать всем показателям качества обслуживания мультисервисного трафика.

Выбор результирующей структуры сети – предмет отдельного решения, не влияющий на изложенную методику. Предполагается, что все МКД, именуемые в NGN как softswitch класса 5, должны быть связаны между собой для обеспечения высокой надёжности системы сигнализации в сети NGN. Кроме того, предусматривается организация двух независимых направлений для обмена информацией с оборудованием междугородной станции (АМТС), именуемой в NGN как softswitch класса 4, которая скорее всего будет располагаться в центре субъекта федерации. Выход к этому softswitch должен осуществляться через разные МКД. Также в интересах надёжности рекомендуется каждый МАК подключать к сети IP по 2-м независимым путям.

Выше описана процедура модернизации шестиузловой ГТС, проведённой в три этапа: МС3 и МС4 (1-й этап), МС1 и МС2 (2-й этап) и МС5 и МС6 (3-й этап).

При выполнении практического задания необходимо провести модернизацию той же самой ГТС, но в последовательности, указанной для каждого варианта в таблице 1.1. Рекомендуется изменить расположение МС в соответствии с их группировкой по этапам. Схему связей такая перестановка не меняет, так как все МС соединены между собой по схеме «каждый с каждым».

Результаты модернизации представить в схемах, состоящих из двух плоскостей: плоскость сигнализации и плоскость передачи информации, по аналогии с приведённым выше примером. Мультисервисные абонентские концентраторы (МАК) расположить по сети произвольно с общим их числом не менее 12. Число МКД должно быть не менее 3.

Таблица 1.1. Варианты модернизации

Номер варианта	Номера заменяемых станций		
	1-й этап	2-й этап	3-й этап
1	1	2,3	4,5,6
2	1,2	3,4,5	6
3	2,3	1	4,5,6
4	4,5,6	2,3	1
5	1	4,5,6	2,3
6	6	1,2	3,4,5
7	3,4	6	1,2,5
8	2,5	6	1,3,4
9	2	1,3	4,5,6
10	4,5	2	1,3,6
11	1,3,5	2	4,6
12	3,6	1,4,5	2
13	3	2,4	1,5,6
14	2,5	3	1,4,6
15	1,6	2,4,5	3
16	4,5	1,2,6	3
17	4	2,5	1,3,6
18	1,3,5	2,6	4
19	3,5,6	4	1,2
20	5,6	1,2,3	4
21	5	1,3	2,4,6
22	4,6	5	1,2,3
23	2,3,4	1,6	5
24	1,4	2,3,6	5

1.5 Отчет по работе:

- Результаты расчета мультисервисных потоков в ветвях сети.

- Схемы модернизированной сети на каждом из этапов.
- Схемы сети сигнализации.

Практическое занятие №2

Конфигурирование статической маршрутизации

2.1 Цель работы: Привитие навыков конфигурирования статической маршрутизации в сетях связи следующего поколения

2.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Linux.

2.3 Задание:

- Произвести физические соединения для построения заданной топологии;
- Сконфигурировать статическую маршрутизацию га маршрутизаторе Cisco;
- Сконфигурировать статическую маршрутизацию на программных маршрутизаторах;
- Произвести проверку связности сети с использованием утилит ping и traceroute.

2.4 Указания к проведению работы.

Для маршрутизации в ОС Linux используется утилита **route**. Данная утилита может использоваться как для просмотра, так и для модификации таблиц маршрутизации. Благодаря протоколам удаленного доступа Telnet и SSH утилита **route** может быть использована для дистанционного управления процедурами маршрутизации на серверах и программных маршрутизаторах, на которых установлена ОС Linux.

Команда **route** имеет следующий формат

route [операция] [тип] адресат gw [шлюз] [метрика] [интерфейс]

При использовании команды **route** без аргументов и параметров отображается содержимое таблицы маршрутизации, рисунок 2.1.

```
sssk@sssk-VirtualBox:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         10.0.2.2        0.0.0.0         UG    1024   0        0 eth0
10.0.2.0        *               255.255.255.0   U     0      0        0 eth0
link-local      *               255.255.0.0     U     1000   0        0 eth0
```

Рисунок 2.1 – Таблица маршрутизации

Параметр **операция** используется для добавления или удаления статического маршрута, соответственно, может принимать значение **add** (добавить) или **del** (удалить).

Параметр **тип** определяет, является ли описываемый маршрут маршрутом к сети (**-net**) или к отдельной машине (**-host**). Данный параметр целесообразно использовать при бесклассовой маршрутизации, чтобы ядро ОС могло надежно отделить адрес сети от адреса узла. Например, адрес 128.138.243.0 с маской 255.255.255.0 является адресом сети, а не узла, несмотря на то, что он относится к классу B. Поэтому если не указать параметр **-net**, утилита **route** воспримет этот адрес как адрес конечного узла (сеть 128.134, узел 243.0).

Параметр **адресат** содержит адрес сети или узла или запись **default**, если конфигурируется маршрут по умолчанию.

Параметр **шлюз** определяет адрес следующего устройства (маршрутизатора), которому должны передаваться пакеты, соответствующие данному маршруту.

Необязательный параметр **метрика** задает предельное число транзитных переходов на пути к адресату.

Для удаления маршрута чаще всего используют команду

route del адрес

Для примера создадим статический маршрут к сети 219.15.0.0, путь к которому должен проходить через шлюз с адресом 10.0.2.1, после чего выведем на экран таблицу с вновь созданным маршрутом, рисунок 2.2.

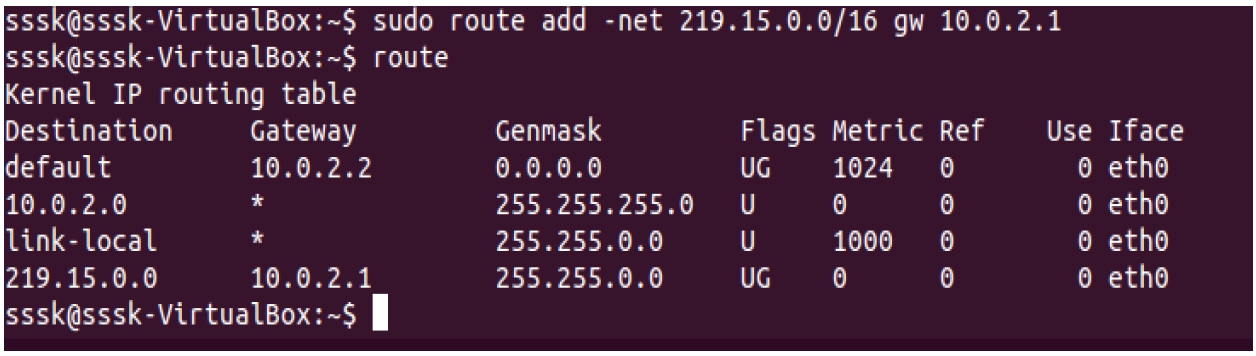

Рисунок 2.2 – Добавление нового маршрута

Сетевые устройства (коммутаторы и маршрутизаторы) Cisco могут находиться в одном из режимов, представленных в таблице 2.1.

Таблица 2.1 – Режимы конфигурирования

Название режима	Приглашение (Prompt)	Описание
User EXEC Mode	Switch>	Пользовательский режим
Privileged EXEC Mode	Switch #	Привилегированный режим
Global Configuration Mode	Switch (config)#	Режим глобального конфигурирования

В таблице 2.1 в первом столбце представлены названия режимов, во втором – приглашение, отображаемое в командной строке, в третьем – описание.

Пользовательский режим (user mode) используется для просмотра состояния устройства, а также для перехода в привилегированный режим (privileged mode). Никаких изменений в конфигурационном файле, в том числе удаление и сохранение текущей конфигурации, в пользовательском режиме производиться не может. В этом режиме доступны только некоторые команды верификации **show**, т. е. команды просмотра состояния устройства.

Для перехода в привилегированный режим необходимо выполнить команду

enable

Следует отметить, что оборудование Cisco допускает сокращенный ввод команд, если невозможно ее двоякое толкование. Например, вместо **enable** можно набрать **en**, и эта команда будет понятна коммутатору, так как никакая другая команда не начинается с сочетания символов en.

Выполнение каждой команды начинается после нажатия клавиши **Enter**. Для повтора ранее введенных команд можно использовать клавишу ↑.

В привилегированном режиме доступны все команды **show**, возможно удаление конфигурации и сохранение конфигурационного файла в памяти NVRAM. Возврат в пользовательский режим производится командой **disable** или **exit**:

Router#exit

Команда **exit** позволяет вернуться на один уровень вверх. Например, после выполнения команды в режиме глобального конфигурирования коммутатор переходит в привилегированный режим. Если необходимо из любого состояния устройства выйти сразу в пользовательский режим, используется комбинация клавиш CTRL-Z.

В глобальном режиме производятся изменения, которые затрагивают коммутатор в целом, поэтому этот режим и называется global configuration mode. Например, в нем можно устанавливать имя коммутатора командой **hostname**. Имя коммутатора не имеет значения в сети, оно удобно при конфигурировании. Пример:

Router(config)#hostname Router_A

Router_A(config)#

Изменение и создание конфигурации маршрутизатора Cisco возможно в режиме глобального конфигурирования, вход в который реализуется из привилегированного по команде **configure terminal** (сокращенно – **conf t**), которая вводит устройство в глобальный режим и позволяет изменять

текущую конфигурацию (running-config). При этом приглашение изменяет вид на **Router(config)#:**

Router>en

Router#conf t

В отличие от коммутаторов, маршрутизаторы для своего правильного функционирования должны быть правильно сконфигурированы. Как минимум, необходимо интерфейсам маршрутизатора присвоить адреса. Если адреса назначаются администратором и конфигурируются администратором вручную, такие адреса называются статическими. Статические адреса назначаются в режиме конфигурирования интерфейса. Для перехода в режим конфигурирования интерфейса из режима глобального конфигурирования используется команда **interface <номер>**. Для назначения адреса используется команда **ip address <адрес> <маска>**. Например, для назначения интерфейсу FastEthernet 0/0 IP-адреса 192.168.1.1 с маской 24, а интерфейсу FastEthernet 0/1 IP-адреса 10.10.10.1 с маской 8 необходимо выполнить следующие команды:

Router(config)#interface fa 0/0

Router(config-if)#ip address 192.168.1.1 255.255.255.0

Router(config-if)#no shutdown

Router(config-if)#exit

Router(config)#interface fa 0/1

Router(config-if)#ip address 10.10.10.1 255.0.0.0

Router(config-if)#no shutdown

По умолчанию интерфейсы маршрутизаторов Cisco находятся в отключенном состоянии, поэтому перед использованием их необходимо включить командой **no shutdown**.

Для просмотра таблицы маршрутизации используется команда **show ip route**. Пример использования данной команды иллюстрируется рисунком 2.3.

```
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 210.16.52.101 to network 0.0.0.0

     192.168.1.0/27 is subnetted, 3 subnets
C       192.168.1.32 is directly connected, FastEthernet1/0
C       192.168.1.64 is directly connected, FastEthernet6/0
C       192.168.1.96 is directly connected, FastEthernet7/0
C    210.16.52.0/24 is directly connected, FastEthernet0/0
S*   0.0.0.0/0 [1/0] via 210.16.52.101
Router#
```

Рисунок 2.3 – Просмотр таблицы маршрутизации

Из рисунка 3.1 следует, что к маршрутизатору непосредственно подключены подсети 192.168.1.32/27, 192.168.1.64/27, 192.168.1.96/27 и сеть 210.16.52.0/24 (отмечены символом C – Connected). Кроме этого, имеется один статический маршрут по умолчанию (отмечен символом S – Static).

Для конфигурирования статических маршрутов используется команда **ip route <сеть назначения> <маска> <next hop>**.

Для выполнения практических заданий по конфигурированию инфокоммуникационной сети необходимо использовать стенд «Инфокоммуникационные сети» (ауд. 221). В состав стенда входят три маршрутизатора (два программных и один аппаратный), составляющие ядро сети, три коммутатора и произвольное количество рабочих станций (PC), составляющие сеть доступа. Соединение сетевых устройств между собой производится с использованием кабеля «витая пара» с разъемами RJ-45. Схема, которую необходимо собрать, с указанием адресов сетевых интерфейсов представлена на рисунке 2.4.

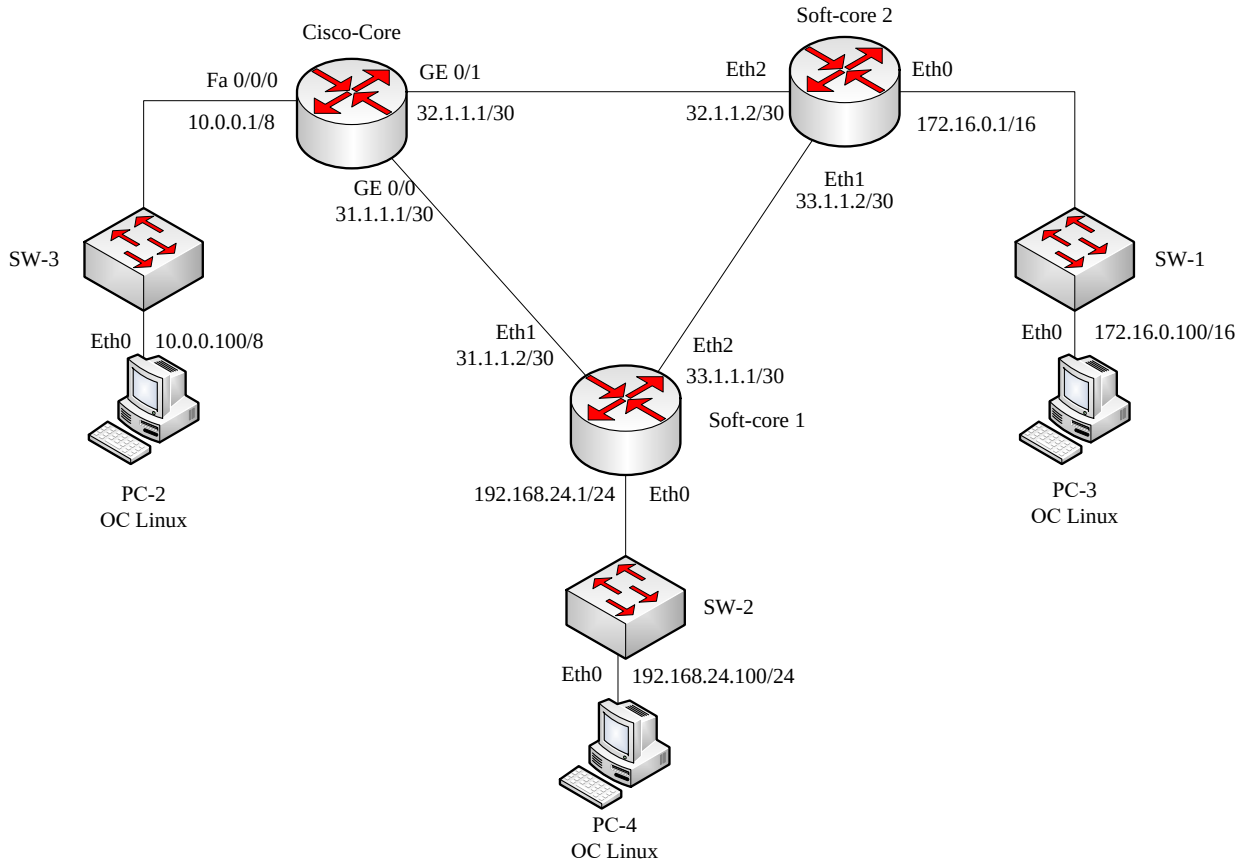

Рисунок 2.4 – Схема инфокоммуникационной сети

Программные маршрутизаторы обозначены на схеме Soft-core и представляют собой специализированные серверы под управлением ОС Linux с установленным пакетом маршрутизации Quagga. Система команд Quagga в теоретической части пособия не рассматривалась, так как она очень близка к рассмотренной выше системе команд Cisco iOS. Однако в дальнейшей работе необходимо иметь в виду, что Quagga не поддерживает сокращенный набор команд.

Аппаратный маршрутизатор представляет собой устройство Cisco 1800, конфигурирование которого не отличается какими то ни было особенностями.

Уровень доступа представлен одним неуправляемым коммутатором уровня 2, одним управляемым коммутатором уровня 2 и одним коммутатором уровня 3. В коммутаторы включается оконечное оборудование пользователя, в качестве которых выступают PC под управлением ОС Linux (Ubuntu).

После сборки схемы сети необходимо убедиться в том, что рабочим станциям (PC-2, 3, 4) присвоены минимальные настройки, к которым относятся IP-адреса, маски подсети, а также значения шлюзов по умолчанию. Если настройки рабочих станций отличаются от тех, которые обозначены на рисунке 1.4, необходимо произвести корректировку в соответствии с методикой, изложенной выше.

Далее необходимо произвести минимальную настройку маршрутизатора Cisco, в частности, присвоить IP-адреса его интерфейсам в соответствии с рисунком 4.1. Кроме того, необходимо прописать статические маршруты к подсетям 192.168.24.0/24 и 172.16.0.0/16 (или маршруты по умолчанию).

Программные маршрутизаторы конфигурируются удаленно посредством протокола SSH. Для этого необходимо зайти на программные маршрутизаторы с рабочих станций тех локальных сетей, шлюзами которых они являются, то есть на Soft-core 1 с PC-4, а на Soft-core 2 с PC-3. Логин и пароль для входа на программные маршрутизаторы admin, для входа в маршрутизатор под суперпользователем пароль – simulator. Для получения доступа к консоли управления Quagga необходим пароль softcore.

Вход на маршрутизатор Soft-core 1 иллюстрируется рисунком 2.5.

```
sudowireshark@a217-PC3:~$ ssh admin@192.168.24.1
admin@192.168.24.1's password:
Last login: Sat Dec 20 18:13:15 2014 from 192.168.24.2
[admin@sc-1 ~]$ su
Пароль:
[root@sc-1 admin]# iptables -F
[root@sc-1 admin]# telnet localhost 2601
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 0.99.15).
Copyright 1996-2005 Kunihiro Ishiguro, et al.


User Access Verification

Password:
soft-core1.lab>
```

Рисунок 2.5 – Вход на Soft-core 1

На рисунке 2.5 также показана команда **iptables –F**, которую необходимо выполнить для очистки правил межсетевого экрана NetFilters.

После входа на маршрутизатор необходимо назначить IP-адреса его интерфейсам, а также прописать статические маршруты к подсетям согласно рисунку 2.4. Конфигурирование маршрутизатора Soft-core 1 иллюстрируется рисунком 2.6.

```
soft-core1.lab> enable
soft-core1.lab# configure terminal
soft-core1.lab(config)# ip forwarding
soft-core1.lab(config)# interface eth1
soft-core1.lab(config-if)# ip address 31.1.1.2/30
soft-core1.lab(config-if)# no shutdown
soft-core1.lab(config-if)# exit
soft-core1.lab(config)# interface eth2
soft-core1.lab(config-if)# ip address 33.1.1.1/30
soft-core1.lab(config-if)# no shutdown
soft-core1.lab(config-if)# exit
soft-core1.lab(config)# ip route 10.0.0.0/8 31.1.1.1
soft-core1.lab(config)# ip route 172.16.0.0/16 33.1.1.2
soft-core1.lab(config)# exit
soft-core1.lab#
```

Рисунок 2.6 – Конфигурирование маршрутизатора Soft-core 1

Аналогичным образом конфигурируется маршрутизатор Soft-core 2.

Для проверки корректности внесенных маршрутов можно просмотреть таблицу маршрутизации, рисунок 2.7.

```
soft-core1.lab# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

S>* 0.0.0.0/0 [1/0] via 33.1.1.2, eth1
S>* 10.0.0.0/8 [1/0] via 31.1.1.1, eth1
C>* 31.1.1.0/30 is directly connected, eth1
C * 33.1.1.0/30 is directly connected, eth2
C>* 33.1.1.0/30 is directly connected, eth1
C>* 70.1.1.0/30 is directly connected, eth1
C>* 127.0.0.0/8 is directly connected, lo
K>* 169.254.0.0/16 is directly connected, eth2
S>* 172.16.0.0/16 [1/0] via 33.1.1.1, eth1
  *                     via 33.1.1.2, eth1
C>* 192.168.24.0/24 is directly connected, eth0
soft-core1.lab#
```

Рисунок 2.7 – Таблица маршрутизации Soft-core 1

Для проверки связности сети необходимо использовать утилиту ping на любой рабочей станции сети. Проверка на PC-4 иллюстрируется рисунком 2.8.

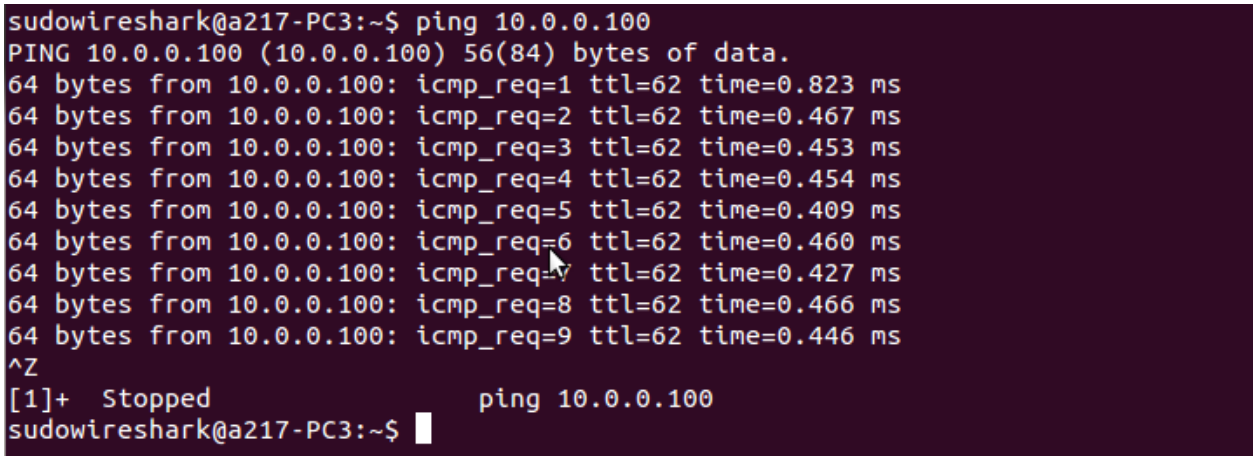

Рисунок 2.8 – Проверка связности сети

Если ICMP-пакеты успешно передаются из одной подсети в другую, начальное конфигурирование можно считать законченным. Необходимо также отследить маршруты прохождения пакетов с использованием утилиты traceroute.

2.5 Отчет по работе:

– Работоспособная сеть.

Практическое занятие № 3

Конфигурирование динамической маршрутизации (протокол RIP)

3.1 Цель работы: Привитие навыков конфигурирования динамической маршрутизации с использованием протокола RIP

3.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Linux.

3.3 Задание:

- Произвести физические соединения для построения заданной топологии;
- Сконфигурировать динамическую маршрутизацию на маршрутизаторе Cisco;
- Сконфигурировать динамическую маршрутизацию на программных маршрутизаторах;
- Произвести проверку связности сети с использованием утилит ping и traceroute.

3.4 Указания к проведению работы.

Протокол RIP (Routing Information Protocol) является одним из первых протоколов маршрутизации и относится к дистанционно-векторным протоколам. Существует две версии RIP – первая версия (RIPv1) использует маршрутизацию на основе классов и описана в RFC 1058, вторая версия (RIPv2) использует бесклассовую маршрутизацию и описана в RFC 1388.

Применение дистанционно-векторной маршрутизации накладывает ограничения на размер составной сети. При этом вводится понятие максимального диаметра сети – максимальное расстояние, на которое может быть передан пакет, после превышения которого пункт назначения считается недостижимым. Для протоколов RIP обеих версий максимальный диаметр сети составляет 15 маршрутизаторов, соответственно, маршрут с метрикой 16 считается недостижимым.

Для рассмотрения процедур, предусмотренных протоколом RIP, рассмотрим пример составной сети, представленный на рисунке 3.1.

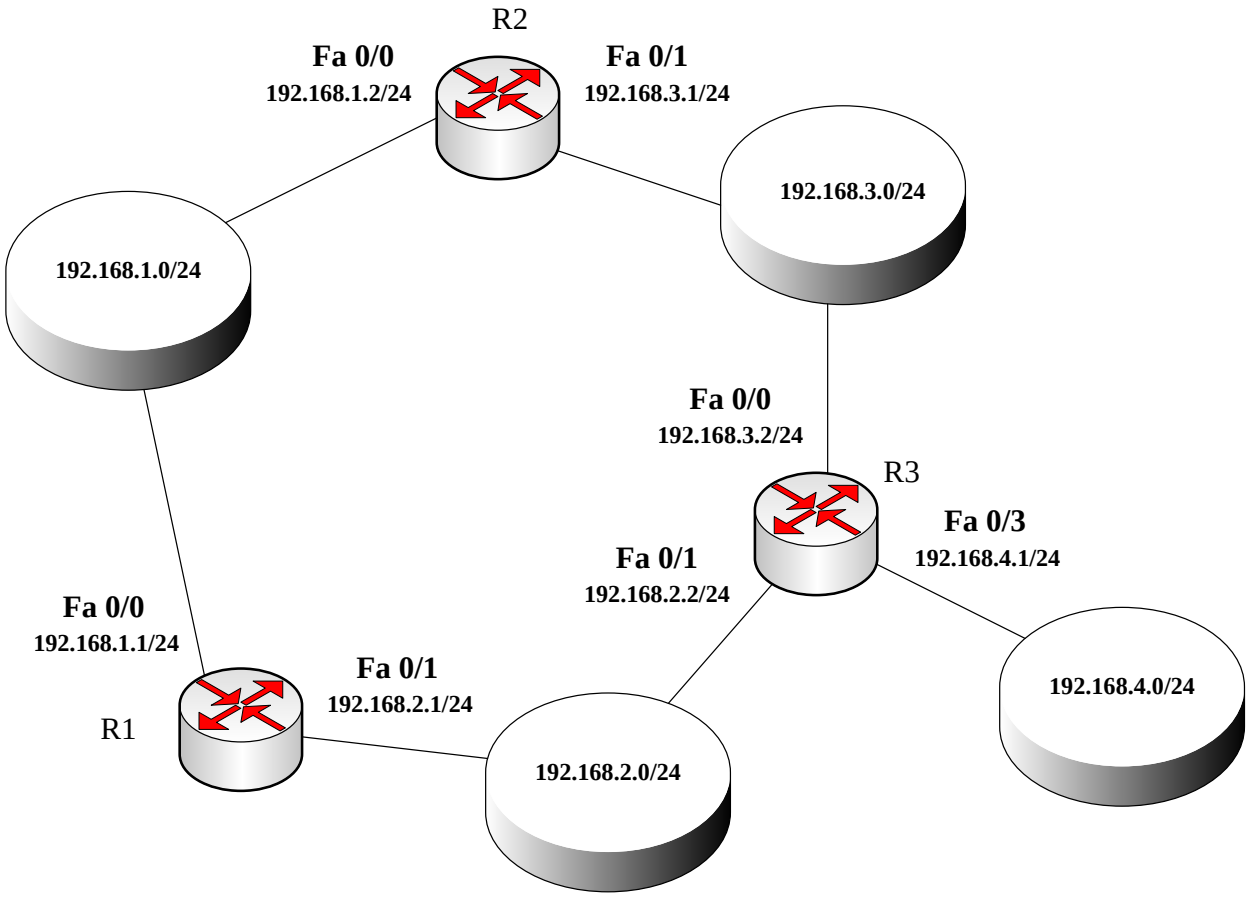

Рисунок 3.1 – Пример сети

На рисунке 3.1 представлены три маршрутизатора R1 – R3, у каждого из которых обозначены порты Fast Ethernet (Fa) с назначенными IP-адресами. Адреса портов, как и ранее, соответствуют адресам подсетей, в которые они входят.

На первом этапе протокола RIP создаются минимальные таблицы маршрутизации, которые содержат только адреса непосредственно подключенных подсетей.

Минимальную таблицу маршрутизатора R1 представим в виде таблицы 3.1.

Минимальные таблицы маршрутизаторов R2 и R3 имеют аналогичный вид и представлены в таблицах 3.2 и 3.3, соответственно.

Таблица 3.1 – Минимальная таблица маршрутизатора R1

Адрес сети назначения	Адрес порта следующего маршрутизатора	Адрес выходного порта	Метрика
192.168.1.0	-	192.168.1.1	1
192.168.2.0	-	192.168.2.1	1

Таблица 3.2 – Минимальная таблица маршрутизатора R2

Адрес сети назначения	Адрес порта следующего маршрутизатора	Адрес выходного порта	Метрика
192.168.1.0	-	192.168.1.2	1
192.168.3.0	-	192.168.3.1	1

Таблица 3.3 – Минимальная таблица маршрутизатора R3

Адрес сети назначения	Адрес порта следующего маршрутизатора	Адрес выходного порта	Метрика
192.168.3.0	-	192.168.3.2	1
192.168.2.0	-	192.168.2.2	1
192.168.4.0	-	192.168.4.1	1

На следующем этапе каждый из маршрутизаторов рассылает минимальную таблицу своим «соседям». Для этого используется UDP-дейтаграмма с номером порта 520. В этой дейтаграмме содержатся сведения о сети, имеющейся в минимальной таблице, и расстоянии до нее.

Например, «соседями» для маршрутизатора R1 являются маршрутизаторы R2 и R3. Поэтому им передаются сообщения примерно следующего вида:

- сеть 192.168.1.0, метрика 1;

- сеть 192.168.2.0, метрика 1.

Аналогичным образом свою минимальную таблицу маршрутизатор R2 передает маршрутизаторам R1 и R3, а маршрутизатор R3 – маршрутизаторам R1 и R2.

После получения информации от своих «соседей» маршрутизатор обрабатывает ее – увеличивает значение принятой метрики на единицу и запоминает порт, на который пришло данное сообщение, а также адрес маршрутизатора (точнее, его порта), передавшего сообщение. Эта информация заносится в таблицу маршрутизации (в которой уже имеются минимальные записи).

Например, после приема RIP-сообщений от маршрутизаторов R2 и R3 таблица маршрутизатора R1 примет вид, представленный в таблице 3.4.

Таблица 3.4 – Таблица маршрутизатора R1

Адрес сети назначения	Адрес порта следующего маршрутизатора	Адрес выходного порта	Метрика
192.168.1.0	-	192.168.1.1	1
192.168.2.0	-	192.168.2.1	1
192.168.1.0	192.168.1.2	192.168.1.1	2
192.168.3.0	192.168.1.2	192.168.1.1	2
192.168.3.0	192.168.2.2	192.168.2.1	2
192.168.2.0	192.168.2.2	192.168.2.1	2
192.168.4.0	192.168.2.2	192.168.2.1	2

Нетрудно заметить, что строки 3 и 4 были заполнены в результате получения информации от маршрутизатора R2, а строки 5-8 – в результате получения информации от маршрутизатора R3.

Затем маршрутизатор сравнивает принятую информацию с той, которая содержалась в его минимальной таблице. В нашем примере

информация о сети 192.168.1.0 содержится как в первой, так и в третьей строках, однако в строке 3 метрика больше, следовательно, эта строка удаляется. Аналогичным образом удаляется строка 6. В строках 4 и 5 содержатся данные о разных маршрутах к одной и той же сети – 192.168.3.0 – и с одним и тем же значением метрики. Поэтому в таблице сохраняется та запись, которая появилась раньше (например, строка 4).

После этого рассмотренные выше процедуры повторяются, только «соседям» рассылаются уже не минимальные таблицы, а таблицы с данными, полученными от других маршрутизаторов. Правило обработки полученной информации и внесения новых данных в таблицу остается прежним – запись о новом маршруте к уже известной сети производится в том случае, если метрика нового маршрута меньше метрики имеющегося маршрута.

В нашем простейшем примере новых записей в таблицу внесено не будет. Тем не менее, маршрутизаторы будут продолжать рассылку своих таблиц каждые 30 секунд, чем обеспечивается корректировка таблиц в случае изменения состояния сети.

Одним из важнейших понятий алгоритмов маршрутизации является время сходимости. Считается, что алгоритм «сошелся», когда все маршрутизаторы имеют согласованную информацию о доступных маршрутах. Время сходимости протокола RIP достаточно велико, поэтому в данном протоколе возможны возникновения петель маршрутизации, что приводит к «зацикливанию» пакетов. В настоящее время эта проблема решается путем введения дополнительных мер (например, использования метода «расщепления горизонта») и ограничением максимального значения метрики.

С другой стороны, данное ограничение не позволяет использовать RIP в крупных сетях. Кроме того, сама логика работы RIP приводит к существенному «засорению» сети служебным трафиком, так как таблицы передаются маршрутизаторами в полном объеме независимо от состояния сети.

Конфигурирование протокола RIP состоит из трех основных этапов:

- разрешение маршрутизатору исполнять протокол RIP;

- выбор версии протокола RIP;

- задание сетевых адресов и интерфейсов, которые должны включаться в пакеты обновления данных маршрутизации.

Первый этап реализуется путем выполнения на маршрутизаторе команды

router rip.

Выбор версии протокола (этап 2) реализуется путем выполнения субкоманды

version.

Отметим здесь, что существуют две версии протокола RIP. Версия 1 основана на классовой адресации, версия 2 способна использовать маски переменной длины в IP-адресации.

Наконец, для реализации третьего этапа используется субкоманда

network.

Данная команда должна использоваться для идентификации только тех сетевых адресов, которые непосредственно подключены к конфигурируемому маршрутизатору и предназначены для включения в пакеты обновления маршрутной информации. В пакеты обновления будут включаются только те интерфейсы, которые имеют IP-адреса, принадлежащие идентификационной сети.

Допустим, что имеется маршрутизатор, у которого интерфейсы имеют следующие адреса:

131.108.4.5/16

131.108.6.9/16

172.16.3.6/16

Исполнение команды

network 131.108.0.0

приведет к тому, что объявления с маршрутной информацией будут посылаться только с данными о подсетях сети 131.108.0.0 и только в

интерфейсы, которые адресуются в сети 131.108.0.0. Чтобы включить в пакеты обновления маршрутной информации интерфейс, находящийся в адресном пространстве 172.16.0.0, необходимо выполнить команду

network 172.16.3.6

Соберем сеть, состоящую из двух маршрутизаторов. Назначим интерфейсам маршрутизаторов те же сетевые адреса.

Произведем конфигурирование маршрутизатора Router0, рисунок 3.2.

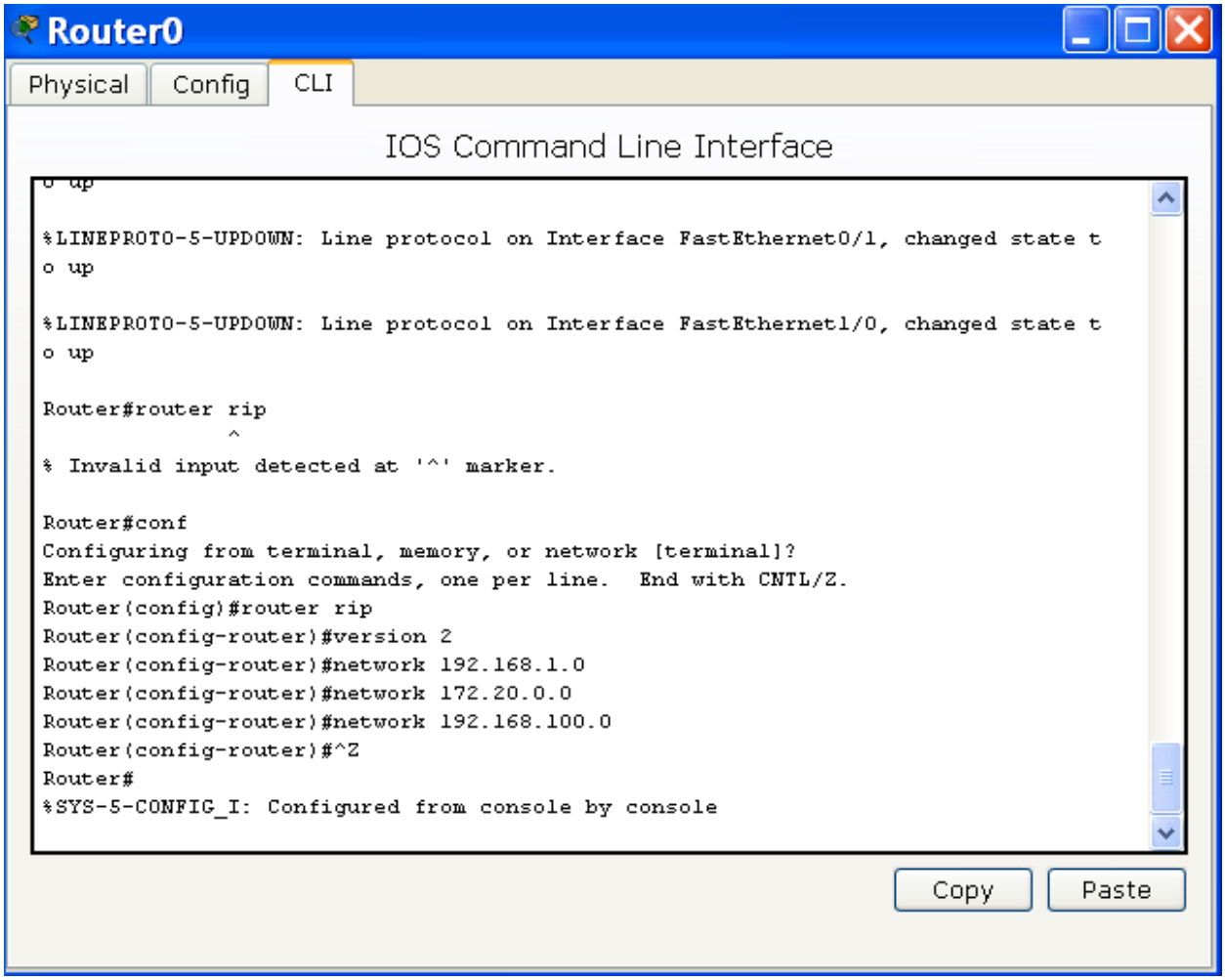

Рисунок 3.2 – Конфигурирование маршрутизатора

Аналогично произведем конфигурирование маршрутизатора Router1, рисунок 3.3.

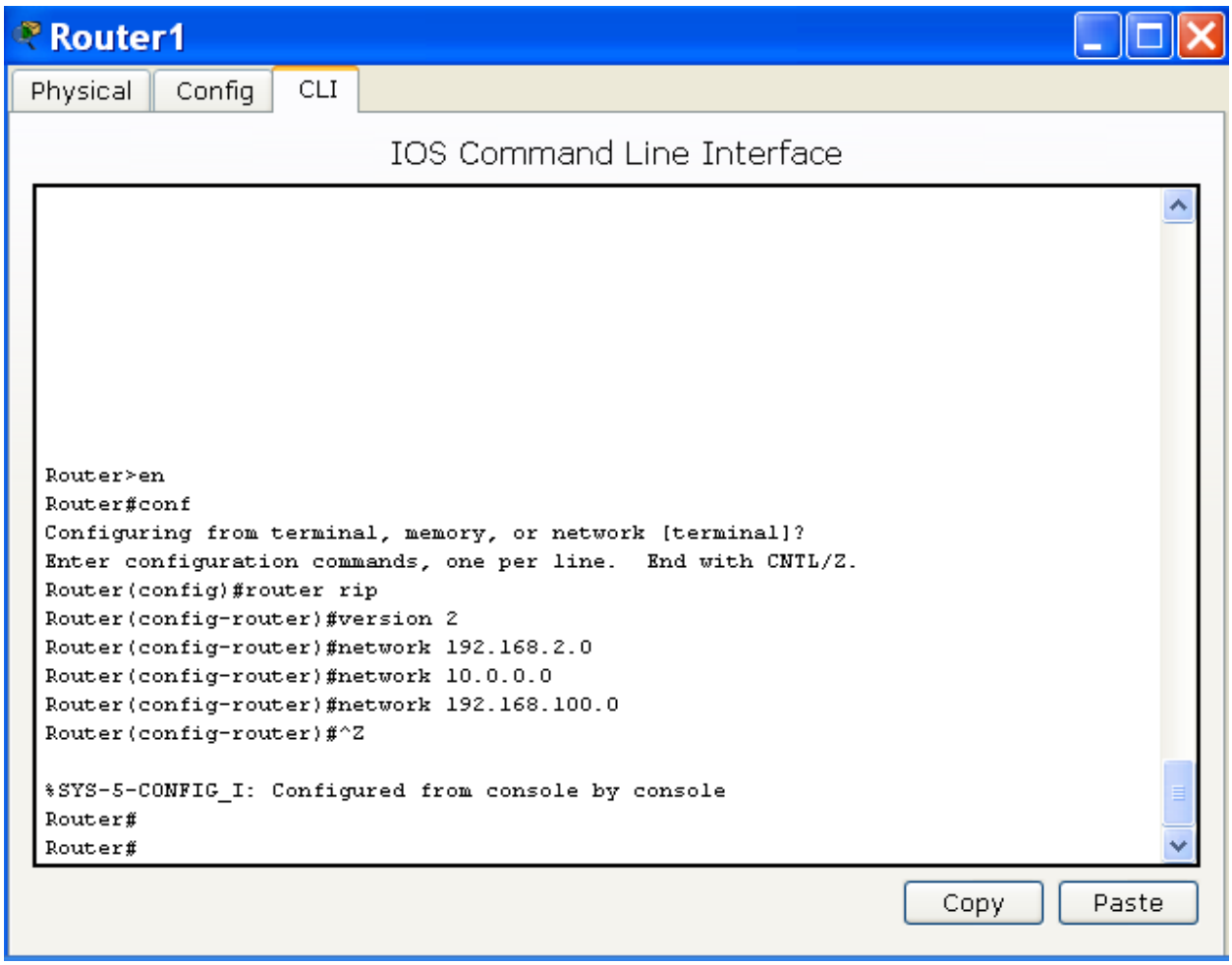

Рисунок 3.3 – Конфигурирование маршрутизатора Router1

Посмотрим таблицу маршрутизации маршрутизатора Router0, рисунок 3.4.

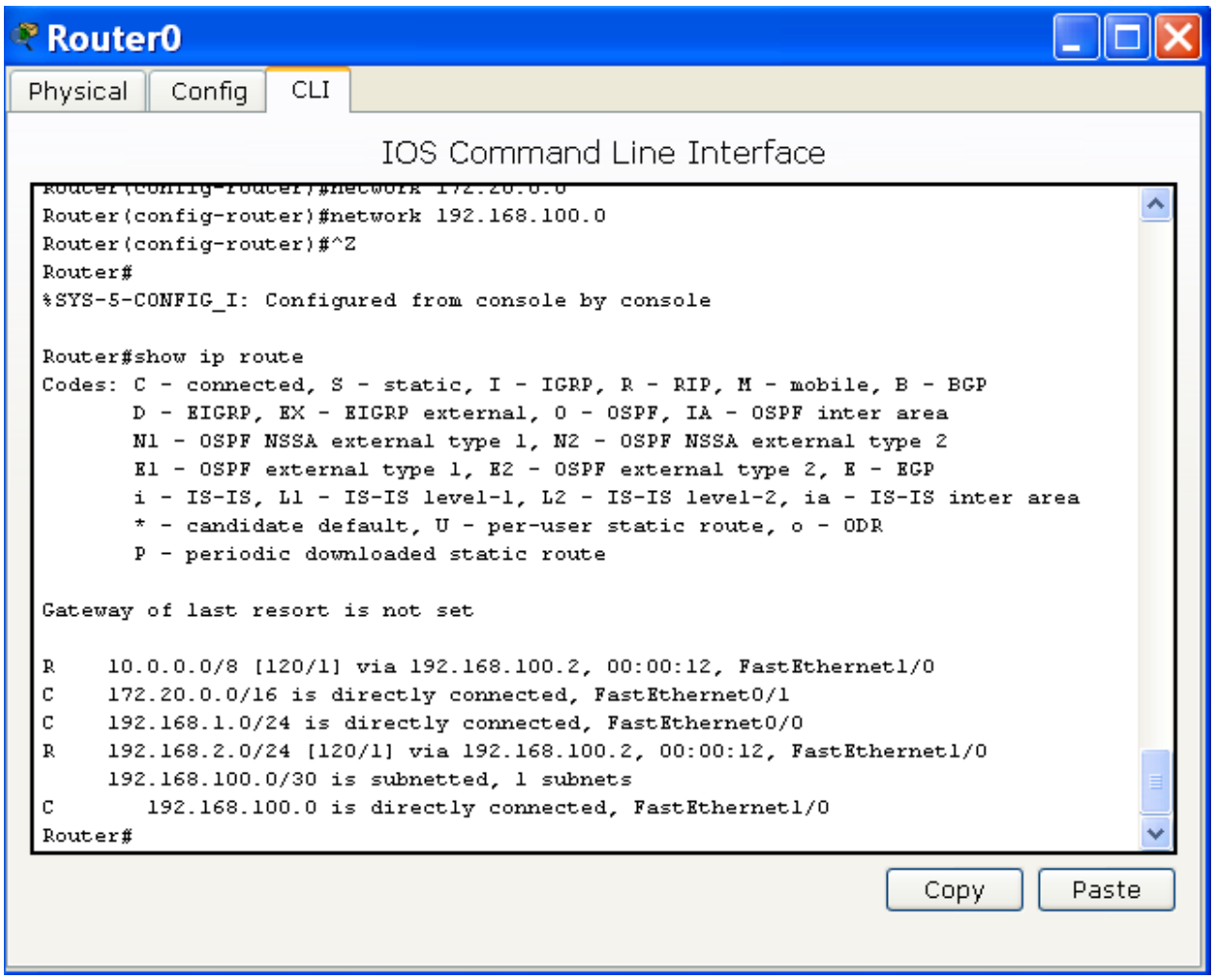

Рисунок 3.4 – Таблица маршрутизации Router0

Из рисунка видно, что в таблице появились две записи, помеченные как R (что означает протокол динамической маршрутизации RIP). Из таблицы следует, что в сеть 10.0.0.0 маршрут проходит через порт «правого» маршрутизатора 192.168.100.2 через собственный выходной порт fast Ethernet 1/0.

Таким образом, маршрутизация в составной сети настроена.

3.5 Отчет по работе:

- Работоспособная сеть.
- Результаты проверки связности сети.

Практическое занятие №4

Конфигурирование динамической маршрутизации (протокол OSPF)

4.1 Цель работы: Привитие навыков конфигурирования динамической маршрутизации с использованием протокола OSPF

4.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Linux.

4.3 Задание:

- Произвести физические соединения для построения заданной топологии;
- Сконфигурировать динамическую маршрутизацию на маршрутизаторе Cisco;
- Сконфигурировать динамическую маршрутизацию на программных маршрутизаторах;
- Произвести проверку связности сети с использованием утилит ping и traceroute.

4.4 Указания к проведению работы.

Протокол OSPF (Open Shortest Path First – выбор первого кратчайшего пути) относится к протоколам состояния связей и описан в RFC 1247 и RFC 2328.

Протокол OSPF позволяет либо задавать метрики произвольно, либо использовать метрику по умолчанию. В качестве метрики по умолчанию используется величина, обратно пропорциональная пропускной способности канала, через который проходит маршрут. Такое значение метрики позволяет более оптимально загрузить сеть. Например, если к пункту назначения имеются два маршрута, один из которых более длинный (например, через два промежуточных маршрутизатора), но с высокой пропускной способностью (например, 1 Гбит/с), а другой – более короткий (через один промежуточный маршрутизатор), но с низкой пропускной способностью (10 или 100 Мбит/с),

то меньшим значением метрики будет с высокой вероятностью характеризоваться первый маршрут.

Конкретное значение метрики по умолчанию в протоколе OSPF вычисляется как время передачи по каналу одного бита информации, деленное на 10 наносекунд. Например, в канале Fast Ethernet один бит передается за 10 наносекунд (10^{-8} с), соответственно, метрика будет равна 1. Для канала 10 Мбит/с метрика составит 10.

В протоколе OSPF каждый маршрутизатор строит описание сети в виде графа. В графе вершинами являются маршрутизаторы и подсети, а ребрами – связи между ними. Для отыскания оптимального пути на графе используется итерационный алгоритм Дийкстры, обладающий, с одной стороны, быстрой сходимостью, а с другой – вычислительной сложностью, значительно возрастающей при укрупнении сети. Поэтому при реализации протокола OSPF составная сеть разбивается на области (Area), в каждой из которых существует назначенный маршрутизатор (Designated Router – DR) и запасной назначенный маршрутизатор (Backup Designated Router – BDR). Соответственно, маршрутизаторы строят графы состояния связей внутри своей области, что уменьшает вычислительную сложность нахождения оптимальных (кратчайших) путей. В простейшем случае используется одна нулевая область (Area 0).

В отличие от протокола RIP, в протоколе OSPF не используются широковещательные рассылки для обновления сведений о состоянии связей. Вместо этого используются рассылки на Multicast-адрес 224.0.0.6, предназначенные назначенному и запасному назначенному маршрутизаторам, и рассылки на Multicast-адрес 224.0.0.5, предназначенные для остальных маршрутизаторов. Соответственно, при активизации на маршрутизаторе протокола OSPF, он автоматически становится членом группы многоадресной рассылки с адресом 224.0.0.5.

Для вычисления оптимальных маршрутов и занесения их в таблицу маршрутизации используется, как указывалось выше, алгоритм Дийкстры, в

качестве исходных данных для которого используется информация, хранящаяся в базе данных. Эта база содержит две таблицы:

- таблица соседних устройств;
- таблица топологии.

Для упрощенного описания процедур протокола OSPF рассмотрим сначала виды сообщений, которыми обмениваются между собой маршрутизаторы:

- сообщение Hello;
- сообщение описания базы данных Data Base Description (DBD);
- сообщение запроса Link-State Request (LSR);
- сообщение обновлений о состоянии связей Link-State Update (LSU);
- сообщение подтверждения Link-State Acknowledgment (LSAck).

Сообщения DBD и LSU могут содержать в себе одно или несколько сообщений Link-State Advertisement (LSA) – объявлений о состоянии связи одного топологического элемента (маршрутизатора, подсети или суммарного маршрута).

Все перечисленные сообщения инкапсулируются непосредственно в IP-пакет. Протоколы транспортных уровней TCP и UDP для передачи сообщений OSPF не используются.

Рассмотрим в несколько упрощенном виде процедуры протокола OSPF, предполагая, что в подсетях используется технология Ethernet (в соединениях маршрутизаторов «точка-точка» и при использовании технологий Frame Relay, ATM эти процедуры несколько отличаются).

При инициализации на маршрутизаторе протокола OSPF данный маршрутизатор сначала должен обнаружить своих «соседей». Это обнаружение производится путем обмена сообщениями Hello, в результате чего заполняется таблица соседних устройств, о которой было сказано выше. Зачастую этот этап называют установлением соседских отношений.

Каждый из маршрутизаторов OSPF характеризуется двумя параметрами:

- идентификатор (Router ID);

- приоритет (priority).

По умолчанию приоритет маршрутизатора имеет значение 1 и может быть изменен в пределах от 0 до 255. В качестве идентификатора может быть использован адрес одного из интерфейсов маршрутизатора, кроме того, он также может быть установлен вручную.

Таблица соседних устройств маршрутизатора представлена в таблице 4.1.

Таблица 4.1 – Структура таблицы соседства маршрутизатора OSPF

Идентификатор соседа (Neighbor ID)	Приоритет соседа (Priority)	Состояние (State)	Время до разрыва (Dead Time)	Адрес соседа (Address)	Интерфейс (Interface)

В поле «Состояние» заносится информация о состоянии соседских отношений. Таких состояний может быть семь:

- нерабочее (Down);

- инициализация (Init);

- двунаправленные отношения (Two-Way);

- выборы DR и BDR (Exstart);

- обмен (Exchange);

- загрузка (Loading);

- полные соседские отношения (Full).

Время до разрыва (Dead Time) – это временной интервал, по истечении которого будут разорваны соседские отношения, если от соседа не поступит ни одного сообщения OSPF.

Адрес соседа (Address) – это адрес сетевого уровня соседнего маршрутизатора.

Интерфейс (Interface) – собственный выходной интерфейс в направлении к соседу.

В процессе установления соседских отношений путем обмена сообщениями Hello маршрутизатор заполняет таблицу 2.5. При обмене сообщениями Hello, например, между двумя маршрутизаторами, после внесения в таблицу каждым из них сведений о своем соседе между ними устанавливаются двунаправленные отношения (Two-Way), соответствующая запись появляется в столбце 3 таблицы 4.1.

После установления двунаправленных отношений производится выбор DR и BDR на основании значений идентификаторов и приоритета. При этом сначала рассматриваются приоритеты, и в качестве DR выбирается маршрутизатор с наивысшим приоритетом. При равенстве приоритетов в качестве DR выбирается маршрутизатор с наивысшим значением идентификатора.

После заполнения таблицы соседства маршрутизаторы обмениваются известной им топологической информацией. Для передачи этой информации используется сообщение описания базы данных Data Base Description (DBD). Как указывалось выше, DBD содержит в себе один или несколько LSA. Сообщение LSA переносит информацию только об одном топологическом элементе – маршрутизаторе, подсети или суммарном маршруте. На каждый LSA отсылается подтверждение LSAck, чем обеспечивается гарантированная доставка топологической информации.

В процессе обмена сообщениями DBD маршрутизаторы находятся в состоянии обмена (Exchange), что отображается в третьем столбце таблицы 4.1.

Важно, что сообщение DBD переносит не всю топологическую информацию, а только список своей таблицы топологии (фактически, список топологических элементов, сведения о которых имеются у него в таблице). Когда маршрутизатор принимает DBD, он проверяет, содержит ли его таблица информацию о тех топологических элементах, список которых был получен в

сообщениях DBD. Если обнаруживается элемент, сведений о котором у данного маршрутизатора нет, он посылает запрос Link-State Request (LSR), в ответ на который сосед пересылает сообщение обновлений о состоянии связей Link-State Update (LSU), содержащий LSA с полной топологической информацией о запрашиваемом элементе. Данная информация заносится в таблицу топологий.

Таким образом, в результате обмена таблицы топологической информации у маршрутизаторов оказываются синхронизированы, что отражается состоянием Full в столбце 3 таблицы 4.1.

После перехода в это состояние маршрутизаторы запускают алгоритм Дийкстры, в результате реализации которого заполняются таблицы маршрутизации.

Помимо обработки сообщений LSA, для пополнения своих таблиц маршрутизаторы производят их ретрансляцию.

В процессе нормального функционирования каждый маршрутизатор генерирует сообщение Hello с интервалом в 10 секунд. Если от какого-либо маршрутизатора не поступает это сообщение, это говорит об изменении состояния связи. Маршрутизатор в этом случае вносит изменение в свою таблицу топологии и рассылает это изменение с использованием сообщений LSA. Соответственно, каждый маршрутизатор, приняв данное сообщение, вносит коррективы в свои таблицы и рассылает сообщение дальше. То же самое происходит при появлении в сети нового маршрутизатора после получения от него сообщения Hello.

Внеся изменения в свои таблицы топологии, каждый из маршрутизаторов применяет алгоритм Дийкстры для отыскания оптимальных маршрутов и заносит их в таблицы маршрутизации.

Таким образом, если в сети не происходит изменений, маршрутизаторы обмениваются только сообщениями Hello, а достаточно объемные обновления, передаваемые сообщениями LSA, производятся только при изменении состояния сети. Однако для повышения надежности

маршрутизаторы все-таки обмениваются своими базами, но это происходит достаточно редко – по умолчанию раз в 30 минут.

Такое снижение нагрузки на сеть является несомненным достоинством протокола OSPF. Однако применение маршрутизатором достаточно сложного алгоритма Дийкстры требует использования высокопроизводительных процессоров.

Дополнительное снижение нагрузки на сеть при использовании протокола OSPF обеспечивается тем, что в отношении Full маршрутизатор находится только с DR и BDR, с остальными маршрутизаторами поддерживается отношение Two-Way.

Кроме того, протокол OSPF способен распараллеливать передачу данных по нескольким маршрутам, если они имеют одинаковую метрику.

Наконец, как указывалось выше, протокол OSPF поддерживает разделение составной сети на области (Area). Обмен топологической информацией происходит только внутри области, что также снижает нагрузку на сеть.

Проведем настройку протокола OSPF на маршрутизаторе Router1.

Войдем в конфигурации в консоль роутера и выполним следующие настройки (при вводе команд маску подсети можно не указывать, т.к. она будет браться автоматически из настроек интерфейса роутера):

Вход в привилегированный режим:

Switch>**en**

Вход в режим конфигурации:

Switch1#**conf t**

Вход в режим конфигурирования протокола OSPF:

Router1(config)#**router ospf 1**

В команде router ospf <идентификатор_процесса> под идентификатором процесса понимается уникальное числовое значение для каждого процесса роутинга на маршрутизаторе. Данное значение должно быть

больше в интервале от 1 до 65535. В OSPF процессам на роутерах одной зоны принято присваивать один и тот же идентификатор.

Подключение клиентской сети к роутеру:

Router1(config-router)#**network 10.11.0.0**

Подключение второй сети к роутеру:

Router1(config-router)#**network 10.10.0.0**

Задание номера версии протокол OSPF:

Router1(config-router)#**version 2**

Выход из режима конфигурирования протокола OSPF:

Router1(config-router)#**exit**

Выход из консоли настроек:

Router1(config)#**exit**

Сохранение настроек в память маршрутизатора:

Switch1#**write memory**

Аналогично необходимо произвести настройку протокола OSPF на маршрутизаторе Router2.

4.5 Отчет по работе:

- Работоспособная сеть.
- Результаты проверки связности сети.

Практическое занятие № 5

Конфигурирование динамической маршрутизации (протокол EIGRP)

5.1 Цель работы: Привитие навыков конфигурирования динамической маршрутизации с использованием протокола EIGRP

5.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Linux.

5.3 Задание:

- Произвести физические соединения для построения заданной топологии;
- Сконфигурировать динамическую маршрутизацию на маршрутизаторе Cisco;
- Сконфигурировать динамическую маршрутизацию на программных маршрутизаторах;
- Произвести проверку связности сети с использованием утилит ping и traceroute.

5.4 Указания к проведению работы.

В свое время широко использовался протокол IGRP (Interior Gateway Routing Protocol – протокол маршрутизации внутреннего шлюза). Данный протокол также являлся дистанционно-векторным и применялся исключительно в оборудовании Cisco Systems. Затем вышла усовершенствованная версия протокола – EIGRP (Enhanced Interior Gateway Routing Protocol), по существу, превратившая его в гибридный протокол маршрутизации. EIGRP используется не только в оборудовании Cisco, но и в оборудовании других производителей. Так как в настоящее время повсеместно вместо IGRP используется протокол EIGRP, именно его и рассмотрим.

В отличие от рассмотренных выше протоколов маршрутизации, EIGRP использует достаточно сложную обобщенную метрику, параметры которой, помимо всего прочего, могут настраиваться при конфигурировании сети.

Для вычисления значения обобщенной метрики в EIGRP используются частные значения метрик, к которым относятся:

- пропускная способность канала между получателем и отправителем (BW);

- задержка (D);

- надежность (R);

- нагрузка (L).

Формула для вычисления обобщенной метрики имеет вид

$$M = \left(k1 \cdot BW + \frac{k2 \cdot BW}{256 - L} + k3 \cdot D \right) \cdot \frac{k5}{R + k4},$$

где $k1...k5$ - весовые коэффициенты.

При этом частные метрики можно разделить на два вида - статические (не зависящие от состояния сети BW и D) и динамические (зависящие от состояния сети R и L).

По умолчанию в протоколе весовым коэффициентам присвоены значения:

$$k1 = k3 = 1,$$

$$k2 = k4 = k5 = 0.$$

Соответственно, значение обобщенной метрики по умолчанию вычисляется по формуле:

$$M = BW + D.$$

Как следует из последней формулы, по умолчанию протокол учитывает только статические метрики, определяемые типом канала связи и интерфейсом. Сами статические метрики рассчитываются по формулам:

$$BW = \left(\frac{10^7}{bw} \right) \cdot 256,$$

где bw - полоса пропускания, заданная на интерфейсе, кбит/с;

$$D = \left(\frac{d}{10} \right) \cdot 256,$$

где d - задержка, определяемая типом интерфейса, мкс.

Для стандартных интерфейсов значения d определяются следующим образом – для Fast Ethernet $d = 100$ мкс, для Ethernet $d = 1000$ мкс и т.д.

При этом метрика составного маршрута определяется исходя из пропускной способности самого «медленного» канала, входящего в данный маршрут, и суммарной задержки всех интерфейсов, через который проходит маршрут.

В результате обмена маршрутной информацией маршрутизаторы EIGRP подобно маршрутизаторам OSPF формируют базу данных, в которую входят таблица соседства и таблица топологии. Однако алгоритм, по которому вычисляется оптимальный маршрут, качественно другой. Алгоритм, используемый EIGRP, называется DUAL (Diffusing Update Algorithm – алгоритм диффузионного обновления).

Для пополнения своей базы данных маршрутизаторы EIGRP обмениваются между собой пятью типами пакетов:

- пакет обновлений Update;
- пакет запроса Query;
- пакет ответа на запрос Reply;
- пакет приветствия Hello;
- пакет подтверждения Ack.

Данные пакеты аналогично протоколу OSPF используют групповой адрес 224.0.0.10 или индивидуальные адреса соседних маршрутизаторов. Так как протокол EIGRP является внутренним (Interior), он функционирует только в пределах так называемой автономной системы (понятие автономной системы более подробно будет рассмотрено в следующем параграфе). Соответственно, перечисленные выше пакеты содержат номер автономной системы. Маршрутизатор EIGRP, приняв пакет, обрабатывает его только в том случае, если его передал маршрутизатор, относящийся к этой же автономной системе.

Аналогично маршрутизатору OSPF, маршрутизатор EIGRP при инициализации должен установить соседские отношения. Эти отношения

устанавливаются путем обмена пакетами Hello, в результате которого каждый маршрутизатор, устанавливающий соседские отношения, заполняет свою таблицу соседства. Поля таблицы соседства маршрутизаторов EIGRP представлены в таблице 5.1.

Таблица 5.1 – Таблица соседства

Номер соседа H	Адрес соседа Address	Собственный интерфейс Interface	Время удержания Holdtime	Время отношений Uptime	Счетчик очереди Count	Последовательный номер Seq Num	Таймер цикла SRTT	Таймер повторной передачи RTO

В отличие от протокола OSPF, в EIGRP используется порядковый номер соседнего маршрутизатора (H). В поле Address указывается его сетевой адрес, а в поле Interface – собственный интерфейс, через который доступен сосед.

Время удержания (Holdtime) – это интервал времени, по истечении которого, в случае отсутствия EIGRP-пакетов от соседнего маршрутизатора, он считается недостижимым. По умолчанию данный интервал равен 15 с.

Время отношений (Uptime) – это интервал времени, прошедший с момента установления соседских отношений.

Счетчик очереди (Count) – это число пакетов, находящихся в очереди передачи.

Последовательный номер (Seq Num) – номер последнего EIGRP-пакета, принятого от соседа. Используется как для контроля порядка приема пакетов, так и для передачи пакета подтверждения.

Таймер цикла (SRTT) – временной интервал, необходимый для отправления соседу пакета и получения от него ответа. Если в течение этого интервала ответ не приходит, пакет отправляется повторно.

Таймер повторной передачи (RTO) – интервал, в течение которого ожидается подтверждение от соседа о приеме им пакета. Если в течение этого интервала подтверждения не приходит, пакет отправляется повторно.

Рассмотрим сеть, представленную на рисунке 5.1.

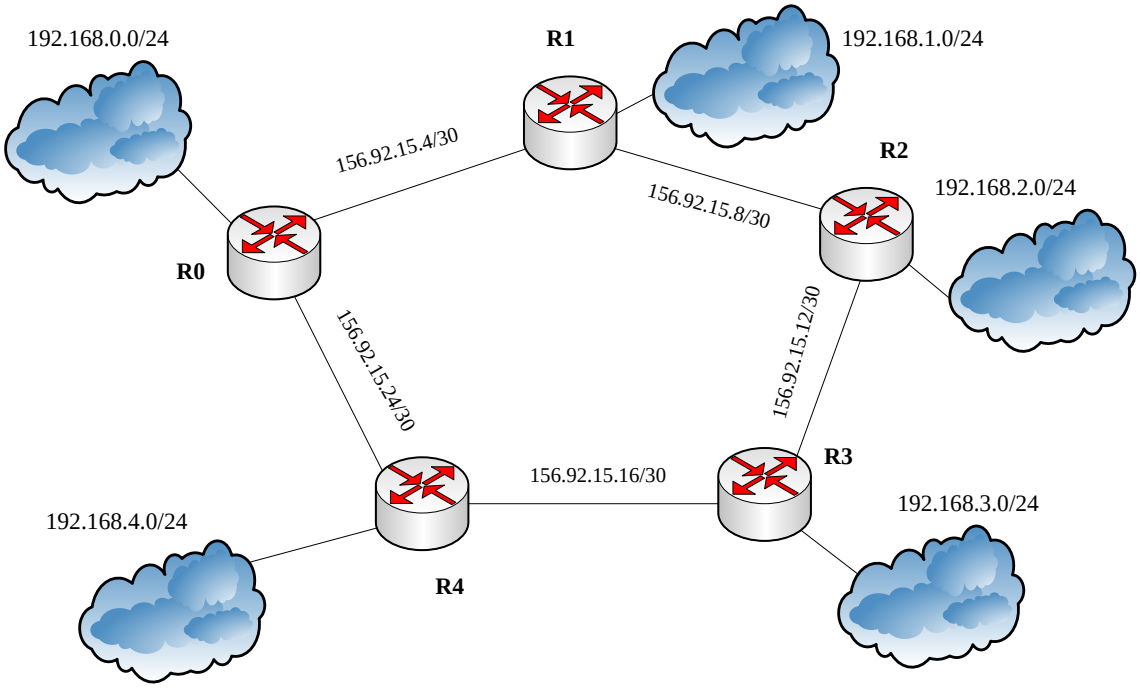

Рисунок 5.1 – Пример сети

Из рисунка видно, что маршрутизаторы связаны друг с другом с использованием вырожденной подсети «точка-точка», кроме того, к каждому из них подключена своя подсеть. Например, маршрутизатор R1 связан с подсетью 192.168.1.0/24, с использованием интерфейса с адресом 192.168.1.1/24. С маршрутизатором R0 он связан через вырожденную подсеть 156.92.15.4/30 с использованием интерфейса с адресом 156.92.15.6/30, а с маршрутизатором R2 – через вырожденную подсеть 156.92.15.8/30 с использованием интерфейса с адресом 156.92.15.6/30.

Вид таблицы соседства маршрутизатора Cisco R1 представлен на рисунке 5.2 (используется команда show ip eigrp neighbor).

```
Router#sh ip eigrp neighbor
IP-EIGRP neighbors for process 200
H   Address         Interface       Hold Uptime    SRTT    RTO   Q   Seq
                                    (sec)          (ms)        Cnt Num
0   156.92.15.5     Fa0/0           13   00:07:52  40      1000  0   15
1   156.92.15.10    Fa0/1           11   00:06:19  40      1000  0   13

Router#
```

Рисунок 5.2 – Таблица соседства маршрутизатора Cisco

В ответ на пакет Hello маршрутизатор отправляет пакет Update по индивидуальному адресу маршрутизатора, от которого был принят пакет Hello (при этом в пакете Update содержится подтверждение получения пакета Hello, что снижает необходимый объем пакетов подтверждений Ack).

Таким образом, в результате такого обмена маршрутизаторы EIGRP заполняют свои таблицы соседства. После установления соседских отношений маршрутизаторы обмениваются между собой своими таблицами топологии. Таблица топологии того же маршрутизатора Cisco (R1, рисунок 5.1) представлена на рисунке 5.3 (используется команда show ip eigrp topology).

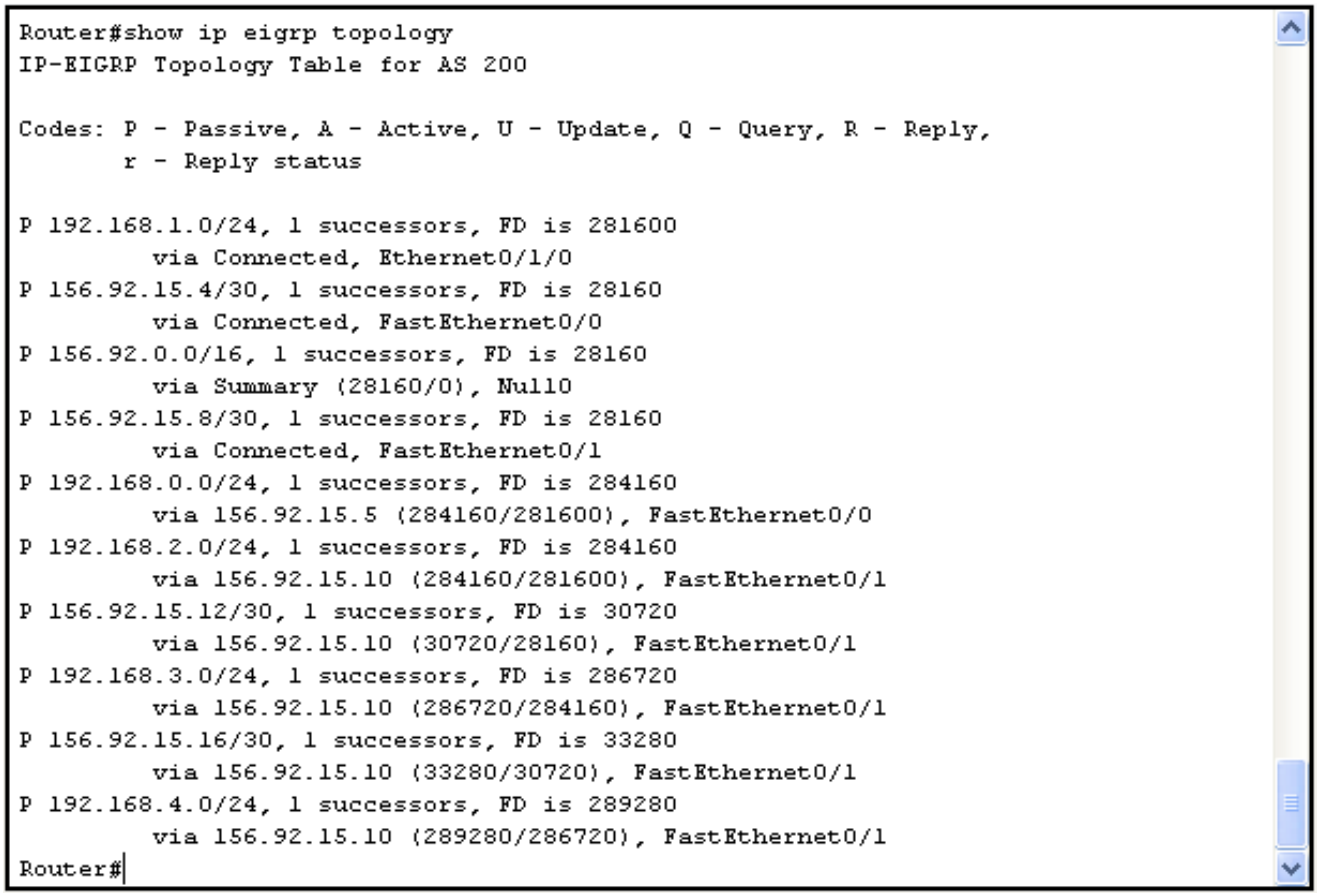

Рисунок 5.3 – Таблица топологии маршрутизатора Cisco

Рассмотрим состав таблицы топологии.

В первой колонке таблицы представлен статус маршрута. Символ P (Passive) обозначает пассивный, то есть готовый к использованию маршрут. Символ A (Active) обозначает активный маршрут, то есть маршрут, по которому не закончен расчет по алгоритму DUAL. Как следует из рисунка 5.3, все маршруты в данном примере пассивные.

После указания состояния маршрута указываются подсеть назначения и число преемников (Successors). Под преемником понимается первичный маршрут, который после вычисления передается в таблицу маршрутизации.

FD (Feasible Distance) обозначает выполнимое расстояние (метрику) до сети назначения по данному маршруту. Выполнимым расстоянием называется полная метрика маршрута, которая вычисляется как сумма заявленного расстояния (то есть полученного от соседа) и расстояния до соседа.

После via указывается источник маршрута. Например, запись via Connected означает, что подсеть подключена непосредственно к данному маршрутизатору. Запись via 156.92.15.5 (284160/281600) означает, что маршрут был анонсирован маршрутизатором с адресом выходного интерфейса 156.92.15.5 с заявленным состоянием 281600, выполнимое расстояние составляет 284160.

Наконец, последняя запись означает выходной интерфейс данного маршрутизатора.

Кроме понятия преемника, в протоколе EIGRP существует понятие вероятного преемника – резервного маршрута, который задействуется при отказе маршрутизатора-преемника. Вероятный преемник не заносится в таблицу маршрутизации, но хранится в таблице топологии.

Протокол EIGRP по умолчанию способен распараллеливать передачу данных по маршрутам с равной метрикой, а после соответствующей настройки – и по маршрутам с различными метриками.

Имея в своей базе таблицу топологий, каждый из маршрутизаторов в соответствии с алгоритмом DUAL рассчитывает оптимальные маршруты и заносит их в таблицу маршрутизации. Пример таблицы маршрутизации, соответствующей таблице топологии (рисунок 5.3) представлен на рисунке 5.4.

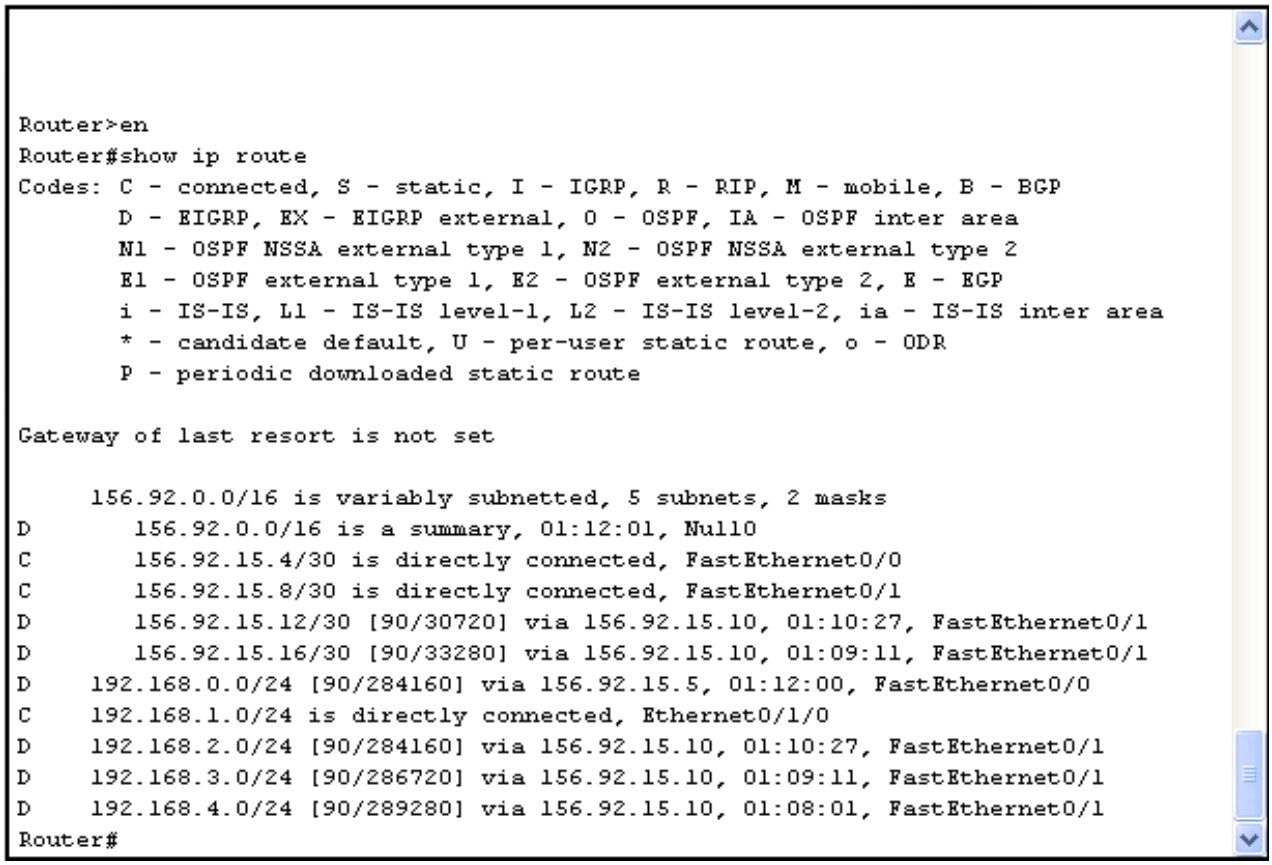

Рисунок 5.4 – Таблица маршрутизации маршрутизатора EIGRP

Таким образом, на данном занятии необходимо собрать схему транспортной мультисервисной сети, состоящей из маршрутизаторов, сконфигурировать на каждом из маршрутизаторов динамическую маршрутизацию, проверить связность сети с использованием утилит ping и traceroute.

5.5 Отчет по работе:

- Работоспособная сеть;
- Результаты проверки связности сети.

Практическое занятие №6

Конфигурирование сети доступа

6.1 Цель работы: Привитие навыков конфигурирования сети доступа, построенной на базе технологии Ethernet

6.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Linux.

6.3 Задание:

- Произвести физические соединения для построения заданной топологии;
- Сконфигурировать коммутаторы доступа для обеспечения возможности работы в сети;
- Произвести проверку связности сети с использованием утилит ping и traceroute.

6.4 Указания к проведению работы.

Коммутаторы Cisco могут находиться в одном из режимов, представленных в таблице 6.1.

Таблица 6.1 – Режимы конфигурирования

Название режима	Приглашение (Prompt)	Описание
User EXEC Mode	Switch>	Пользовательский режим
Privileged EXEC Mode	Switch #	Привилегированный режим
Global Configuration Mode	Switch (config)#	Режим глобального конфигурирования

В таблице 6.1 в первом столбце представлены названия режимов, во втором – приглашение, отображаемое в командной строке, в третьем – описание.

Пользовательский режим (user mode) используется для просмотра состояния устройства, а также для перехода в привилегированный режим (privileged mode). Никаких изменений в конфигурационном файле, в том числе удаление и сохранение текущей конфигурации, в пользовательском режиме производиться не может. В этом режиме доступны только некоторые команды верификации show, т. е. команды просмотра состояния устройства.

Для перехода в привилегированный режим необходимо выполнить команду

enable

Следует отметить, что оборудование Cisco допускает сокращенный ввод команд, если невозможно ее двоякое толкование. Например, вместо enable можно набрать en, и эта команда будет понятна коммутатору, так как никакая другая команда не начинается с сочетания символов en.

Выполнение каждой команды начинается после нажатия клавиши Enter. Для повтора ранее введенных команд можно использовать клавишу ↑.

В привилегированном режиме доступны все команды show, возможно удаление конфигурации и сохранение конфигурационного файла в памяти NVRAM. Возврат в пользовательский режим производится командой disable или exit:

Switch#exit

Команда exit позволяет вернуться на один уровень вверх. Например, после выполнения команды в режиме глобального конфигурирования коммутатор переходит в привилегированный режим. Если необходимо из любого состояния устройства выйти сразу в пользовательский режим, используется комбинация клавиш CTRL-Z.

В глобальном режиме производятся изменения, которые затрагивают коммутатор в целом, поэтому этот режим и называется global configuration mode. Например, в нем можно устанавливать имя коммутатора командой hostname. Имя коммутатора не имеет значения в сети, оно удобно при конфигурировании. Пример:

Switch(config)#hostname Switch_A

Switch_A(config)#

В режиме глобального конфигурирования на коммутатор можно устанавливать пароли. Существует несколько видов паролей для обеспечения защиты устройств Cisco. Первые два пароля, enable secret и enable password, используются для обеспечения авторизованного входа в привилегированный режим. На коммутаторе устанавливается один (или оба) из этих паролей. После установки пароля система запрашивает его у пользователя, когда вводится команда enable. Формат команд установки паролей «cisco» и «cisco1» для входа в привилегированный режим приведен ниже:

Switch_A(config)#enable secret cisco

Switch_A(config)#enable password cisco1

Пароль enable secret криптографируется по умолчанию, поэтому является более строгим. Если установлены оба пароля – enable secret и enable password, – то в приведенном примере система будет реагировать на пароль cisco. Пароль enable password по умолчанию не криптографируется, поэтому его можно посмотреть, например, по команде show running-configuration (сокращенно sh run), которая выполняется из привилегированного режима.

Изменение и создание конфигурации коммутатора Cisco возможно в режиме глобального конфигурирования, вход в который реализуется из привилегированного по команде configure terminal (сокращенно – conf), которая вводит устройство в глобальный режим и позволяет изменять текущую конфигурацию (running-config). При этом приглашение изменяет вид на Switch(config)#:

Switch >ena

Switch #conf t

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

Switch(config)#

Для просмотра адресной таблицы используется команда

show mac address-table

Пример выполнения команды show mac address-table показан на рисунке 6.1.

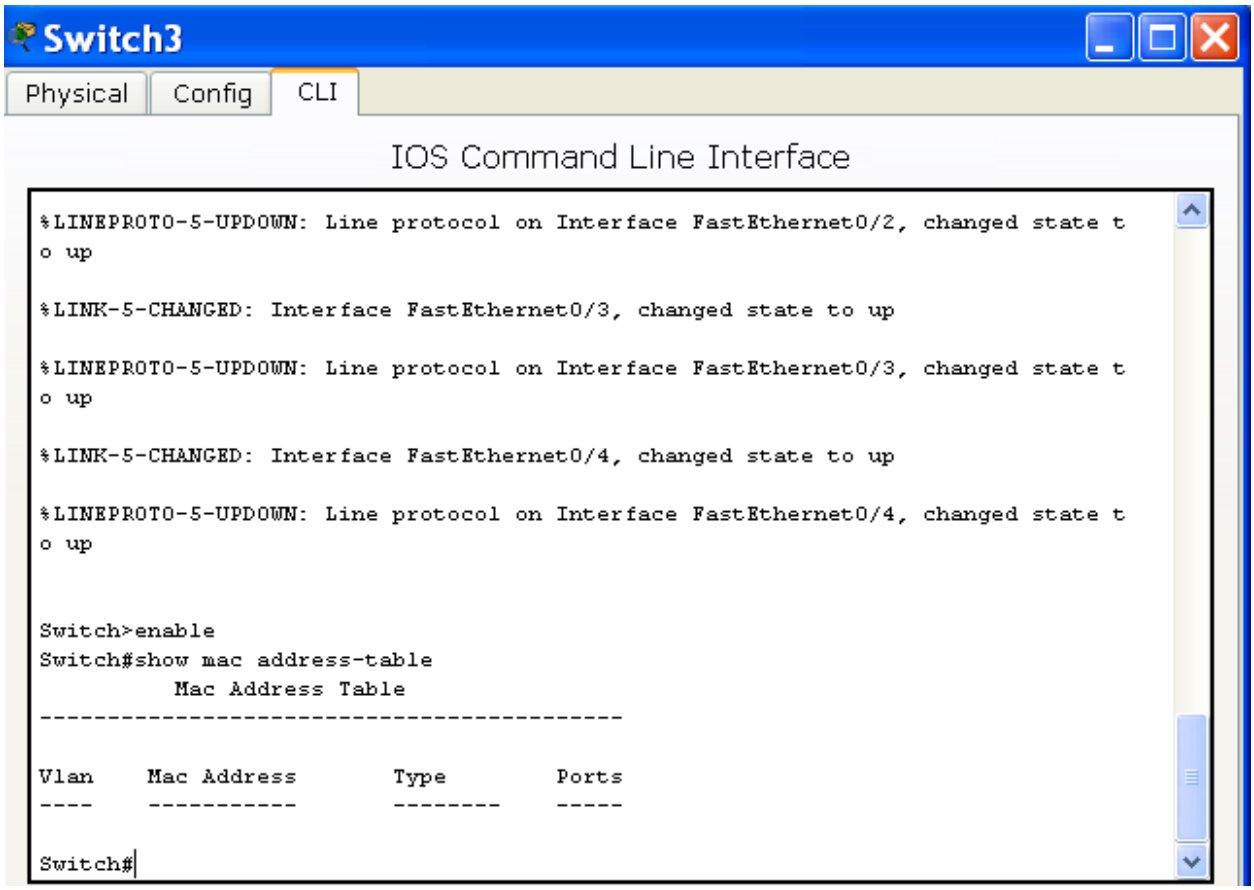

Рисунок 6.1 – Результат выполнения команды show mac address-table

Как видно из рисунка 6.1, адресная таблица пуста, в ней нет ни статических записей (мы их не вносили), ни динамических (коммутатор не прошел процедуру обучения). Для прохождения процедуры обучения достаточно передать от каждого из PC любые кадры, например, ICMP-запросы с использованием утилиты **ping**. Если после этого опять просмотреть адресную таблицу, то она примет вид, показанный на рисунке 6.2.

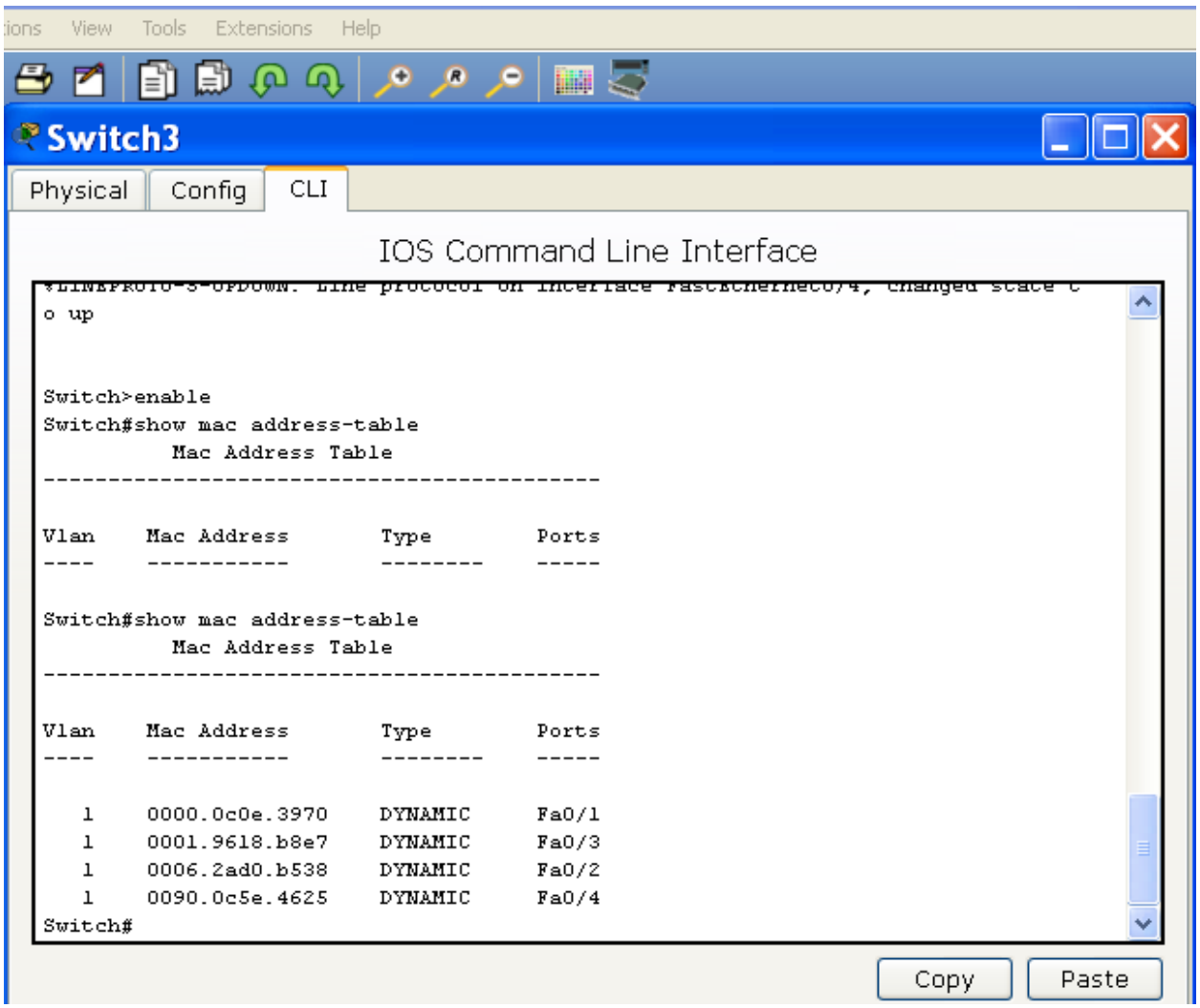

Рисунок 6.2 – Результат выполнения команды show mac address-table

Как видно, таблица изменилась. В первом ее столбце указан номер VLAN, MAC-адрес рабочей станции, тип записи (динамическая – Dynamic), и номер интерфейса коммутатора, с которого доступна рабочая станция.

Виртуальные локальные сети VLAN – это технология, позволяющая организовывать несколько независимых виртуальных сетей внутри одной физической сети. С помощью VLAN можно выполнять гибкое разнесение пользователей по различным сегментам сети с разной адресацией, даже если они подключены к единому устройству, а также дробить широковещательные домены.

Принцип организации двух VLAN на одном коммутаторе иллюстрируется рисунком 6.3.

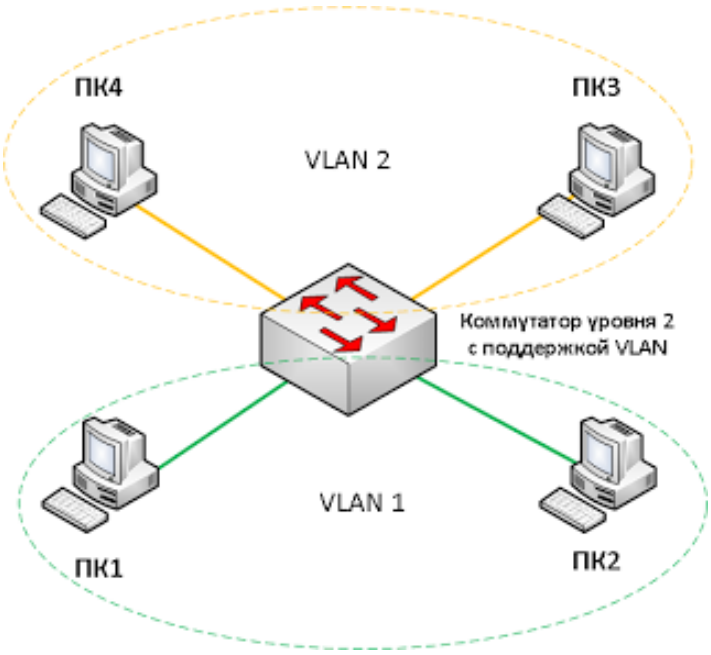

Рисунок 6.3 – Организация двух VLAN на одном коммутаторе

Компьютеры 1 и 2 объединены в одну VLAN, компьютеры 3 и 4 объединены в другую VLAN. Хотя все компьютеры подключены к одному и тому же коммутатору, все они не смогут общаться между собой. Компьютер номер 1 сможет общаться только с компьютером 2, компьютер 3 будет видеть только компьютер 4. То есть данная ситуация будет аналогична тому, как если бы мы подключили компьютеры 1 и 2 к одному коммутатору, а компьютеры 3 и 4 к другому коммутатору, и не соединили бы эти коммутаторы между собой. Как легко заметить, в данном случае, технология VLAN помогла нам разделить единую физическую сеть на несколько виртуальных не связанных между собой сетей, при этом компьютеры, находящиеся в этих виртуальных сетях, работают точно так же, как это было бы в обычной сети.

Для взаимодействия устройств в VLAN сетях их порты настраиваются специальным образом. Существует два типа настройки портов: настройка порта в режиме доступа (Access Mode) и настройка порта в режиме магистрали (Trunk Mode).

Порты доступа применяются обычно для подключения оконечных устройств. В простейшем случае, порту доступа задается определенная VLAN, и он передает весь поступающий на него трафик именно в нее. Порты, к

которым подключены компьютеры 1,2,3 и 4 на рисунке 7.4, являются портами доступа.

Магистральные порты предназначены для передачи трафика сразу нескольких VLAN и обычно используются для соединения сетевых устройств между собой. Порты 5 обоих коммутаторов на рисунке 6.4, являются магистральными портами.

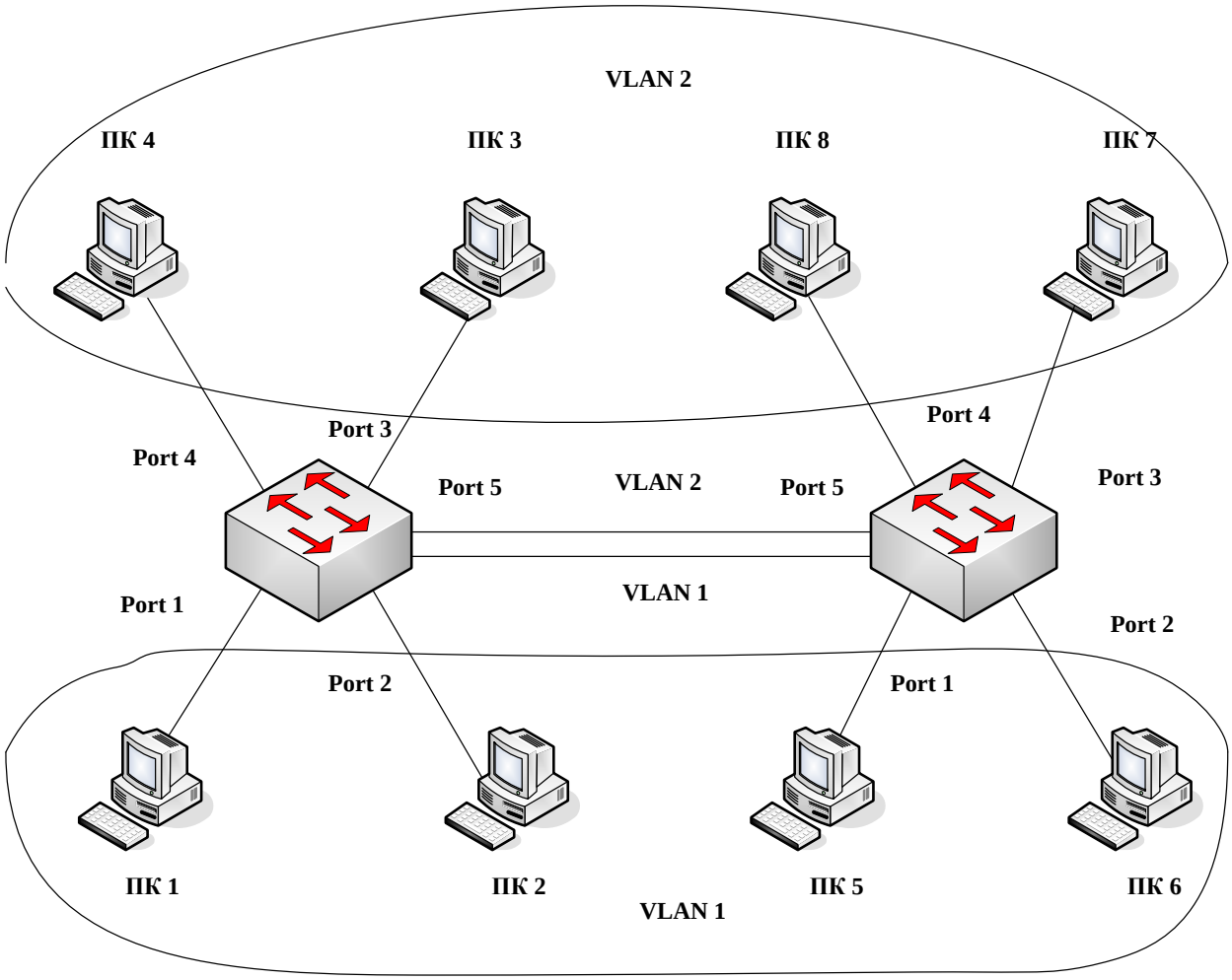

Рисунок 6.4 – Использование магистральных портов и портов доступа

На данном рисунке порты 1 – 4 работают в режиме доступа, порт 5 – в режиме магистрали и передает через себя трафик сразу двух виртуальных локальных сетей VLAN 1 и VLAN 2 (поэтому условно на рисунке порты 5 обоих коммутаторов связаны между собой двумя линиями, хотя физически это – одна линия).

Передавать трафик VLAN между коммутаторами можно не только с помощью магистральных портов, но и с помощью портов доступа, но так как порты доступа могут пропускать трафик только одной VLAN, для соединения устройств между собой потребуется выделение портов, количество которых будет равно количеству передаваемых между устройствами VLAN. Данный способ обычно находит применение только в том случае, если между устройствами необходимо передать трафик небольшого числа VLAN.

Для настройки VLAN необходимо перейти в привилегированный режим, выполнив команду enable. Информацию о существующих на коммутаторе VLAN можно, выполнив команду show vlan brief (можно просто sh vl br).

Рассмотрим пример создания двух VLAN на одном коммутаторе. Перейдем в привилегированный режим, выполнив команду enable, и просмотрим информацию о существующих на коммутаторе VLAN (рисунок 6.5).

В результате выполнения команды на экране появится: номера VLAN – первый столбец; название VLAN - второй столбец; состояние VLAN (работает она в данный момент или нет) – третий столбец; порты, принадлежащие к данной VLAN – четвертый столбец. Как мы видим, по умолчанию на коммутаторе существует пять VLAN . Все порты коммутатора по умолчанию принадлежат VLAN 1. Остальные четыре VLAN являются служебными и используются не очень часто.

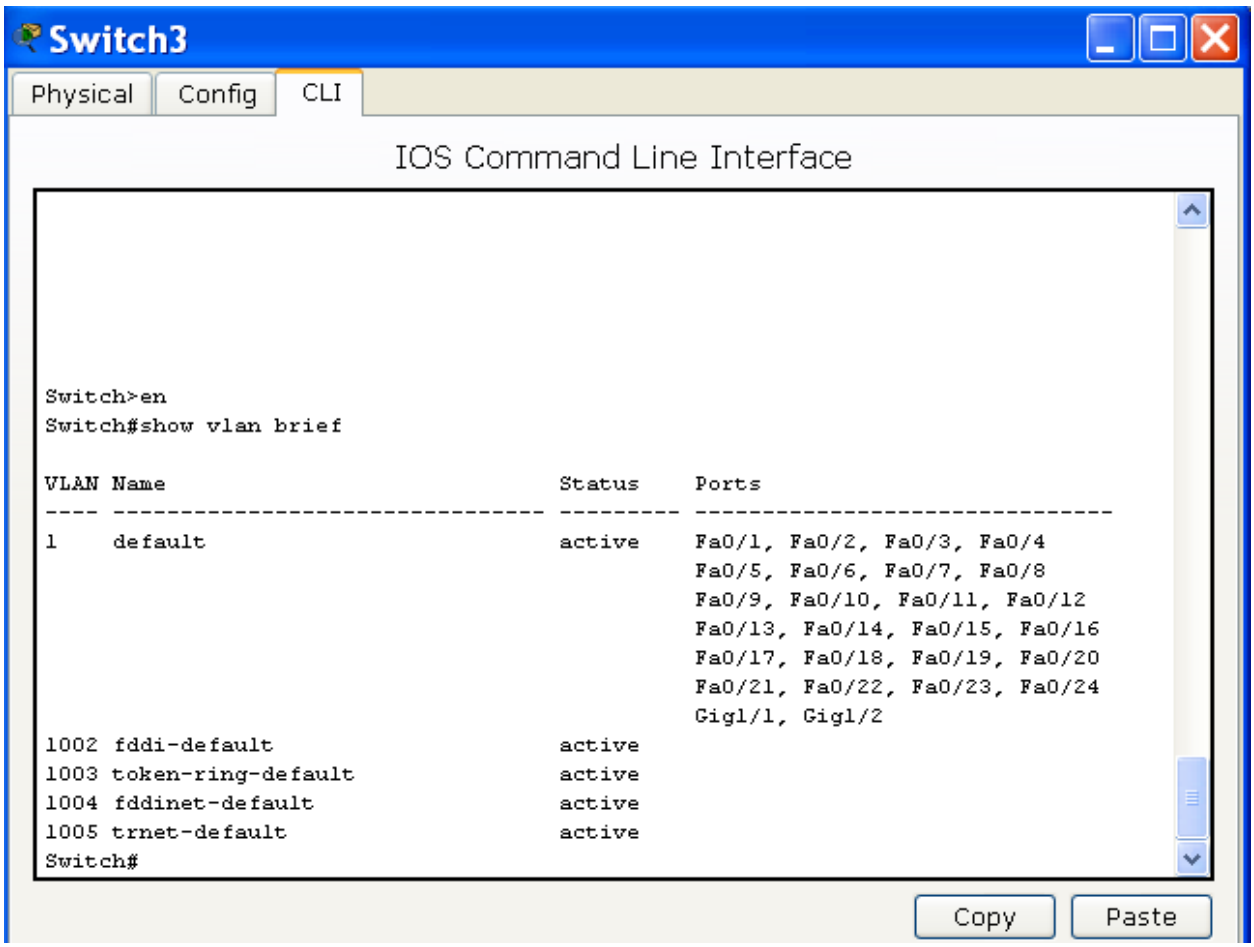

Рисунок 6.5 – Просмотр конфигурации VLAN

Для реализации сети, которую мы запланировали сделать, создадим на коммутаторе еще две VLAN. Для этого в привилегированном режиме необходимо выполнить команду conf t для перехода в режим глобального конфигурирования. Вводим команду vlan 2. Данной командой создается на коммутаторе VLAN с номером 2. Указатель ввода Switch(config)# изменится на Switch(config-vlan)#, это свидетельствует о том, что конфигурируется уже не весь коммутатор в целом, а только отдельная VLAN, в данном случае номер 2. Если использовать команду «vlan x», где x номер VLAN, когда VLAN x еще не создана на коммутаторе, то она будет автоматически создана и будет осуществлен переход к ее конфигурированию.

Для решения поставленной задачи коммутатор необходимо сконфигурировать следующим образом.

Switch(config)#vlan 2

```
Switch(config-vlan)#name subnet_192
Switch(config)#interface range fastEthernet 0/1-2
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 2
```

Разберем данную конфигурацию. Как уже говорилось ранее, командой vlan 2 мы создаем на коммутаторе новую VLAN с номером 2. Команда name subnet_192 присваивает имя subnet_192 виртуальной сети номер 2. Выполняя команду interface range fastEthernet 0/1-2, мы переходим к конфигурированию интерфейсов fastEthernet 0/1 и fastEthernet 0/2 коммутатора. Ключевое слово range в данной команде указывает на то, что мы будем конфигурировать не один единственный порт, а целый диапазон портов, в принципе ее можно не использовать, но тогда последние три строки придется заменить на:

```
Switch(config)#interface fastEthernet 0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 2
Switch(config)#interface fastEthernet 0/2
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 2
```

Команда switchport mode access конфигурирует выбранный порт коммутатора, как порт доступа. Команда switchport access VLAN 2 указывает, что данный порт является портом доступа для VLAN номер 2.

Как и в предыдущих примерах, команды можно набирать сокращенно, кроме того, вместо fastEthernet можно использовать обозначение fa.

Просмотрим результат конфигурирования, выполнив команду show vlan br еще раз, рисунок 6.6.

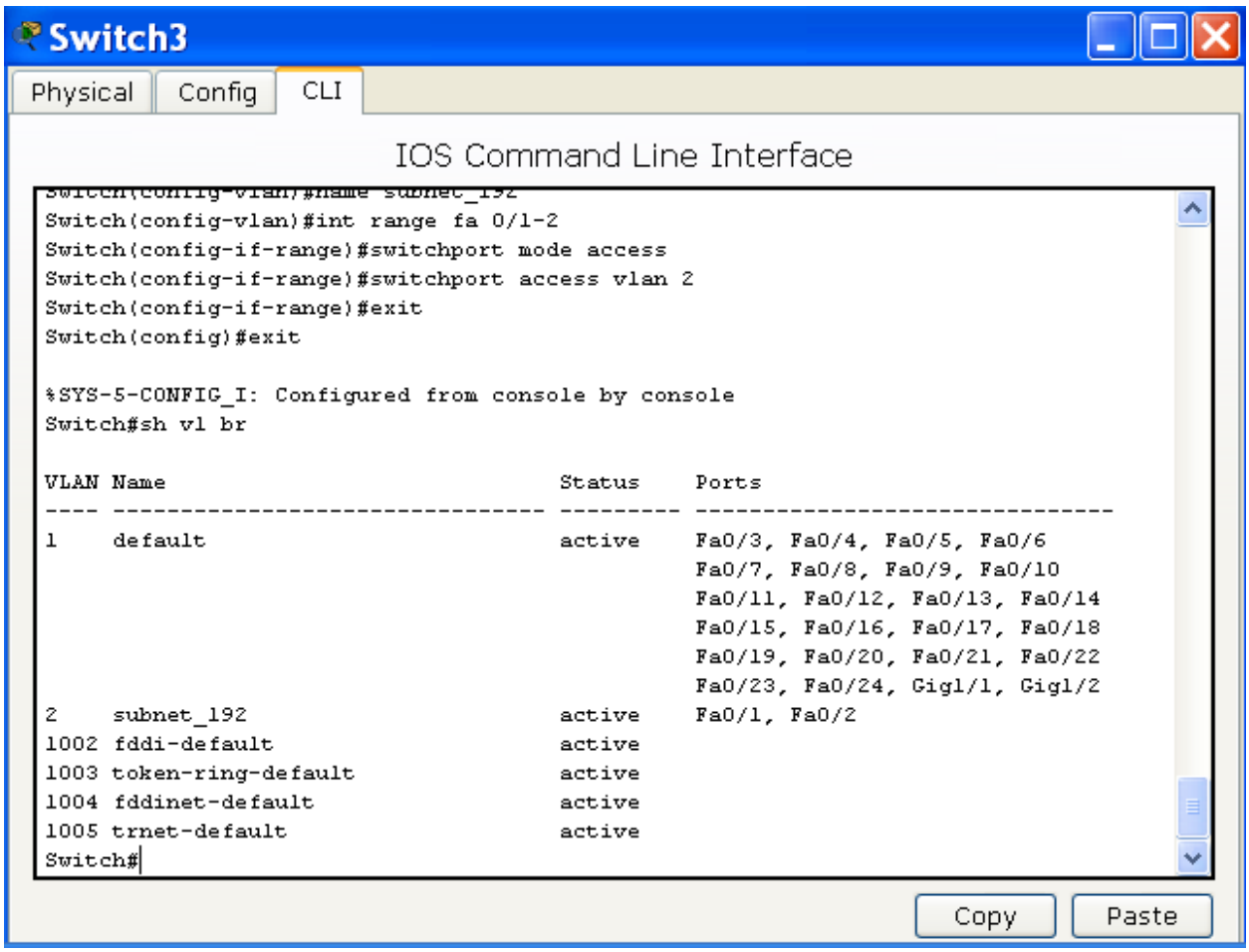

Рисунок 6.6 – Просмотр конфигурации VLAN

Из рисунка видно, что в коммутаторе появилась вторая VLAN с именем subnet_192, к которой относятся порты 0/1 и 0/2.

Далее аналогичным образом создадим vlan 3 с именем subnet_172, и сделаем его портами доступа интерфейсы fastEthernet 0/3 и fastEthernet 0/4.

Соответственно, компьютеры, находящиеся в разных виртуальных сетях, будут недоступны друг другу, что легко проверить с использованием утилиты **ping**.

Наибольшую практическую ценность представляет конфигурирование VLAN на нескольких коммутаторах с использованием магистральных портов. Рассмотрим конфигурирование коммутаторов в этом случае.

Рассмотрим сеть, показанную на рисунке 6.7.

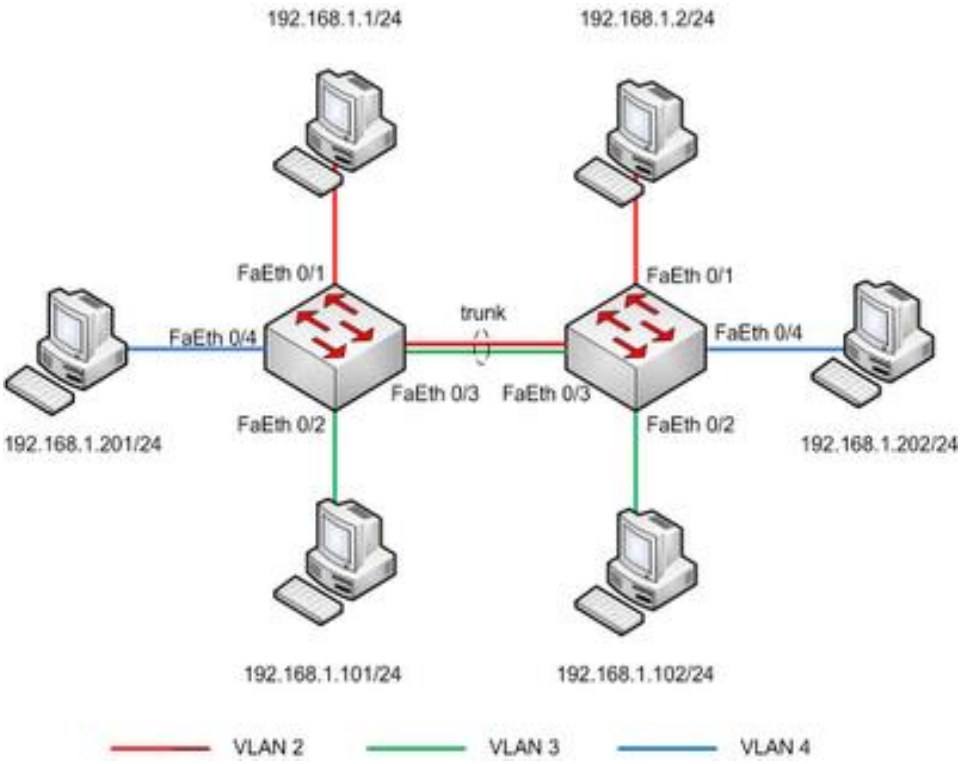

Рисунок 6.7 – Конфигурирование коммутаторов

По аналогии с предыдущим примером произведем конфигурирование обоих коммутаторов:

Switch(config)#vlan 2

Switch(config-vlan)#name subnet_2

Switch(config-vlan)#int fa 0/1

Switch(config-if)#switchport mode access

Switch(config-if)#switchport access vlan 2

Switch(config-if)#exit

Switch(config)#vlan 3

Switch(config-vlan)#name subnet_3

Switch(config-vlan)#int fa 0/2

Switch(config-if)#switchport mode access

Switch(config-if)#switchport access vlan 3

Switch(config-if)#exit

Switch(config)#vlan 4

Switch(config-vlan)#name subnet_4

Switch(config-vlan)#int fa 0/4

Switch(config-if)#switchport mode access

Switch(config-if)#switchport access vlan 4

Switch(config-if)#exit

Как следует из перечня команд, на обоих коммутаторах созданы три VLAN. Теперь необходимо сконфигурировать на третьем порту каждого из коммутаторов магистральный режим:

Switch(config)#int fa 0/3

Switch(config-if)# switchport mode trunk

В результате трафик всех трех созданных ранее VLAN будет проходить через порт 3.

Используя на интерфейсе команду switchport mode trunk, мы перевели его в магистральный режим, в котором интерфейс пропускает через себя трафик всех существующх на коммутаторе VLAN, но иногда необходимо передавать через данный интерфейс трафик не всех VLAN, а лишь некоторых. Для этого на обоих коммутаторов выполним команды:

Switch(config)#interface fastEthernet 0/3

Switch(config-if)#switchport trunk allowed vlan 2-3

Команда "switchport trunk allowed vlan 2-3" указывает магистральному порту коммутатора, трафик каких VLAN ему пропускать через себя. После того, как будет выполнена эта команда, компьютер PC4 должен перестать видеть компьютер PC5. Команда "switchport trunk allowed vlan" при своем использовании каждый раз задает разрешенные порты заново, то есть если выполнить команду switchport trunk allowed vlan 5, а потом выполнить команду switchport trunk allowed vlan 6, то разрешенным окажется только трафик VLAN номер 6. Для добавления VLAN к списку разрешенных служит команда switchport trunk allowed vlan x, где x номер добавляемого VLAN. Для удаления VLAN из списка разрешенных используется команда switchport trunk allowed vlan remove x, где x номер удаляемого VLAN. Для просмотра информации о

настроенных на коммутаторе магистральных портах служит команда show int trunk.

Таким образом, с использованием коммутаторов второго уровня единую сеть можно разделить на виртуальные сети с изолированным друг от друга трафиком. Однако на практике часто возникают задачи гибкого объединения нескольких VLAN между собой. Эту задачу решают маршрутизаторы и коммутаторы третьего уровня, которые будут рассмотрены ниже.

Необходимо отметить, что рассмотренная выше терминология (Access-порт, Trunk-порт) характерна только для устройств Cisco Systems. В ряде случаев порты, способные пропускать через себя трафик нескольких VLAN, называются тэгированными (tagging). Объясняется это тем, что такой порт перед передачей кадра помещает в него дополнительную информацию (тэг), анализируя которую, принимающий порт имеет возможность определить номер VLAN, к которой этот кадр относится. Соответственно, принимающий порт этот тэг удаляет.

Порты, не вносящие изменений в передаваемые кадры (Access-порты в терминологии Cisco), называются нетэгированными (untagging).

6.5 Отчет по работе:

- Сконфигурированная сеть доступа;
- Результаты разделения сети на VLAN;
- Результаты проверки связности сети.

Практическое занятие №7

Конфигурирование консольного доступа к сетевому оборудованию

7.1 Цель работы: Привитие навыков конфигурирования консольного доступа к коммутаторам и маршрутизаторам в защищенном режиме.

7.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Windows пакетом Cisco Packet Tracer или GNS3.

7.3 Задание:

Используя программные продукты (Cisco Packet Tracer или GNS3) или реальное оборудование, подключиться к консольному порту устройства (коммутатора или маршрутизатора). Создать пароль для входа в привилегированный режим, используя параметр **secret**. Создать три учетные записи с разными уровнями привилегий, используя функцию AAA. При создании учетных записей использовать логин **<фамилия в английской транскрипции№учетной записи>**. Проверить работоспособность системы аутентификации и авторизации по локальной базе пользователей.

7.4 Указания к проведению работы.

Как известно, при консольном доступе после загрузки устройства оно переходит в пользовательский режим (рисунок 7.1).

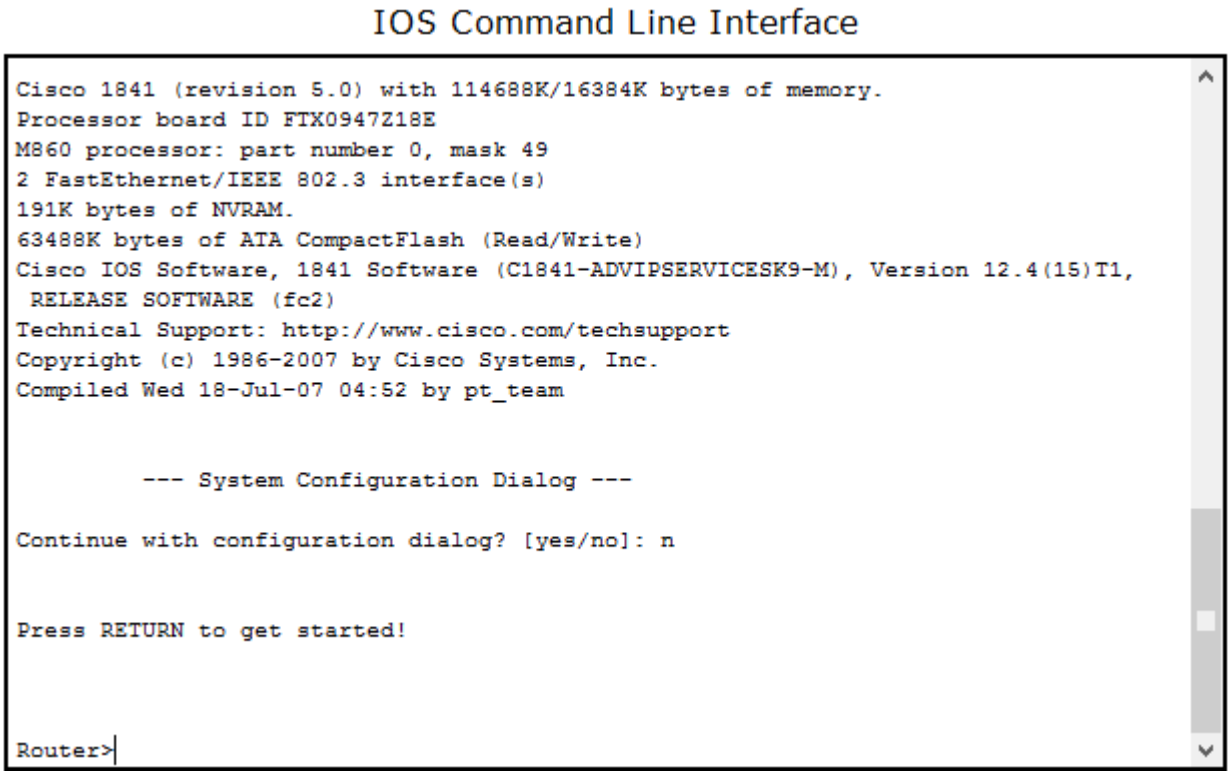

Рисунок 7.1 – Вид приглашения пользовательского режима

Для перехода в привилегированный режим в устройствах Cisco используется команда **enable.** Так как привилегированный режим является потенциально опасным, рекомендуется защитить переход в этот режим паролем. Как известно [1], в устройствах Cisco существует несколько видов паролей. В частности, пароли **enable password** и **enable secret** как раз и обеспечивают авторизацию входа в привилегированный режим.

Данные пароли устанавливаются в режиме конфигурирования с использованием команд:

R1(config)#enable password manin

R1(config)#enable secret manin1

Первая команда устанавливает пароль типа **enable password** (для примера был выбран пароль **manin**), вторая – типа **enable secret** (пароль **manin1**).

Пароль типа **enable password** хранится на устройстве в незашифрованном виде, что делает его слабозащищенным. Например, если злоумышленник подключится по консоли, войдет в пользовательский режим

и использует команду **show running-config**, он без труда сможет увидеть установленный пароль (рисунок 7.2).

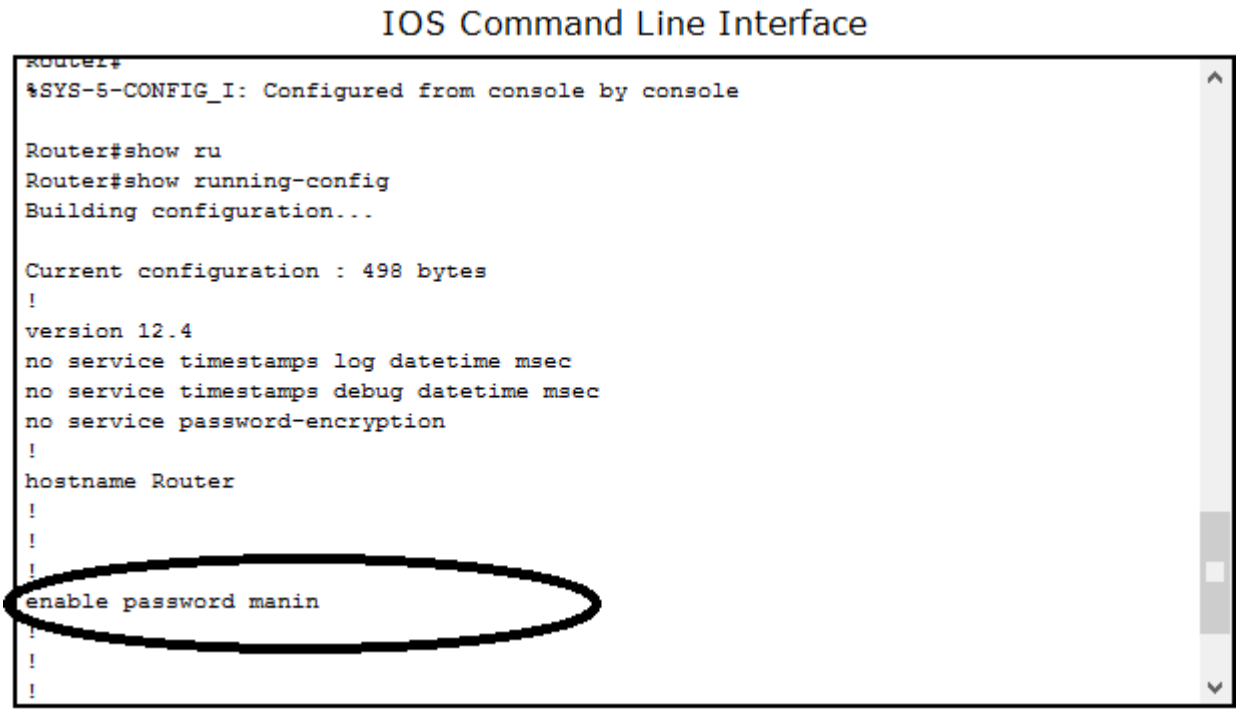

Рисунок 7.2 – Просмотр в конфигурации устройства заданного пароля

Самым очевидным решением этой проблемы является хранение пароля в зашифрованном виде. Для шифрования пароля можно использовать команду **service password-encryption**, выполняемую в режиме глобального конфигурирования. В этом случае при просмотре конфигурации пароль будет представлен в зашифрованном виде (рисунок 7.3).

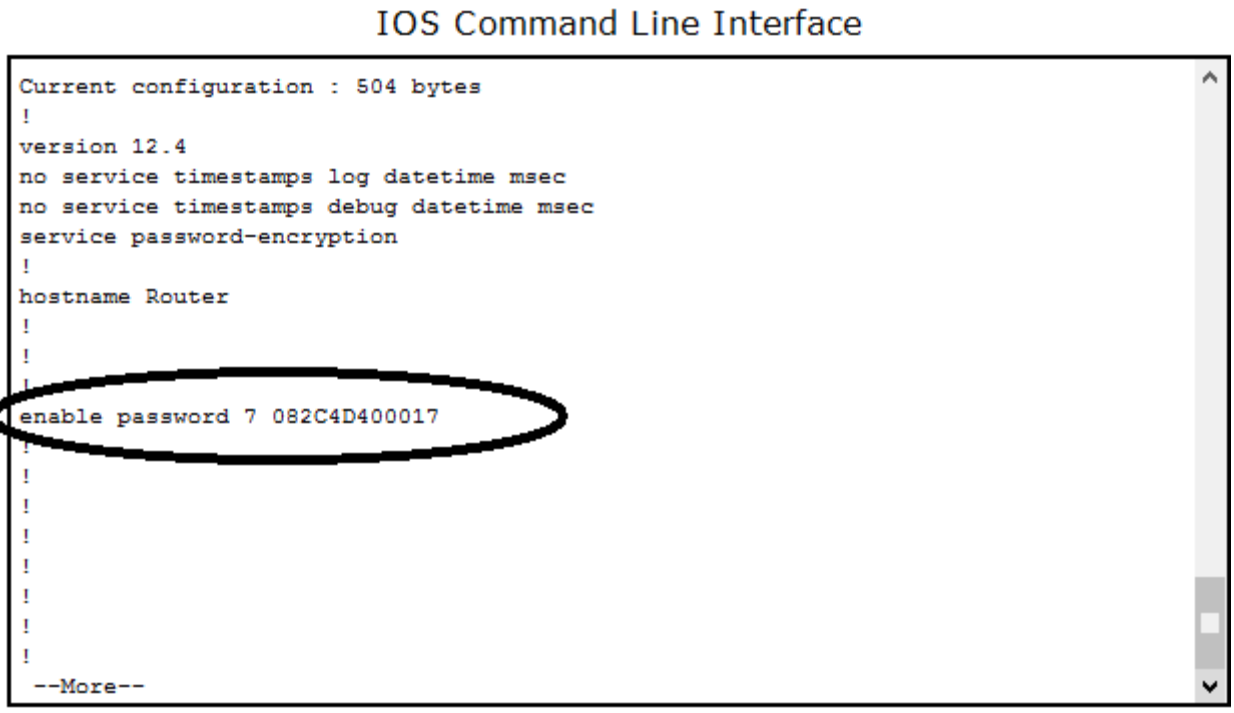

Рисунок 7.3 – Просмотр в конфигурации устройства зашифрованного пароля

Однако на практике не рекомендуется данный способ защиты перехода в привилегированный режим. Дело в том, что шифрование в этом случае производится с использованием алгоритма шифрования, основанного на операции XOR и достаточно легко поддающегося взлому. Цифра 7 перед зашифрованным паролем как раз и указывает на то, что используется этот слабо защищенный алгоритм шифрования.

Пример расшифровки пароля, показанного на рисунке 7.3, с использованием одной из известных программ показан на рисунке 7.4.

Рисунок 7.4 – Расшифровка пароля

Второй тип пароля – **enable secret** – использует более мощный алгоритм шифрования, основанный на использовании хэш-функций (MD5). Создадим пароль **enable secret** и посмотрим его отображение в конфигурации устройства, рисунок 7.5.

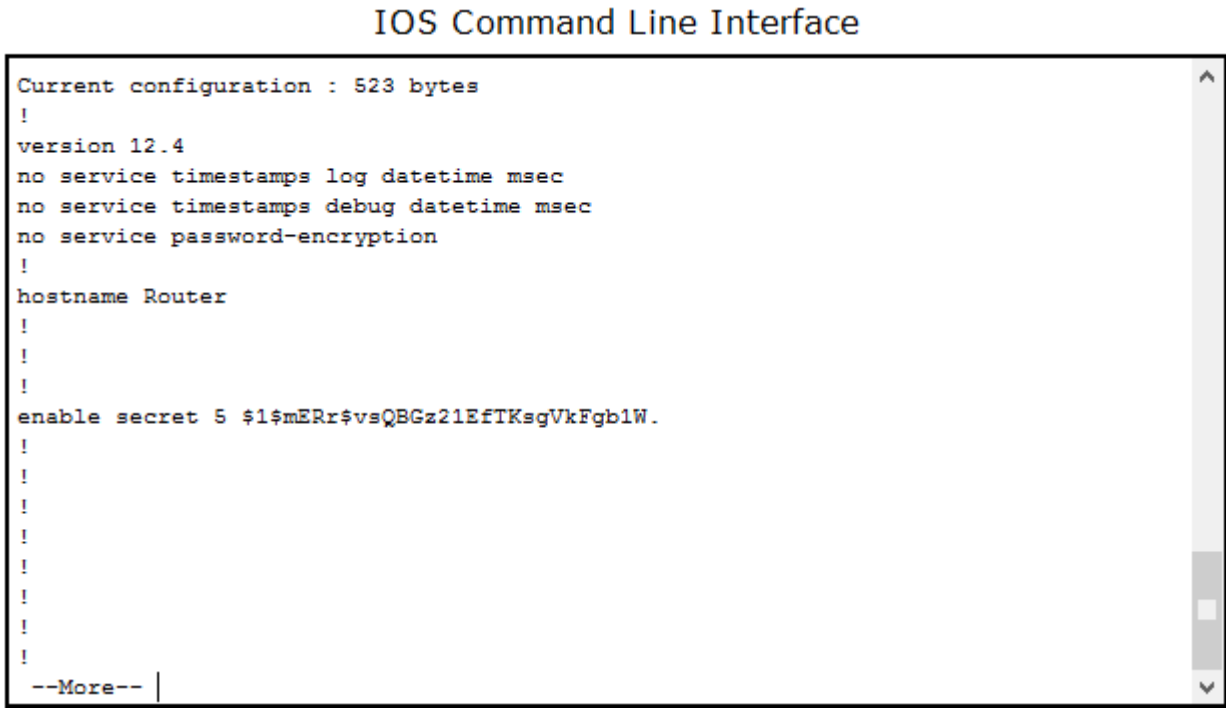

Рисунок 7.5 – Отображение пароля **enable secret**

Как видно из рисунка, теперь перед паролем стоит цифра 5, что и указывает на использование алгоритма MD5, и мы видим результат выполнения хэш-функции, которая, как известно, необратима [2].

Однако следует помнить, что в этом случае пароль можно подобрать. Существуют специальные таблицы, включающие в себя результаты выполнения хэш-функций для наиболее часто встречающихся паролей. В Интернете есть специализированные ресурсы, подбирающие пароли к хэш-функциям. Трудоемкость подбора пароля **enable secret** напрямую зависит от его сложности. Отсюда следует очевидный вывод – пароль должен быть достаточно сложным и нетривиальным.

Однако на практике могут иметь место случаи, когда все-таки необходимо использовать пароль **enable password**. В этом случае

большинство устройств Cisco допускают использование обоих паролей, но пароль **enable secret** будет иметь более высокий приоритет.

Помимо защиты от несанкционированного доступа к привилегированному режиму, при конфигурировании оборудования необходимо обеспечить защиту и пользовательского режима. Как указывалось выше, при входе в пользовательский режим злоумышленник также может выполнить ряд команд и получить некоторые сведения об устройстве. В частности, на устройствах Cisco в пользовательском режиме могут быть выполнены следующие команды (рисунок 7.6).

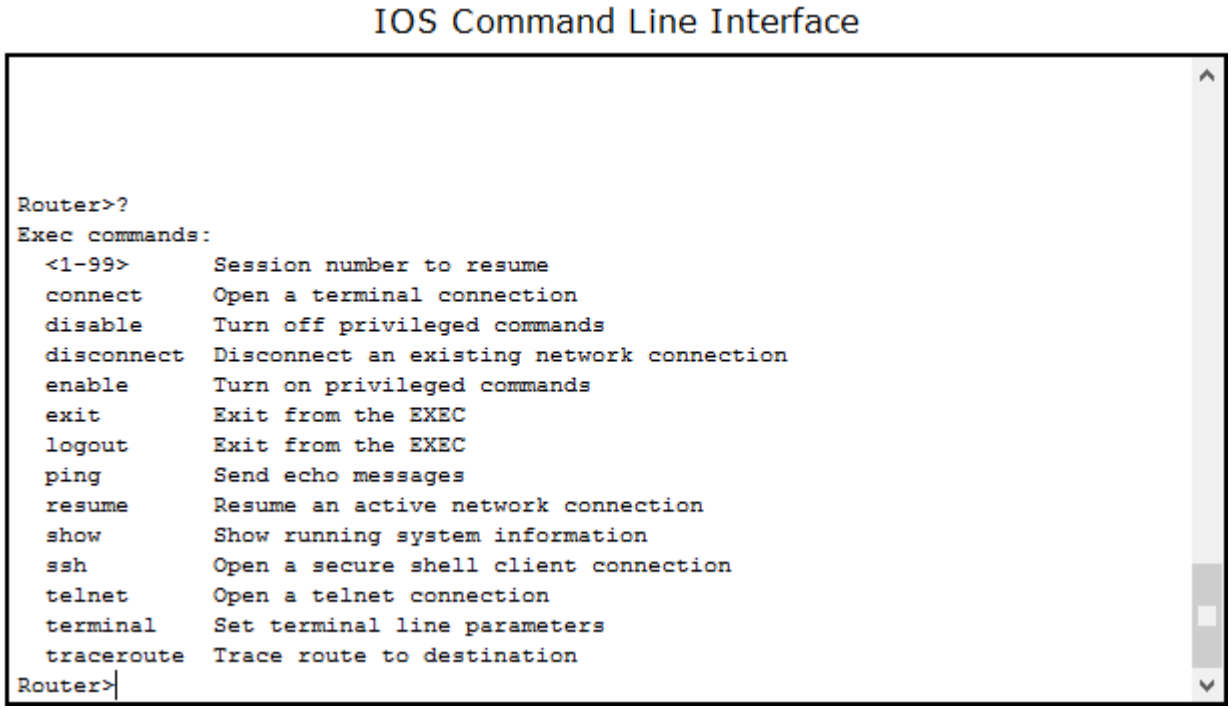

Рисунок 7.6 – Команды, доступные из пользовательского режима

На рисунке 7.6 представлены только первые слова возможных команд. Может быть несколько команд, начинающихся с этого слова. Например, на рисунке 7.7 представлен список команд, начинающихся со слова **show**.

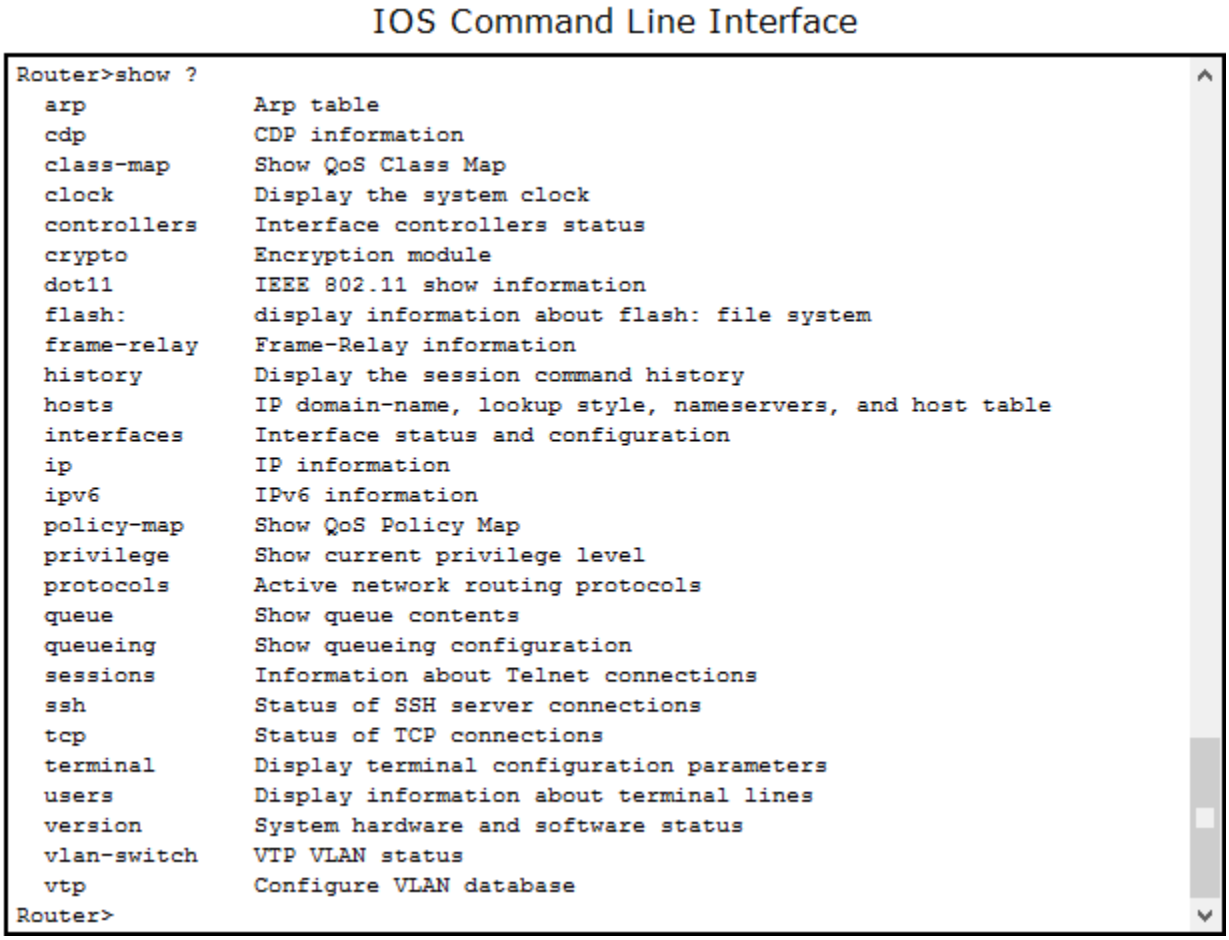

Рисунок 7.7 – Список команд, начинающихся со слова **show**

Более того, на всех устройствах Cisco по умолчанию установлен проприетарный (фирменный) протокол CDP (Cisco Discovery Protocol), позволяющий обнаруживать соседние устройства того же производителя и получать о них некоторую информацию (версию IOS, адреса, типы устройств, и т.д.). Таким образом, злоумышленник, находящийся в пользовательском (не привилегированном!) режиме получает возможность узнать некоторые сведения не только об устройстве, к которому он подключен, но и о его соседях. Для этого используется команда **show cdp neighbors detail**.

Для того, чтобы защитить не только привилегированный, но и пользовательский режим, используется аутентификация по учетным записям пользователей.

Как известно, при использовании учетной записи аутентификация проводится, как минимум, по двум параметрам – логину и паролю. Очевидно,

что логины и пароли легальных пользователей должны где-то храниться. Для хранения учетных записей используется два подхода:

1. Хранение в локальной базе непосредственно на устройстве.

2. Хранение на специализированном сервере (AAA-сервере).

Наиболее надежным и удобным в эксплуатации способом является использование AAA-сервера (серверов). Однако в небольшой сети использование AAA-сервера обычно представляется нецелесообразным, поэтому рассмотрим сначала первый способ. Об использовании AAA-сервера речь пойдет на следующем занятии.

При создании базы пользователей необходимо знать, что устройства Cisco позволяют достаточно гибко управлять теми правами (привилегиями), которые предоставляются каждому конкретному пользователю. Имеется 16 уровней привилегий (от 0 до 15), при этом по умолчанию в устройстве Cisco настроены три уровня:

1. Уровень 0. Пользователь с этим уровнем может выполнять минимальный набор команд. На практике используется крайне редко.

2. Уровень 1. Соответствует пользовательскому режиму, то есть пользователь с этим уровнем может выполнять все команды, доступные в пользовательском режиме.

3. Уровень 15. Соответствует привилегированному режиму, то есть пользователь с этим уровнем может выполнять все команды, доступные в привилегированном режиме.

Уровни 2 – 14 можно настраивать, то есть, если какому-либо пользователю присваивается уровень из этого диапазона, можно в явном виде указать, какие команды привилегированного режима будут ему доступны. Если учесть, что число работников, обслуживающих сеть, обычно невелико, такой подход позволяет каждому из работников предоставить только те права, которые необходимы ему для работы.

Создание учетной записи производится в режиме глобального конфигурирования с использованием команды

Router1(config)#username <логин> privilege <уровень> password/secret <пароль>

В команде используется параметр либо **password**, либо **secret**, разница между ними была рассмотрена выше. Соответственно, более предпочтительным является использование параметра **secret**.

В случае, если создается учетная запись с привилегиями уровней 2 – 14, доступные на этом уровне команды указываются в явном виде:

Router1(config)#privilege exec level <уровень> <команда>

Рассмотрим процесс создания локальной базы на конкретном примере. Предположим, что в организации имеется три администратора (назовем их admin1, admin2 и admin3). Admin1 является главным, ему доступны все команды (уровень 15). Admin2 имеет доступ к командам **show running-config** и **ping**, admin3 – к командам **show ip route**, **ping** и **traceroute**.

Создание локальной базы пользователей иллюстрируется рисунком 7.8.

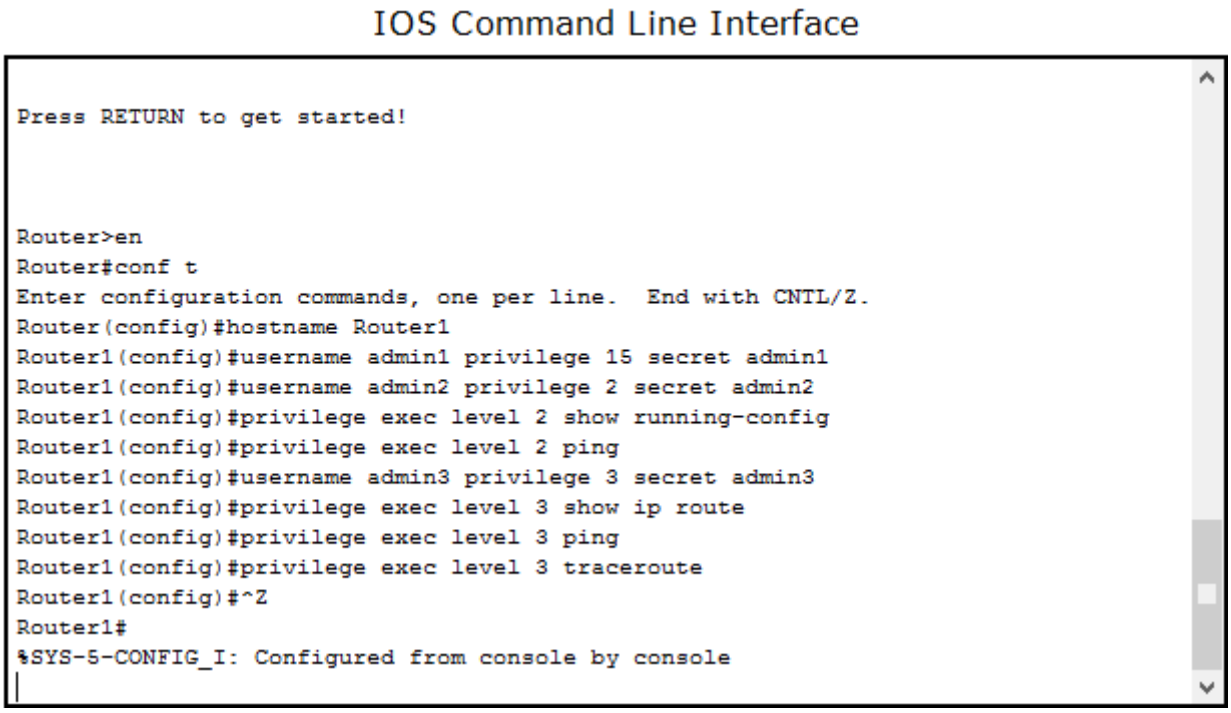

Рисунок 7.8 – Создание локальной базы пользователей на маршрутизаторе

После этого необходимо войти в режим конфигурирования консольного порта и указать, что вход необходимо производить с использованием локальной базы, рисунок 7.9.

```
Router1>en
Router1#conf
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line.  End with CNTL/Z.
Router1(config)#line con
Router1(config)#line console 0
Router1(config-line)#login local
Router1(config-line)#
```

Рисунок 7.9 – Конфигурирование консольного порта

Зайдем на маршрутизатор через консольный порт как admin2 и попробуем выполнить сначала запрещенную, а затем разрешенную для этого пользователя команду, рисунок 7.10.

```
Username:
Username: admin2
Password:

Router1#show ip route
             ^
% Invalid input detected at '^' marker.

Router1#show running-config
Building configuration...

Current configuration : 970 bytes
!
version 12.4
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Router1
!
!
!
!
!
!
!
!
username admin1 privilege 15 secret 5 $1$mERr$7n6je7c9FKvO.o.40Rj1Q0
username admin2 privilege 2 secret 5 $1$mERr$4CFVt/60iQmc.ia/CrCAa/
username admin3 privilege 3 secret 5 $1$mERr$JpU75fgWPP7c43kksZEKc1
!
 --More--
```

Рисунок 7.10 – Пример выполнения команд

Из рисунка видно, что команда **show ip route** не была выполнена (она запрещена для этого пользователя), а команда **show running-config** вывела информацию о состоянии устройства.

На практике представленный выше способ используется редко. Все современные устройства Cisco поддерживают функцию AAA (Authentication, Authorization and Accounting). Для включения данной функции используется команда

Router1(config)#aaa new-model

Аутентификация настраивается командой

Router1(config)#aaa authentication login <method-list> local

Параметр **local** указывает на необходимость использования локальной базы, параметр **method-list** указывает на используемый метод аутентификации. Например, при использовании метода **default** аутентификация применяется не только к консольному, но и к удаленному доступу, о котором пойдет речь ниже.

В заключение особенностей конфигурирования консольного доступа следует отметить следующее. Практически все современное оборудование имеет функцию сброса к заводским установкам. При этом могут быть сброшены и установленные пароли. В устройствах Cisco сброс паролей может быть запрещен соответствующей командой (**no service password-recovery)**, однако функция сброса пароля может оказаться полезной. Поэтому наряду с техническими мероприятиями по защите доступа к сетевым устройствам обязательно должны реализовываться и организационные – ограничение физического доступа к устройству, использование выделенных серверных комнат, вандалоустойчивых шкафов, и т.д.

7.5 Отчет по работе:

- Результаты проверки корректной доступа к сетевым устройствам (коммутаторам и маршрутизаторам).

Практическое занятие №8

Конфигурирование удаленного доступа к сетевому оборудованию

8.1 Цель работы: Привитие навыков конфигурирования удаленного доступа к коммутаторам и маршрутизаторам в защищенном режиме.

8.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Windows пакетом Cisco Packet Tracer или GNS3.

8.3 Задание:

- Используя консольный порт, создать в локальной базе маршрутизатора одну учетную запись с первым уровнем привилегий. Создать пароль входа в привилегированный режим, используя параметр **secret**. Проверить работоспособность системы аутентификации;
- Подключить к одному из Etrhernet-портов маршрутизатора рабочую станцию, произведя необходимые настройки (IP-адреса порта маршрутизатора и рабочей станции). Осуществить удаленное подключение к маршрутизатору с использованием протокола Telnet. Проверить работоспособность системы аутентификации;
- Сконфигурировать маршрутизатор для работы с протоколом SSH, исключив возможность использования протокола Telnet. Используя SSH-клиент, осуществить удаленное подключение к маршрутизатору с использованием протокола SSH. Проверить работоспособность системы аутентификации.

ПРИМЕЧАНИЕ. Если для выполнения задания используется Cisco Packet Tracer, для подключения рабочей станции по протоколу SSH необходимо использовать командную строку и команду:

PC> ssh –l <логин> <IP-адрес>

При использовании эмулятора GNS3 или реального оборудования необходимо использовать любой SSH-клиент, например, PuTTY.

8.4 Указания к проведению работы.

Независимо от используемого протокола удаленного доступа сам доступ организуется с использованием виртуального интерфейса **vty**. Таким образом, необходимо сначала рассмотреть способы конфигурирования виртуальных интерфейсов.

Первый способ аналогичен настройке консольного порта (см. рисунок 1.9) с тем исключением, что обращаться надо не к консольному порту (**line 0**), а к виртуальному (**vty**). Так как к устройству могут удаленно подключаться несколько пользователей, организуется несколько виртуальных интерфейсов (то есть несколько одновременных удаленных подключений). В более старых версиях IOS число виртуальных интерфейсов 5 (от 0 до 4), в более новых – 16 (от 0 до 15), однако для конфигурирования никакой разницы нет.

В этом случае для включения аутентификации по локальной базе пользователей всех пяти виртуальных интерфейсов используются команды:

Router1(config)#line vty 0 4

Router1(config-line)#login local

Второй способ заключается в использовании AAA. Если при конфигурировании консольного порта уже была включена функция AAA с параметром **default**, то аутентификация будет работать как для консольного, так и для виртуальных интерфейсов.

Дальнейшее конфигурирование удаленного доступа зависит от того, какой из протоколов предполагается использовать. Рассмотрим конфигурирование удаленного доступа на базе протоколов telnet и ssh.

Как известно [3], протокол Telnet является одним из самых старых протоколов удаленного доступа, поэтому сегодня трудно найти оборудование, не поддерживающее этот протокол. Конфигурирование удаленного доступа по протоколу Telnet не является сложным, что является его несомненным достоинством. Однако данный протокол обладает одним существенным

недостатком, существенным с точки зрения безопасности сетей – все данные передаются в открытом виде, что дает возможность злоумышленнику получить доступ как к логинам и паролям, передаваемым в процессе аутентификации, так и непосредственно к передаваемым данным.

Рассмотрим фрагмент сети, показанный на рисунке 8.11. Для моделирования процесса конфигурирования удаленного доступа будем использовать сетевой эмулятор GNS3 [4], а не Cisco Packet Tracer, ввиду его большей функциональности.

Рисунок 8.11 – Фрагмент сети

Из рисунка 8.11 видно, что рассматриваемый фрагмент сети содержит маршрутизатор (в нашем случае это Cisco 3745), к порту f 0/0 подключена рабочая станция под управлением ОС Windows XP.

Для обеспечения удаленного доступа к маршрутизаторам с использованием этих рабочих станций необходимо на каждом из них создать локальную базу пользователей, и настроить вход через виртуальный интерфейс с аутентификацией по локальной базе. Приведем необходимые для этого команды для маршрутизатора R1 (команды конфигурирования интерфейсов маршрутизатора здесь приводить не будем, считаем, что интерфейсы были сконфигурированы в консольном режиме):

R1(config)#enable secret floor1 – задаем пароль для входа в привилегированный режим;

R1(config)#username manin1 privilege 1 secret 1234 – создаем пользователя manin1 с уровнем привилегий 1 и паролем 1234;

R1(config)#line vty 0 4 – входим в режим конфигурирования виртуальных интерфейсов;

R1(config-line)#login local – задаем режим аутентификации по локальной базе.

Конфигурирование маршрутизатора с использованием консольного порта показано на рисунке 8.12. Как видно из рисунка, на интерфейсе fa 0/0 маршрутизатора также был сконфигурирован протокол DHCP, результаты получения рабочей станцией Windows сетевых настроек иллюстрируются рисунком 8.13.

Рисунок 8.12 – Конфигурирование маршрутизатора

Таким образом, на маршрутизаторе была создана локальная база, включающая одного пользователя (manin1) с уровнем привилегий 1. Рекомендуется использовать именно первый уровень привилегий при входе, так как в этом случае пользователь попадает в непривилегированный режим, и для дальнейшей работы ему необходимо будет ввести еще один пароль (авторизация). Это существенно повышает безопасность доступа к устройству.

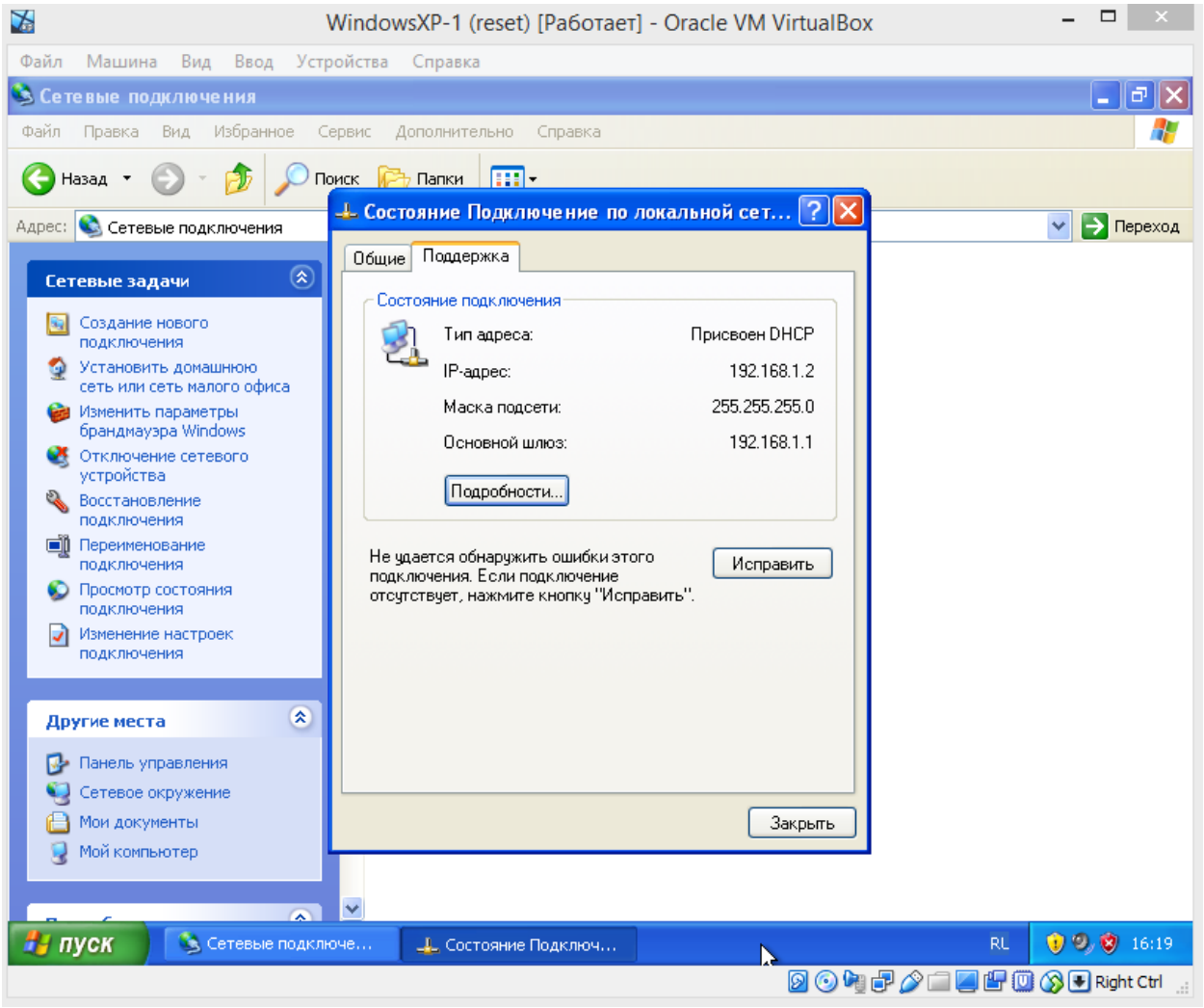

Рисунок 8.13 – Получение рабочей станцией сетевых настроек по протоколу DHCP

После запуска на рабочей станции протокола Telnet было предложено ввести логин и пароль (рисунок 1.14). Так как для пользователя manin1 был определен первый уровень привилегий, данный пользователь после аутентификации был переведен в непривилегированный режим. После набора команды **enable** (переход в привилегированный режим) было предложено ввести дополнительный пароль (рисунок 8.14).

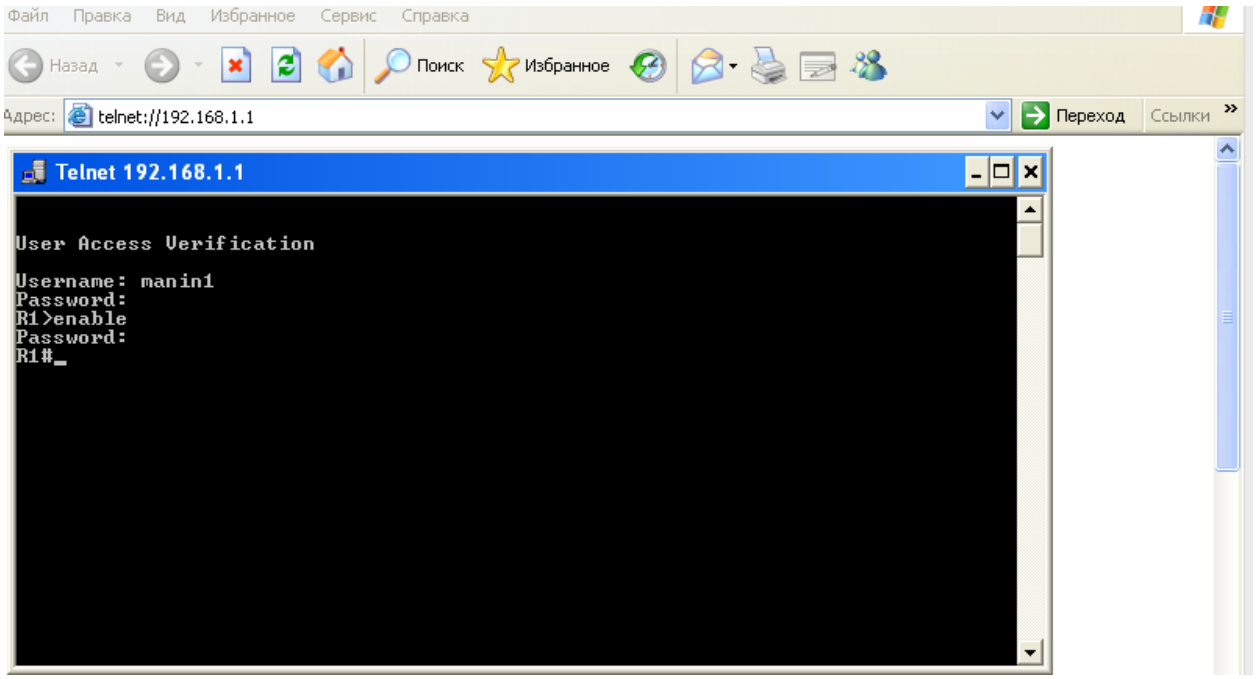

Рисунок 8.14 – Аутентификация пользователя при удаленном доступе

Как указывалось выше, протокол Telnet передает данные в открытом виде, следовательно, их можно перехватить. Для перехвата трафика будем использовать программу WireShark [5]. Результат анализа пакетов, проходящих между рабочей станцией и маршрутизатором при удаленном доступе с использованием данной программы иллюстрируется рисунком 1.15.

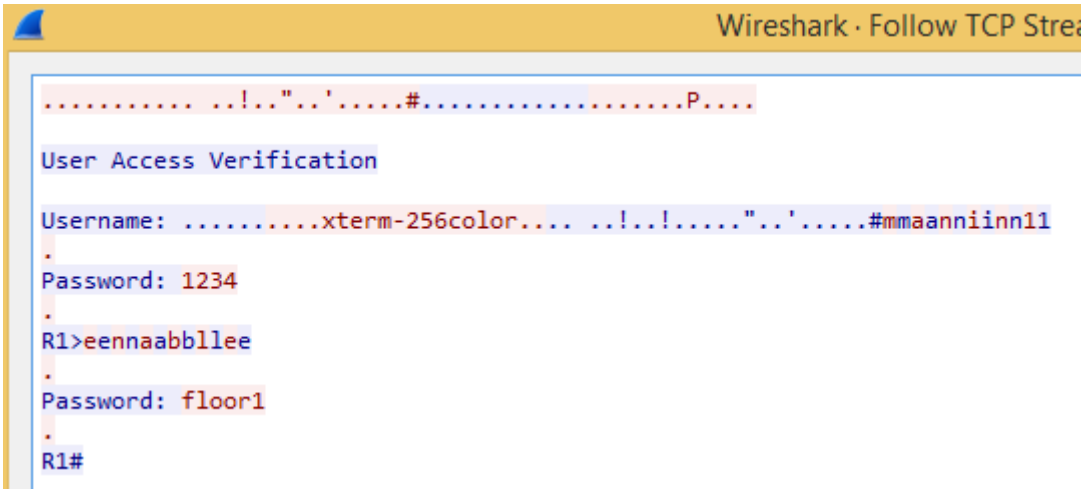

Рисунок 8.15 – Результат анализа передаваемых данных

Из рисунка 8.15 видно, что был использован вход пользователя manin1 с паролем 1234, для входа в привилегированный режим был использован пароль floor1, вход был выполнен успешно.

Как известно, протокол SSH передает все данные в зашифрованном виде, что существенно повышает безопасность сети. Предпочтительным является использование последней версии протокола SSHv.2 как наиболее безопасной.

Однако в отличие от Telnet, при использовании SSH маршрутизатор нуждается в дополнительных настройках, в частности, на нем должен быть развернут SSH-сервер. Рассмотрим конфигурирование маршрутизатора, рисунок 8.16.

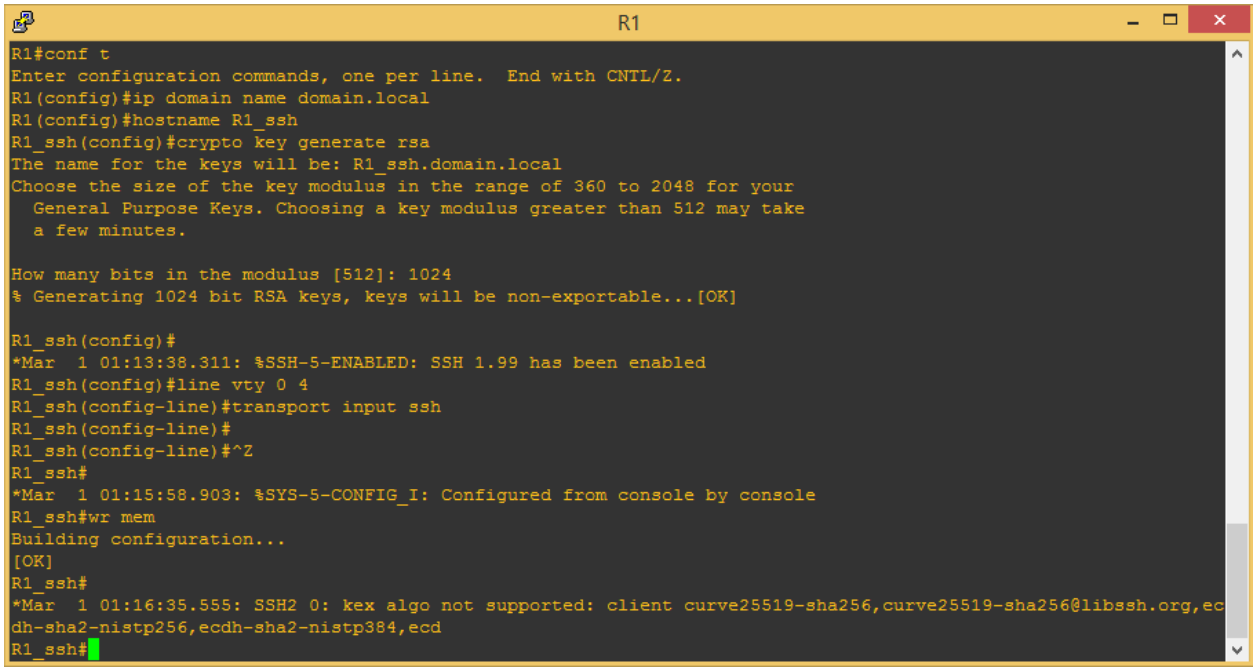

Рисунок 8.16 – Конфигурирование SSH на маршрутизаторе

Команда **ip domain <имя>** используется для указания имени домена, к которому относится маршрутизатор. Это имя используется для генерации ключей шифрования (совместно с именем маршрутизатора, поэтому при конфигурировании стандартное имя R1 было заменено на R1_ssh). Команда **key generate rsa** генерирует RSA-ключ для шифрования передаваемых данных, после чего IOS просит ввести длину используемого ключа (мы

использовали 1024). Команда **transport input ssh** указывает маршрутизатору, что для удаленного входа должен использоваться только протокол SSH.

Удаленный вход на маршрутизатор с использованием программы Putty иллюстрируется рисунком 8.17, результат анализа передаваемых данных с использованием программы WireShark – рисунком 8.18.

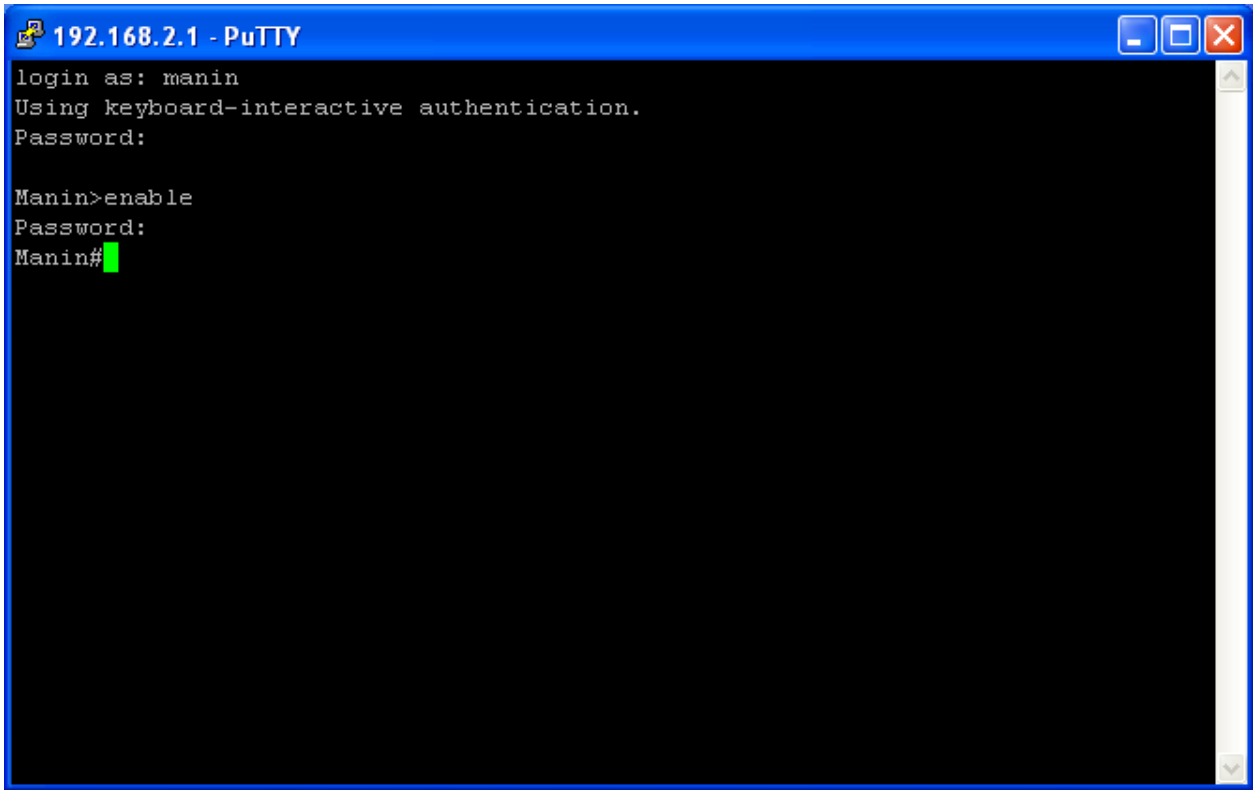

Рисунок 8.17 – Удаленный доступ по протоколу SSH

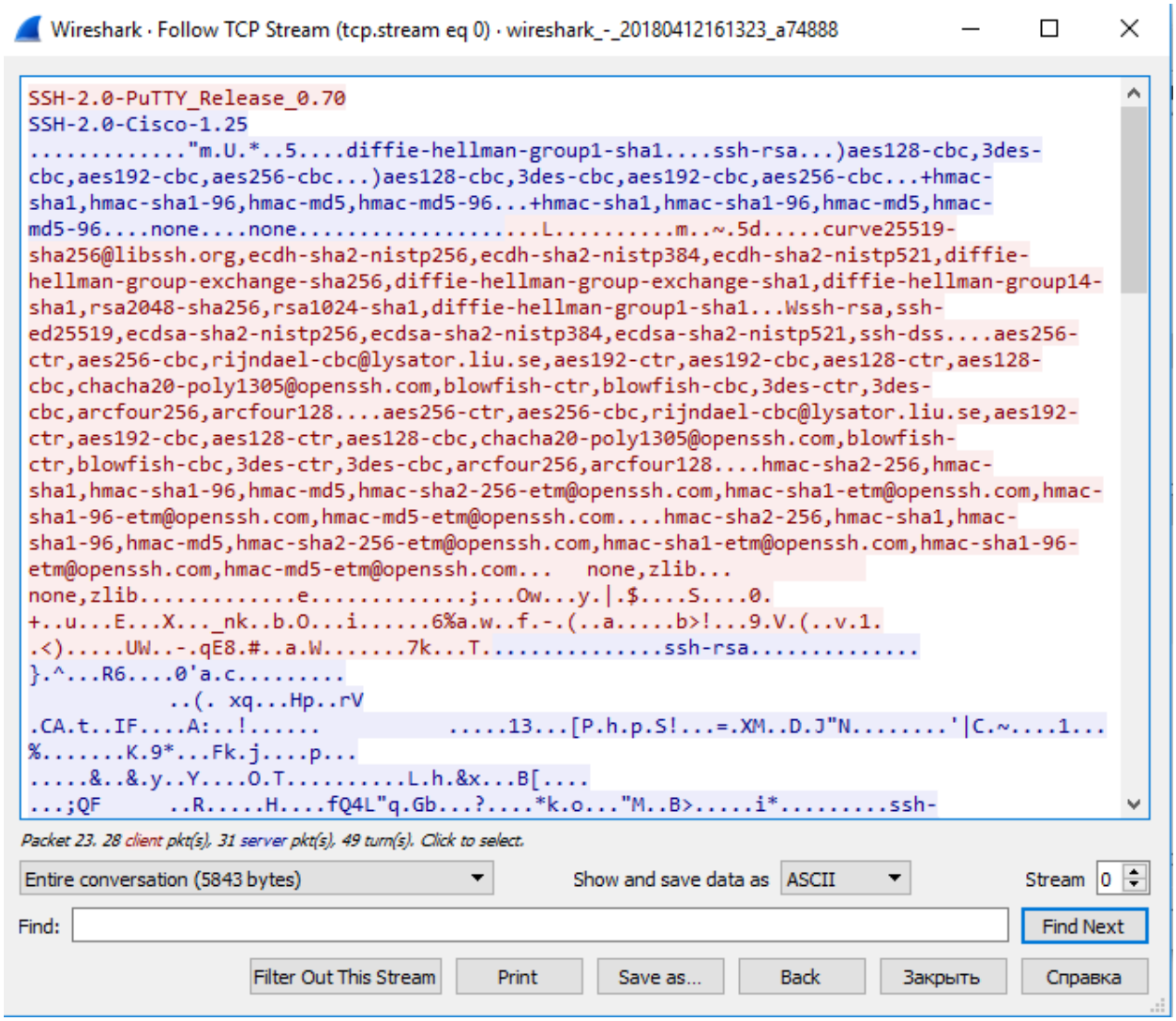

Рисунок 8.18 – Результат анализа передаваемых данных

Из рисунка 1.18 видно, что при использовании протокола SSH перехватить логин и пароли, используемые для удаленного доступа, становится проблематично.

В заключение первой главы необходимо отметить следующее. Все современные сетевые устройства позволяют настроить аутентификацию удаленного доступа с использованием более нового и универсального метода, который получил название **aaa new-model**. Основное отличие его состоит в том, что при настройке создается так называемый метод аутентификации (method-list), который определяет, как будет происходить аутентификация при различных способах доступа. Если в качестве method-list используется **default**,

назначенный способ аутентификации будет работать при любых способах доступа к устройству.

В рассмотренном выше примере можно было использовать команды:

R1(config)#aaa new-model

R1(config)#aaa authentication login default local

В этом случае появляется возможность для разных способов доступа (consol, vty, aux) задавать свои методы аутентификации. В нашем примере при использовании любого способа доступа будет применена аутентификация по локальной базе пользователей (параметр **local**).

Кроме того, на практике не стоит пренебрегать дополнительными возможностями ограничения доступа, которые предоставляет IOS. Например, можно ограничить тайм-аут сессии, число и интервал времени, в течение которого можно повторно вводить пароль в случае ошибки, и т.д. Приведем здесь наиболее часто употребляемые команды:

R1(config-line)#exec-timeout 5 0 – величина тайм-аута сессии (5 мин. 0 сек.);

R1(config)#ip ssh time-out 15 – ограничение времени ввода логина и пароля;

R1(config-line)#transport input ssh – разрешение удаленного входа только по протоколу SSH (Telnet работать не будет);

R1(config)#login delay 5 – задание интервала между повторными вводами пароля;

R1(config)#login block-for 60 attempts 3 within 30 – блокировка входа на 60 секунд, если в течение 30 секунд было 3 неудачных попытки входа.

8.5 Отчет по работе:

- Результаты проверки корректной доступа к сетевым устройствам (коммутаторам и маршрутизаторам).

Практическое занятие №9

Конфигурирование сети с AAA-сервером

9.1 Цель работы: Привитие навыков конфигурирования защищенного доступа к сетевому оборудованию с использованием AAA-сервера.

9.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Windows пакетом Cisco Packet Tracer или GNS3.

9.3 Задание:

- Используя Cisco Packet Tracer или GNS3, собрать схему фрагмента сети, показанную на рисунке 9.1. Настроить IP-адресацию, обеспечив доступность сервера с управляющего ПК.;
- Настроить аутентификацию и авторизацию двух пользователей с различными правами, используя протокол TACACS+.

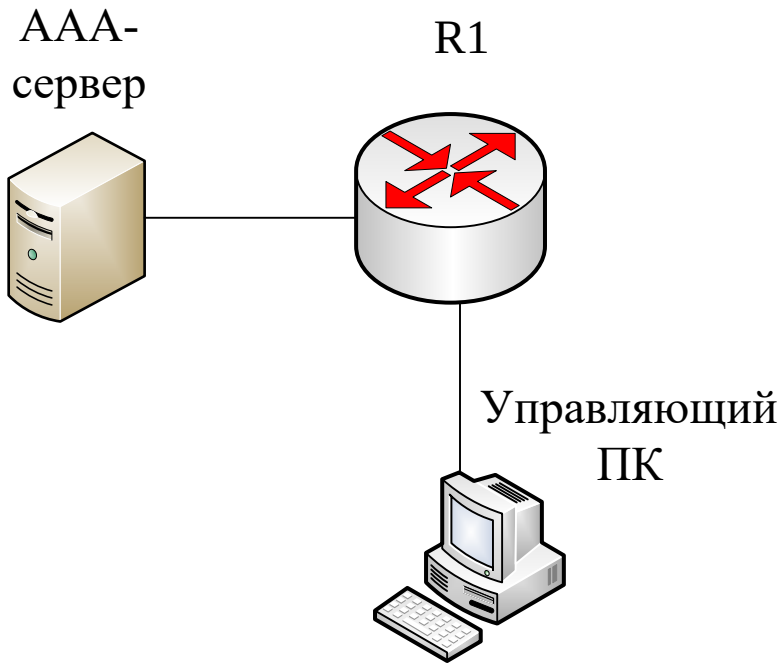

Рисунок 9.1 – Схема фрагмента сети

9.4 Указания к проведению работы.

Как указывалось в предыдущей главе, помимо использования локальной базы пользователей, для аутентификации и авторизации пользователей широкое применение находят AAA-серверы.

Использование AAA-сервера является более удобным и безопасным способом. Во-первых, для добавления, исключения или изменения прав пользователей в этом случае нет необходимости обращаться к каждому из множества сетевых устройств. Во-вторых, база, хранящаяся на сервере, лучше защищена. В-третьих, при периодической смене паролей обращаться приходится только к серверу, а не ко всему множеству сетевых устройств.

Общий алгоритм аутентификации и авторизации пользователей состоит в следующем.

1. Пользователь пытается подключиться к сетевому устройству (например, маршрутизатору) с использованием одного из рассмотренных в предыдущей главе способов (например, с использованием протокола SSH), вводя свои учетные данные (логин, пароль).

2. Маршрутизатор обращается к AAA-серверу и передает ему учетные данные пользователя.

3. AAA-сервер отыскивает в своей базе соответствующую учетную запись (Authentication) и определяет уровень привилегий (Authorization).

4. AAA-сервер пересылает маршрутизатору соответствующую информацию (успешность аутентификации, уровень привилегий).

5. AAA-сервер фиксирует данное событие (Accounting)

6. Маршрутизатор открывает доступ пользователя к оборудованию с учетом уровня привилегий.

Очевидно, что в этом случае информационный обмен, содержащий конфиденциальные данные (логин, пароли) происходит не только между компьютером пользователя и маршрутизатором, но и между маршрутизатором и AAA-сервером. Очевидно, что для этого необходимо использовать протоколы, надежно защищающие передаваемые данные от

перехвата. К таким протоколам относятся RADIUS, DIAMETER, TACACS+, и ряд других.

Протокол RADIUS (Remote Authentication in Dial-In User Service) передает данные в составе UDP-сегментов (порты 1812 и 1813) и работает по сценарию клиент-сервер. Упрощенная диаграмма информационного обмена показана на рисунке 9.2.

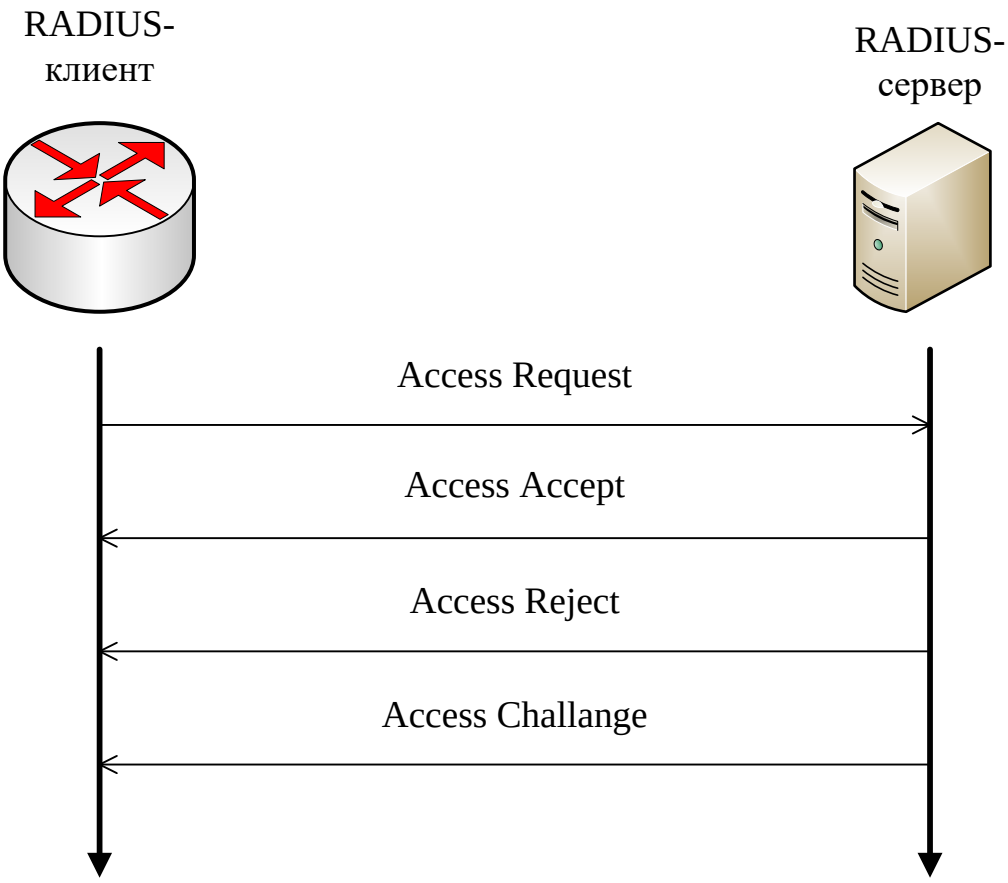

Рисунок 9.2 – Информационный обмен по протоколу RADIUS

Как видно из рисунка, на запрос клиента (Request) сервер может выдать разрешение на доступ (Accept), запрет доступа (Reject), либо запросить дополнительную информацию (Challenge).

Особенностью протокола является то, что при передаче шифруются не все данные, а только пароль. Кроме того, RADIUS является открытым протоколом, поэтому его поддерживает практически все современное оборудование.

Протокол TACACS+ (Terminal Access Controller Access Control System) является разработкой Cisco Systems, обеспечивает передачу данных в составе TCP-сегментов (порт 49) и шифрует все передаваемые данные.

Рассмотрим процесс конфигурирования AAA-сервера, используя бесплатный проект [6]. Для этого в GNS3 соберем схему, показанную на рисунке 9.3.

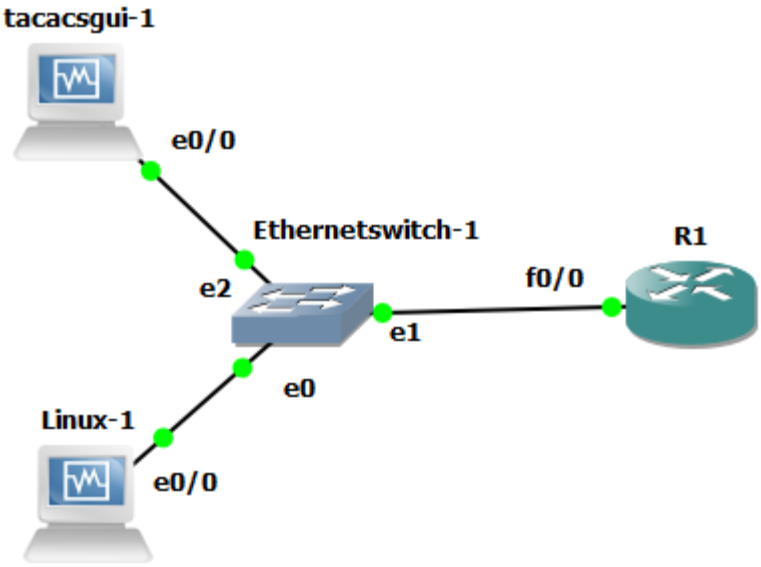

Рисунок 9.3 – Схема сети

Как видно из рисунка 9.3, в схеме имеется AAA-сервер (tacacsgui-1) и рабочая станция (Linux-1), с которой будет осуществляться удаленный доступ к маршрутизатору R1.

Общий вид графического интерфейса tacacsgui показан на рисунке 9.4. Для доступа к нему использовался Интернет-браузер Mozilla Firefox и сокет 10.6.20.10:8008.

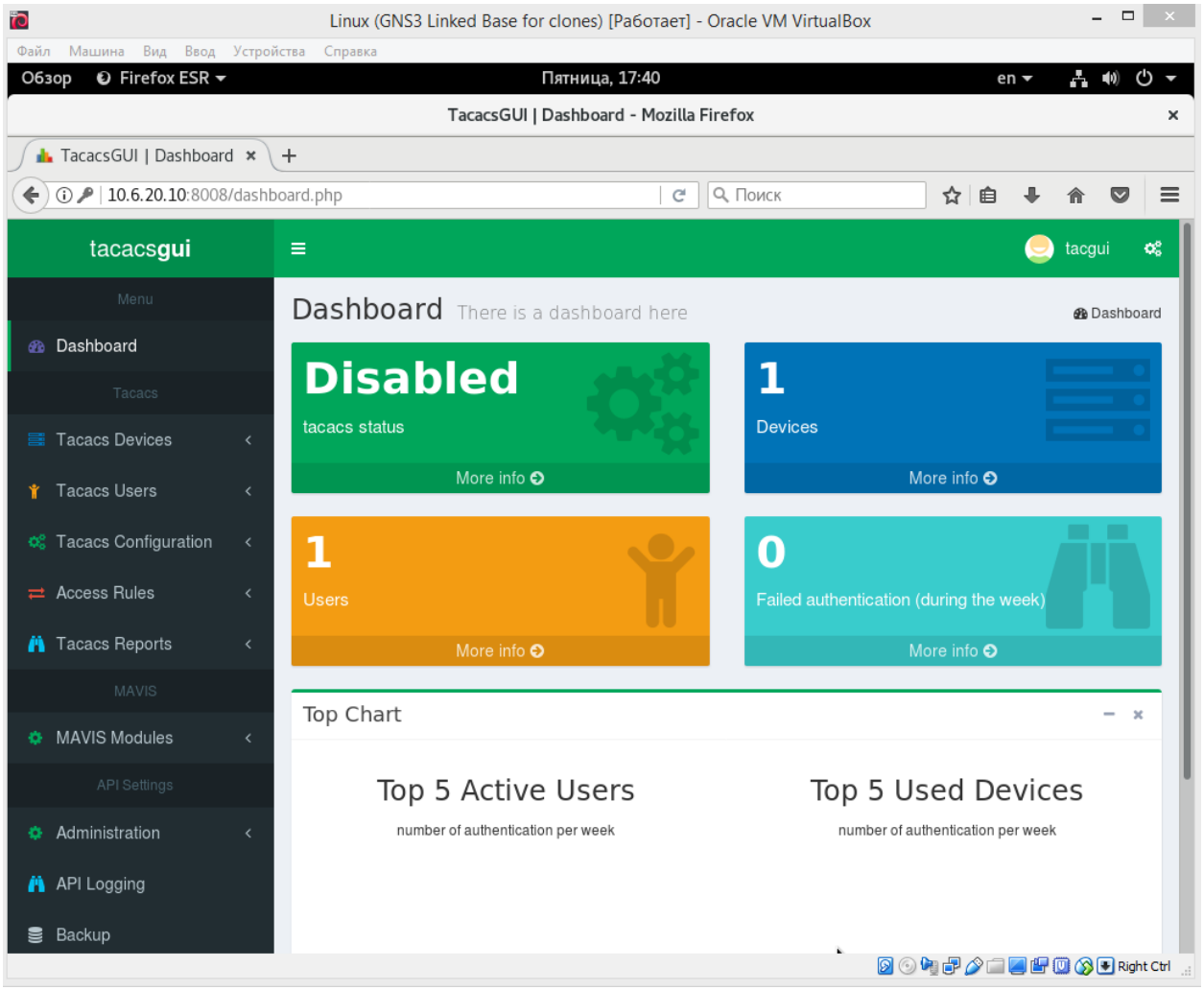

Рисунок 9.4 – Графический интерфейс AAA-сервера

Рисунок 9.5 иллюстрирует процесс добавления устройства, к которому необходимо обеспечить доступ (маршрутизатор R1), а рисунок 2.5 – процесс добавления пользователя (пользователь manin1, пароль 1234, пароль доступа в привилегированный режим – cisco).

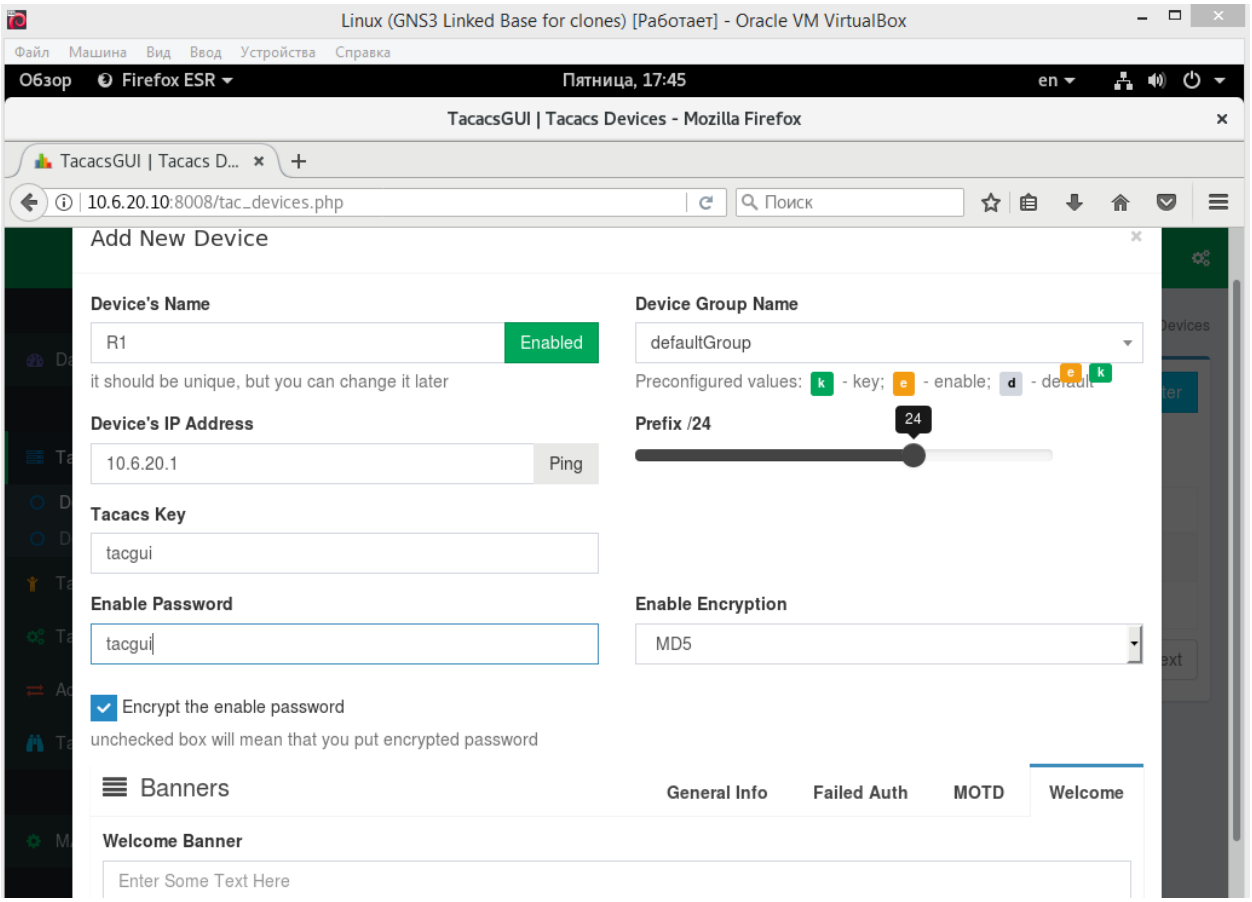

Рисунок 9.5 – Добавление устройства

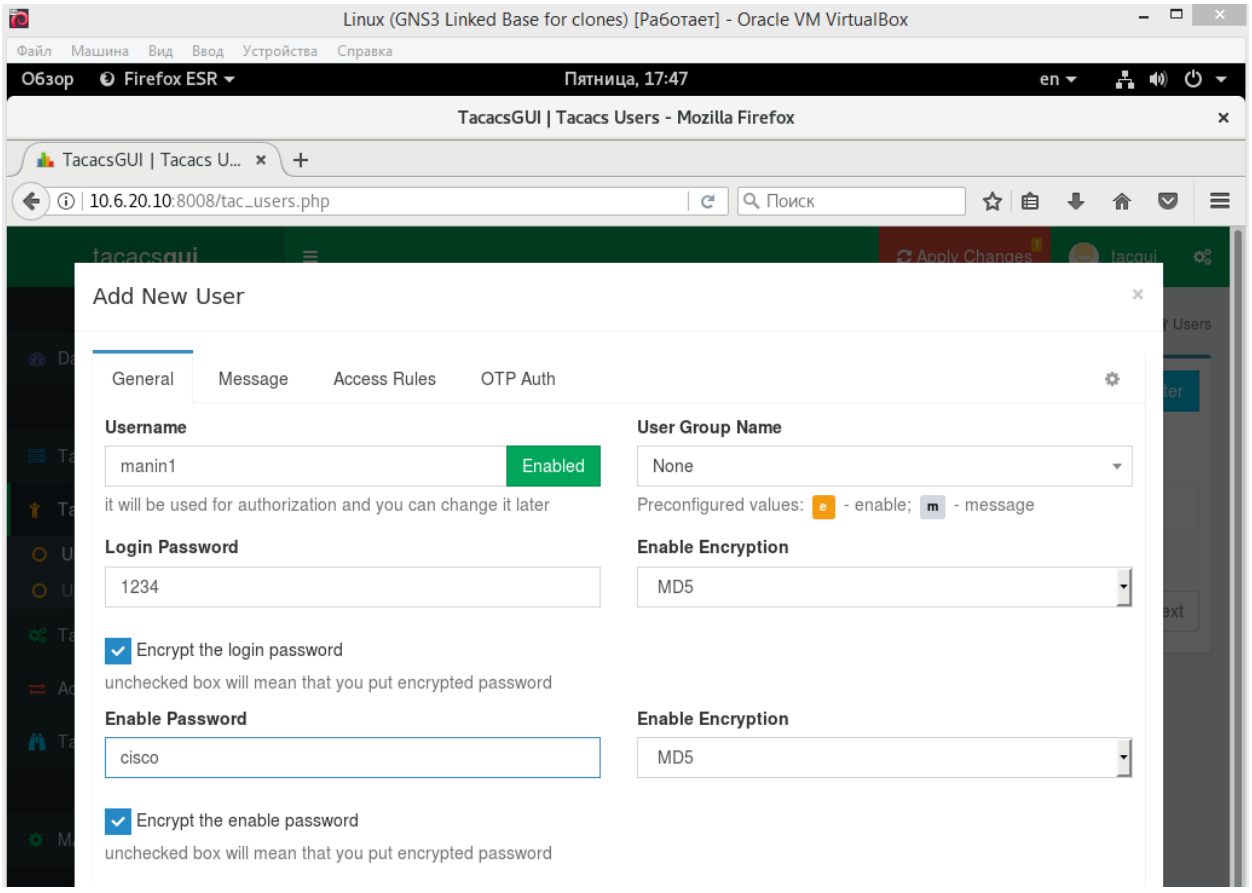

Рисунок 9.6 – Добавление пользователя

Для пользователя manin1 назначим наивысший уровень привилегий 15 (рисунок 9.7).

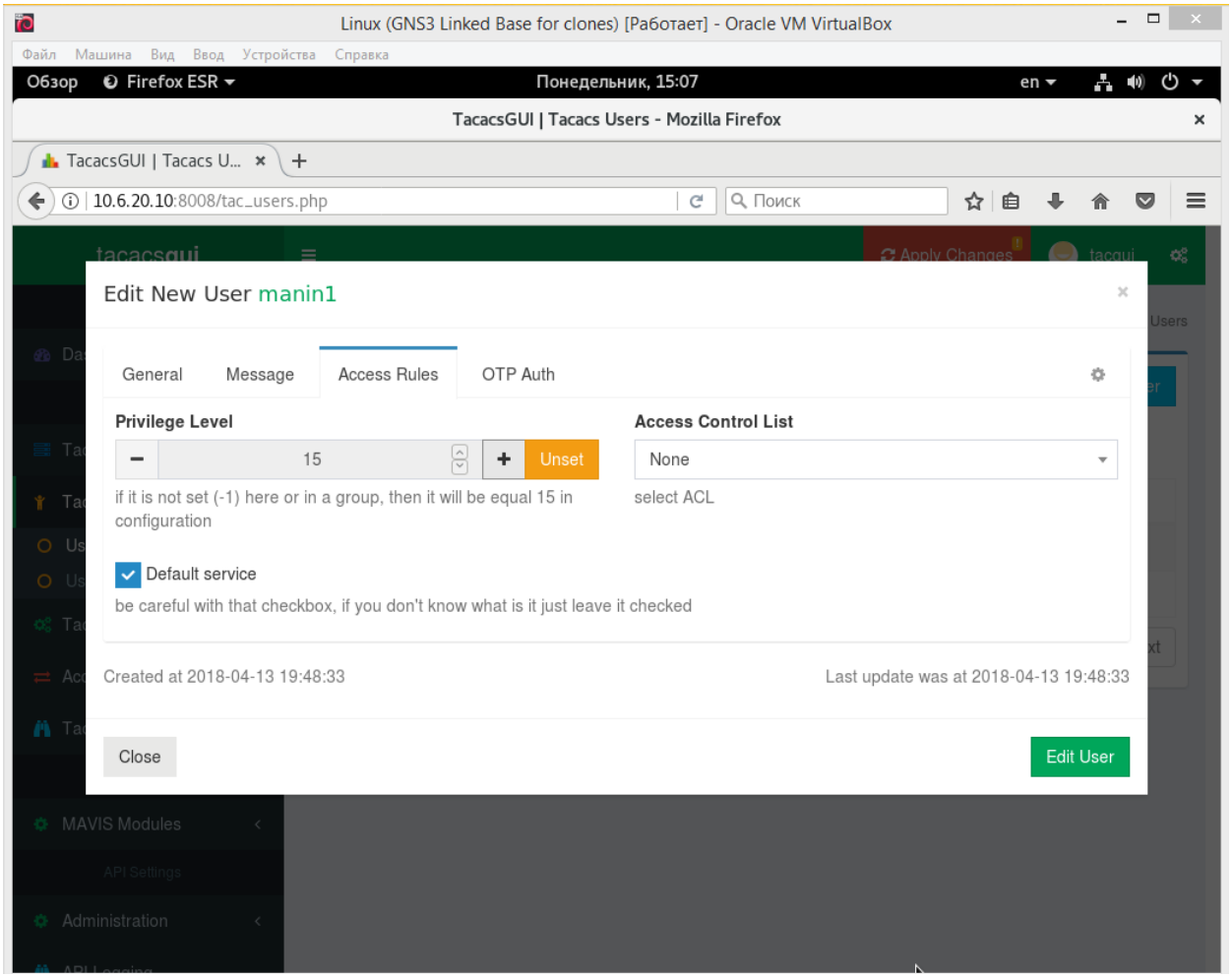

Рисунок 9.7 – Назначение уровня привилегий

На главной вкладке необходимо протестировать введенные конфигурации и применить их, рисунок 9.8.

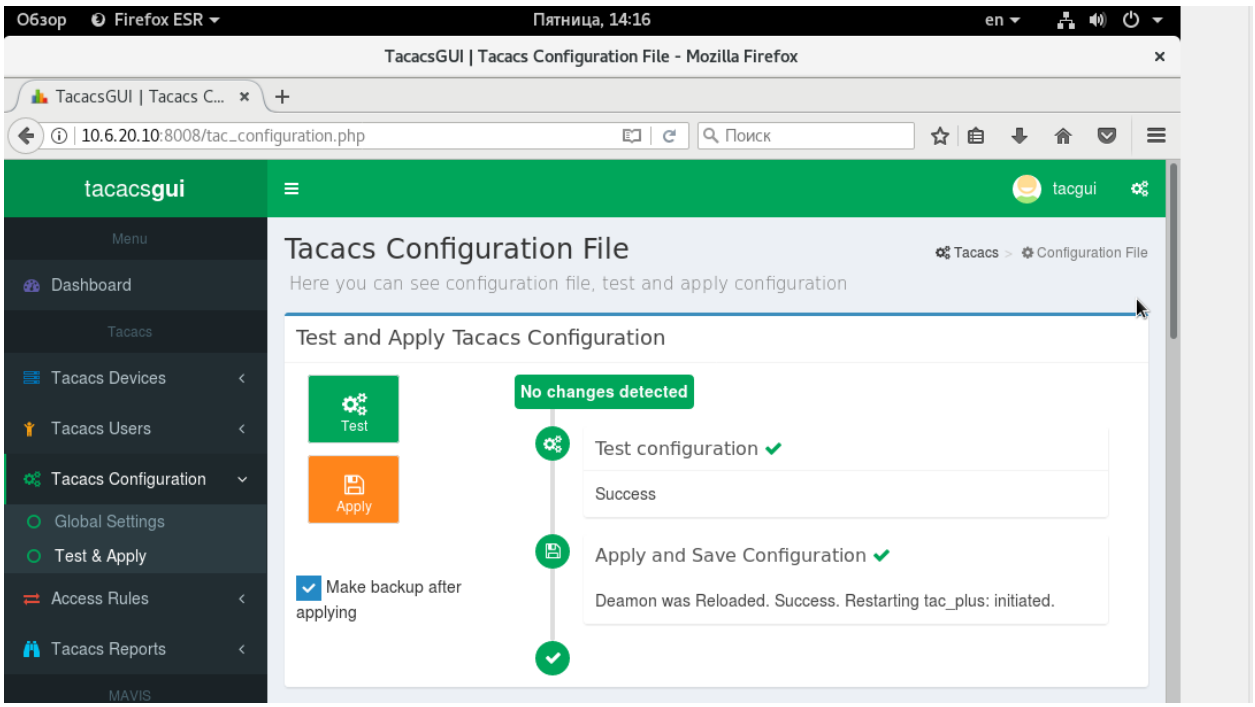

Рисунок 9.8 – Тестирование и применение конфигурации

В этом случае дополнительное конфигурирование требуется и на маршрутизаторе.

Рассмотрим сначала настройку аутентификации. Для этого необходимо на маршрутизаторе создать локальную учетную запись (на случай, если сервер станет недоступен):

R1(config)#username manin2 privilege 15 secret 1234

Кроме того, зададим пароль для перехода в привилегированный режим:

R1(config)#enable secret cisco

Необходимость создания локальной учетной записи продиктована тем, что при отсутствии технической возможности доступа к AAA-серверу должен быть обеспечен доступ по локальной базе.

Включим режим AAA на маршрутизаторе:

R1(config)#aaa new-model

Затем необходимо указать маршрутизатору параметра используемого AAA-сервера – его IP-адрес и ключ (key), который будет использоваться для шифрования информационного обмена между сервером и маршрутизатором:

R1(config)#tacacs-server host 10.6.20.10

R1(config)#tacacs-server key tacgui

Осталось создать метод аутентификации:

R1(config)#aaa authentication login default group tacacs+ local

Напомним, что параметр **login** в последней команде указывает, что создается method-list по умолчанию, то есть его действие распространяется на все интерфейсы (физические и виртуальные). Параметр **group tacacs+ local** указывает, что аутентификация производится сначала с использованием протокола TACACS+, а при невозможности (например, при недоступности сервера) – с использованием локальной базы.

Конфигурирование маршрутизатора иллюстрируется рисунком 9.9.

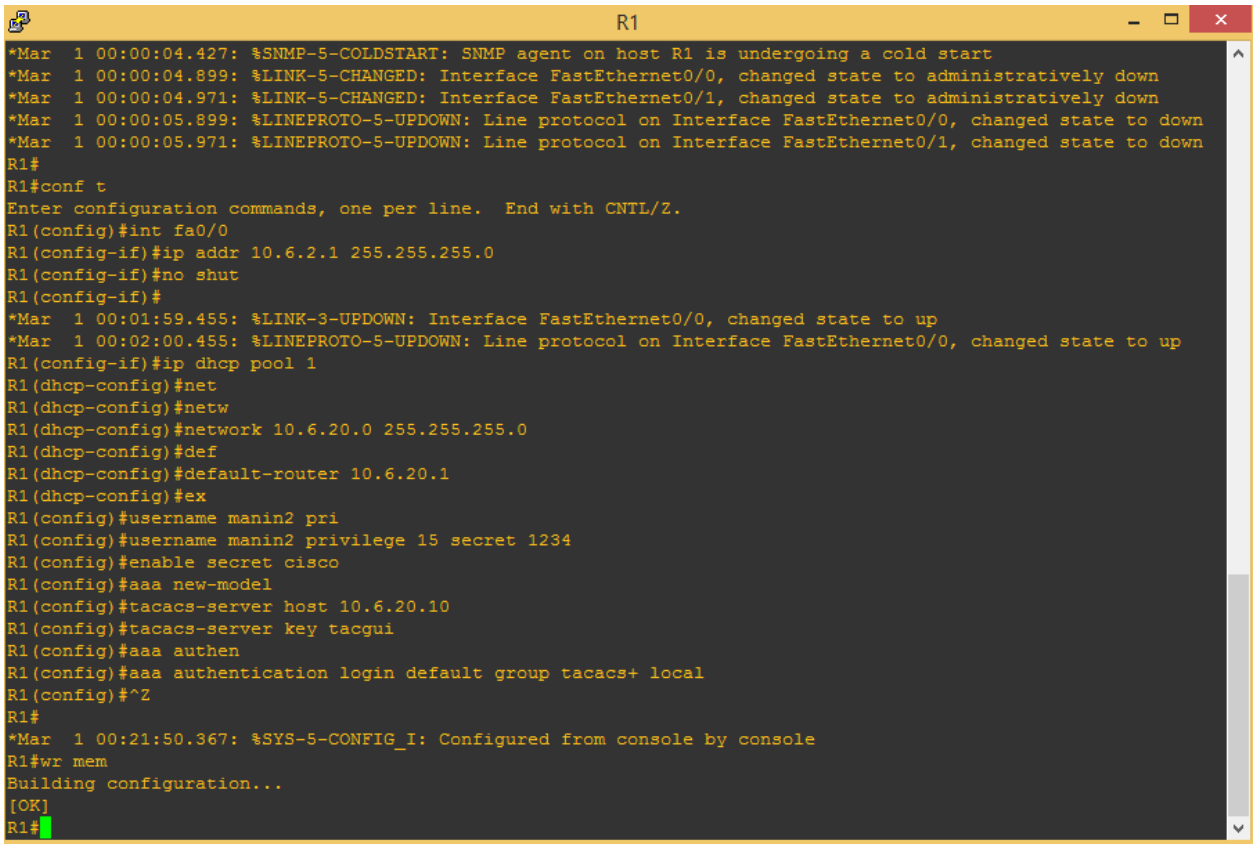

Рисунок 9.9 – Конфигурирование маршрутизатора

Добавим в сеть еще одну рабочую станцию под управлением ОС Windows XP, и попробуем удаленно подключиться к маршрутизатору, используя учетную запись manin1, сконфигурированную на AAA-сервере, рисунок 9.10.

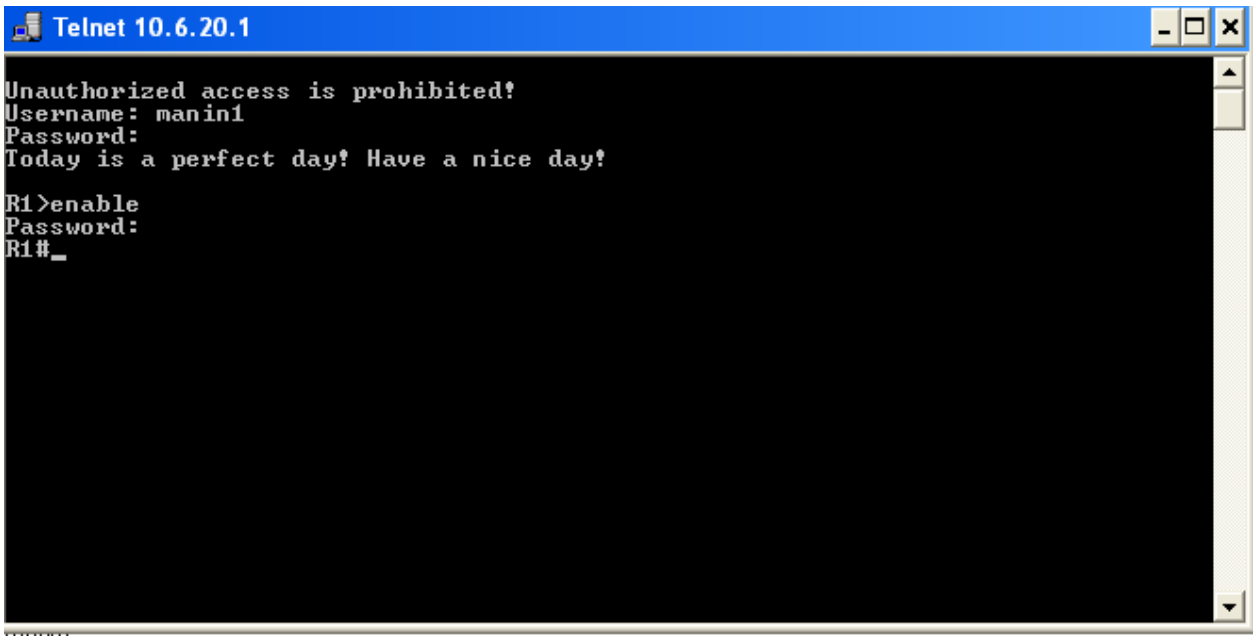

Рисунок 9.10 – Удаленное подключение к маршрутизатору с использованием AAA-сервера

Из рисунка 9.10 видно, что аутентификация прошла успешно. Отключим AAA-сервер и снова попробуем подключиться к маршрутизатору, рисунок 9.11.

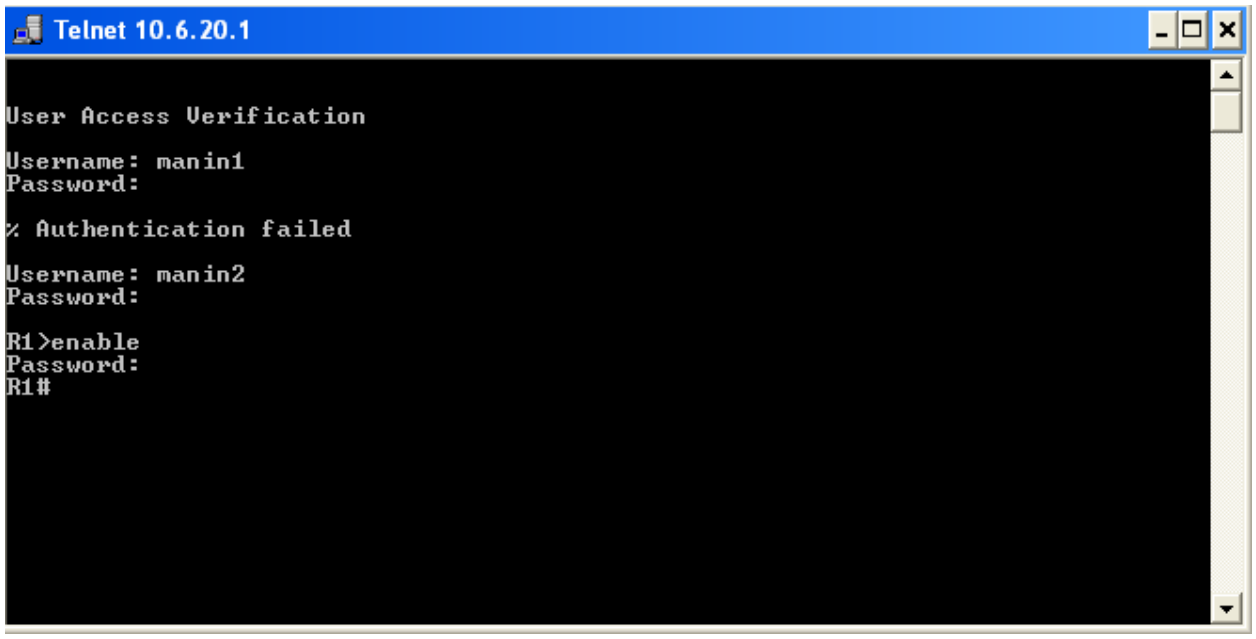

Рисунок 9.11 – Удаленное подключение к маршрутизатору при отключенном AAA-сервере

Как видно из рисунка 9.11, аутентификация по учетной записи manin1, хранящейся на сервере, оказалась безуспешной. Аутентификация по учетной записи manin2, хранящейся в локальной базе, прошла успешно благодаря параметру **group tacacs+ local**, использованному при конфигурировании маршрутизатора.

Протокол TACACS+, в отличие от RADIUS, разделяет процедуры аутентификации и авторизации. Поэтому если для нашего примера дополнительно настроить авторизацию для пользователя manin1, при удаленном доступе этот пользователь сразу попадет в привилегированный режим (так как мы настроили для него 15-й уровень привилегий, рисунок 9.7). Для большей наглядности создадим на AAA-сервере еще одного пользователя manin3 с уровнем привилегий 1 и разрешим ему использование команд **ping, show running-config** и **exit**, рисунок 9.12.

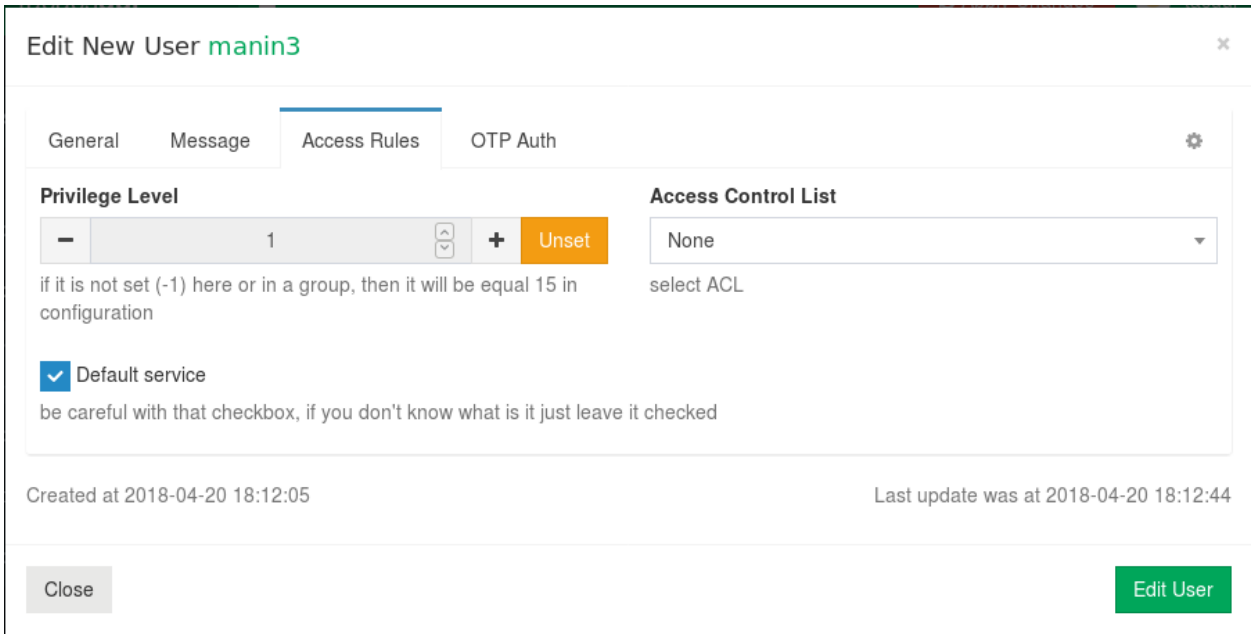

Рисунок 9.12 – Добавление пользователя manin3 с уровнем привилегий 1

Авторизация настраивается с использованием следующих команд:

R1(config)#aaa authorization exec default group tacacs+ local – включение режима авторизации (автоматический переход на нужный режим при подключении к устройству);

R1(config)#aaa authorization config-command

R1(config)# aaa authorization commands 1 default group tacacs+ local

R1(config)# aaa authorization commands 15 default group tacacs+ local

– авторизация каждой вводимой команды как на уровне 1, так и на уровне 15.

При удаленном подключении пользователя manin1 (уровень 15) мы сразу попадаем в привилегированный режим, рисунок 9.13.

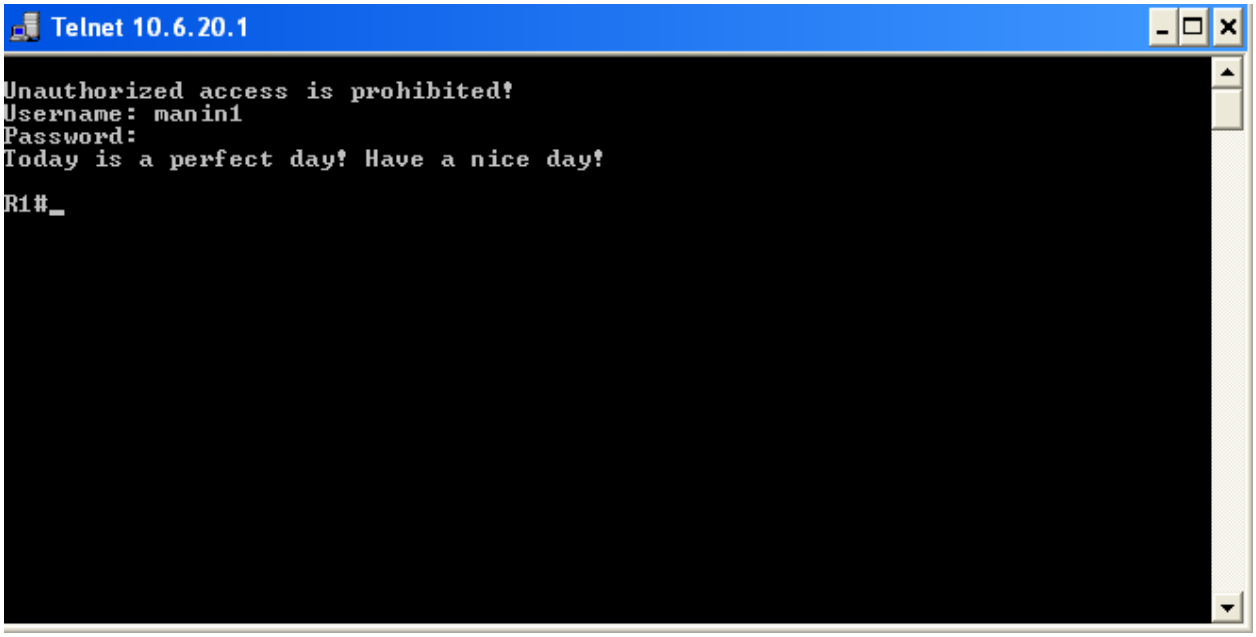

Рисунок 9.13 – Авторизованный вход

9.5 Отчет по работе:

– Демонстрация процесса аутентификации и авторизации двух пользователей с различными правами.

Практическое занятие №10

Конфигурирование межсетевых экранов с пакетной фильтрацией

10.1 Цель работы: Привитие навыков конфигурирования межсетевых экранов с пакетной фильтрацией.

10.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Windows пакетом Cisco Packet Tracer или GNS3.

10.3 Задание:

- В Cisco Packet Tracer создать сеть, состоящую из пяти подсетей. В одной из подсетей установить два сервера, один из которых должен быть сконфигурирован как FTP, а другой – как WEB-сервер;
- Всем компьютерам подсети 192.169.X.0 (где X – номер студента в списке группы) предоставить полный доступ ко всем серверам;
- Всем компьютерам подсети 192.169.X+1.0 предоставить доступ только к FTP-серверу по протоколу FTP;
- Всем компьютерам подсети 192.169.X+2.0 предоставить доступ только к WEB-серверу;
- Компьютерам оставшейся подсети запретить доступ к внешним ресурсам.

10.4 Порядок выполнения работы

Как известно [7], разделяют межсетевые экраны (FireWall) с пакетной фильтрацией и с сохранением состояний (экраны прикладного уровня). Рассмотрим сначала использование пакетной фильтрации.

Межсетевые экраны с фильтрацией пакетов представляют собой маршрутизаторы (например, Cisco) или работающие на сервере программы, сконфигурированные таким образом, чтобы фильтровать входящие и исходящие пакеты. Поэтому такие экраны называют иногда пакетными

фильтрами. Фильтрация осуществляется путем анализа IP-адреса источника и получателя, а также портов входящих TCP- и UDP-сегментов и сравнением их с сконфигурированной таблицей правил. Эти межсетевые экраны просты в использовании, дешевы, оказывают минимальное влияние на производительность вычислительной системы. Основным недостатком является их уязвимость при подмене адресов IP. Во всей линейке оборудования Cisco Systems пакетная фильтрация реализована с помощью так называемых списков контроля доступа (Access Control List).

Списки доступа ACL могут быть созданы для всех сетевых протоколов, функционирующих на маршрутизаторе, например IP или IPX, и устанавливаются на интерфейсах маршрутизаторов. Запрет или разрешение сетевого трафика через интерфейс маршрутизатора реализуется на основании анализа совпадения определенных условий. Для этого списки доступа представляются в виде последовательных записей, в которых используют адреса и протоколы. Сетевые фильтры (списки доступа) создаются для входящих или исходящих пакетов на основании анализируемых параметров (адреса источника, адреса назначения, протокола и номера порта верхнего уровня), указанных в списке доступа ACL (рисунок 10.1).

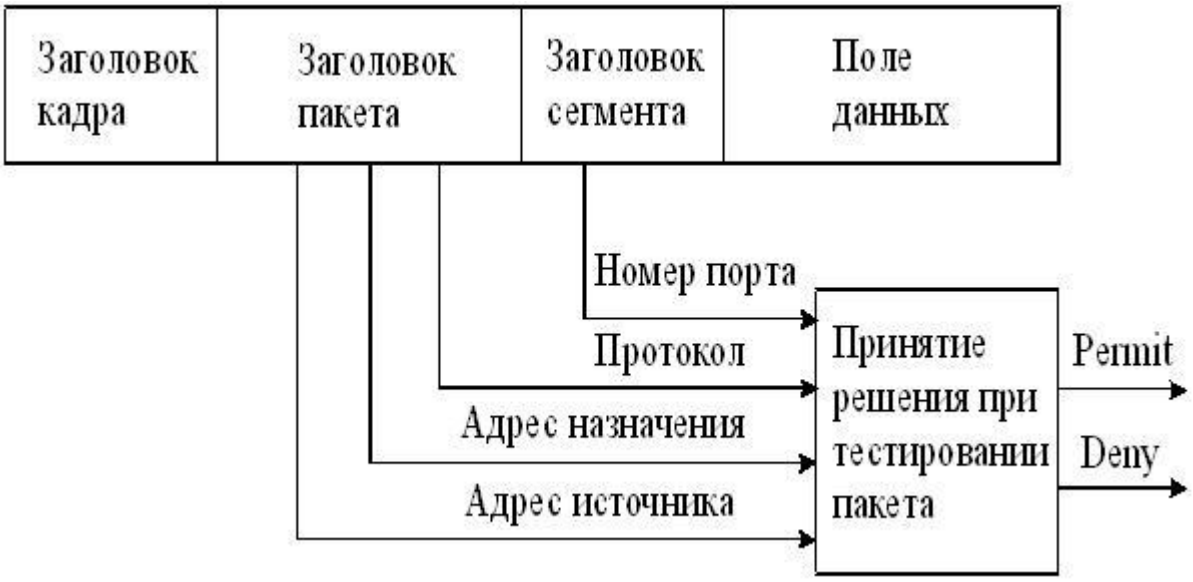

Рисунок 10.1 – Анализ заголовков пакета

Как видно из рисунка 10.1, на основе проведенного анализа служебной информации, устройство, реализующее межсетевое экранирование, принимает решение о дальнейшей передаче (permit) или о фильтрации (deny).

Списки доступа могут быть определены для каждого установленного на интерфейсе протокола и для каждого направления сетевого трафика (исходящего и входящего). Поэтому для входящего и исходящего трафиков через интерфейс создаются отдельные списки.

Если списки доступа не формируются на маршрутизаторе, то все проходящие через маршрутизатор пакеты будут иметь доступ к сети.

Список доступа ACL составляется из утверждений (условий), которые определяют, следует ли пакеты принимать или отклонять во входных и выходных интерфейсах маршрутизатора. Программное обеспечение IOS Cisco проверяет пакет последовательно по каждому условию. Если условие, разрешающее продвижение пакета, расположено наверху списка, никакие условия, добавленные ниже его, не будут запрещать продвижение пакета. Если в списке доступа необходимы дополнительные условия, то список целиком должен быть удален и создан новый с новыми условиями.

Функционирование маршрутизатора по проверке соответствия принятого пакета требованиям списка доступа производится следующим образом. Когда кадр поступает на интерфейс, маршрутизатор проверяет IP-адрес. Если адрес назначения соответствует адресу интерфейса, то маршрутизатор извлекает (декапсулирует) из кадра пакет и проверяет его на соответствие условиям списка ACL входного интерфейса. При отсутствии запрета или отсутствии списка доступа пакет инкапсулируется в новый кадр второго уровня и отправляется интерфейсу следующего устройства.

Проверка условий (утверждений) списка доступа производится последовательно. Если текущее утверждение верно, пакет обрабатывается в соответствии с командами **permit** или **deny** списка доступа, остальная часть условий ACL не проверяется. Если все утверждения ACL неверны, то неявно

заданная по умолчанию команда **deny any** (запретить все остальное) в конце списка не позволит передавать дальше по сети несоответствующие пакеты.

Существуют разные типы списков доступа: стандартные (standard ACLs), расширенные (extended ACLs) и именованные (named ACLs). Когда список доступа конфигурируется на маршрутизаторе, каждый список должен иметь уникальный идентификационный номер или уникальное имя. Номер идентифицирует тип созданного списка доступа и должен находиться в пределах определенного диапазона, заданного для этого типа списка (таблица 10.1).

Таблица 10.1 – Диапазоны идентификационных номеров ACL

Диапазон номеров	Название списка доступа
1-99	IP standard access-list
100-199	IP extended access-list
1300-1999	IP standard access-list (extended range)
2000-2699	IP extended access-list (extended range)
600-699	Appletalk access-list
800-899	IPX standard access-list
900-999	IPX extended access-list

В стандартном списке доступа для принятия решения в IP-пакете анализируется только адрес источника сообщения, чтобы фильтровать сеть (IPX-стандарт может фильтровать как адрес источника, так и назначения).

Расширенные списки доступа (Extended Access Lists) проверяют как IP-адрес источника, так и IP-адрес назначения, поле протокола в заголовке пакета сетевого уровня и номер порта в заголовке транспортного уровня.

Таким образом, для каждого протокола, для каждого направления трафика и для каждого интерфейса может быть создан свой список доступа. Исходящие фильтры не затрагивают трафик, который идет из местного маршрутизатора.

Из рекомендаций по установке списков доступа можно отметить следующее. Стандартные списки доступа рекомендуется устанавливать по

возможности ближе к адресату назначения, а расширенные – ближе к источнику. Поэтому стандартные списки доступа должны блокировать устройство назначения и располагаться поближе к защищаемой сети, а расширенные списки доступа должны быть установлены близко к источнику сообщений.

Список доступа производит фильтрацию пакетов по порядку, поэтому в строках списков следует задавать условия фильтрации, начиная от специфических условий и заканчивая общими. Условия списка доступа обрабатываются последовательно от вершины списка к основанию, пока не будет найдено соответствующее условие. Если никакое условие не найдено, тогда пакет отклоняется и уничтожается, поскольку неявное условие **deny any** (запретить все остальное) присутствует неявно в конце любого списка доступа. Не удовлетворяющий списку доступа пакет протокола IP будет отклонен и уничтожен, при этом отправителю будет послано сообщение ICMP. Новые записи (линии) всегда добавляются в конце списка доступа.

Конфигурирование списков доступа производится в два этапа:

1. Создание списка доступа в режиме глобального конфигурирования.
2. Привязка списка доступа к интерфейсу в режиме детального конфигурирования интерфейса.

Формат команды создания стандартного списка доступа следующий:

Router(config)#access-list {№} {permit / deny} {адрес источника}.

Списки доступа могут фильтровать как трафик, входящий в маршрутизатор (in), так и трафик, исходящий из маршрутизатора (out). Направление трафика указывается при привязке списка доступа к интерфейсу. Формат команды привязки списка к интерфейсу следующий:

Router(config-if)#{протокол} access-group {номер} {in или out}.

После привязки списка доступа его содержимое не может быть изменено. Не удовлетворяющий администратора список доступа должен быть удален командой **no access-list** и затем создан заново.

Расширенный список доступа создается командой:

Router(config)#access-list {№} {permit / deny} {трансп.протокол} {адр.ист} {адр.пол.} eq {№ порта или название прикладного протокола}

Правила назначения в списке доступа номера порта (или, что тоже самое, прикладного протокола) представлены в таблице 10.2.

Таблица 10.2 – Правила назначения прикладных протоколов

Обозначение	Действие
lt n	Все номера портов, меньшие n.
gt n	Все номера портов, большие n.
eq n	Порт n
neq n	Все порты, за исключением n.
range n m	Все порты от n до m включительно.

Распространенные прикладные протоколы и соответствующие им стандартные номера портов приведены в таблице 10.3.

Таблица 10.3 – Номера портов некоторых прикладных протоколов

Номер порта	Транспортный протокол	Прикладной протокол	Ключевое слово в команде access-list
20	TCP	FTP	data ftp_data
21	TCP	Управление сервером FTP	ftp
22	TCP	SSH	
23	TCP	Telnet	telnet
25	TCP	SMTP	Smtp
53	UDP, TCP	DNS	Domain
67, 68	UDP	DHCP	nameserver
69	UDP	TFTP	Tftp
80, 8080	TCP	HTTP (WWW)	www
110	TCP	POP3	pop3
161	UDP	SNMP	Snmp

Рассмотрим пример создания стандартного списка доступа для сети, схема которой показана на рисунке 10.2. На рисунке укажем узлы, находящиеся в подсетях 192.168.0.0/24 и 192.168.1.0/24.

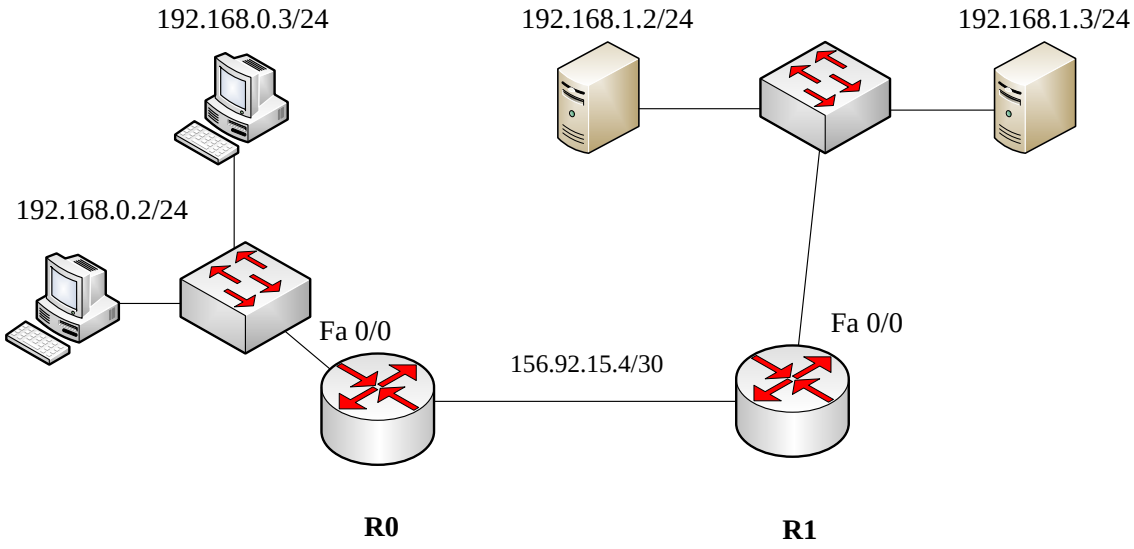

Рисунок 10.2 – Схема сети

Предположим, что к серверу, находящемуся в подсети 192.168.1.0/24 по адресу 192.168.1.2/24, доступ из подсети 192.168.0.0/24 разрешен только компьютеру 192.168.0.2/24. Это правило можно сконфигурировать с использованием стандартного списка доступа на интерфейсе Fa 0/0 маршрутизатора R1.

Для этого в режиме глобального конфигурирования на маршрутизаторе R1 необходимо выполнить следующие команды:

Router1(config)#access-list 10 permit 192.168.0.2

Router1(config)#interface fa 0/0

Router1(config-if)#ip access-group 10 out

Первая команда создает на маршрутизаторе список доступа с номером 10, который разрешает (permit) передачу пакетов с адресом источника 192.168.0.2.

Вторая команда является командой перехода к конфигурированию интерфейса Fa 0/0.

Третья команда привязывает список доступа с номером 10 к интерфейсу Fa 0/0 и указывает на направление передачи – исходящее (out).

Созданный таким образом список доступа будет состоять из двух строк. Первая строка в явной форме разрешает передавать на интерфейс маршрутизатора Fa 0/0 пакеты с адресом источника 192.168.0.2. Вторая строка в неявном виде запрещает (deny any) передавать на этот интерфейс все остальные пакеты.

Проанализируем действия маршрутизатора R1 при поступлении на его внешний интерфейс пакета после создания списка доступа.

Если пакет поступил из подсети 156.92.15.4 и предназначен серверу 192.168.1.2, маршрутизатор, определив по таблице маршрутизации выходной интерфейс, передает этот пакет в буфер интерфейса Fa 0/0.

Затем анализируется список, начиная с первой строки. Если источник имеет адрес 192.168.0.2 (совпадение в первой строке списка произошло), пакет инкапсулируется в кадр Ethernet и передается серверу. Если источник имеет любой другой адрес (совпадения в первой строке списка не произошло), происходит обращение ко второй неявной строке списка (deny any), и пакет отбрасывается.

В случае, если необходимо обеспечить доступ к серверу и второго компьютера подсети 192.168.0.0/24 с адресом 192.168.0.3, команды конфигурирования будут выглядеть следующим образом:

Router1(config)#access-list 10 permit 192.168.0.2

Router1(config)#access-list 10 permit 192.168.0.3

Router1(config)#interface fa 0/0

Router1(config-if)#ip access-group 10 out

Очевидно, что список доступа теперь содержит три строки – две явные и одну неявную.

Очевидно, что рассмотренный способ конфигурирования списков доступа удобен в том случае, если доступ к какому-либо ресурсу (серверу) необходимо обеспечить небольшому количеству источников (компьютеров). Если же, например, в подсети 192.168.0.0/24 значительное количество компьютеров, такое конфигурирование становится неудобным и

подверженным ошибкам, так как для каждого из них необходимо отдельно создавать строку списка.

Поэтому при создании списков доступа можно использовать wildcard маски. В этом случае в строке списка может содержаться указание на передачу или фильтрацию пакетов не с адресами конечных узлов, а с адресами сетей (подсетей), в которые они входят.

Правило использования масок в этом случае можно сформулировать следующим образом – нулевые значения разрядов маски означают требование обработки соответствующих разрядов адреса, а единичные значения разрядов маски означают игнорирование соответствующих разрядов адреса. Например, если wildcard маска имеет вид 0.0.0.0, то проверять условие необходимо для всех разрядов адреса источника прибывшего пакета. Если же маска имеет вид 0.0.0.255, то проверять условие необходимо только для первых трех байтов адреса источника.

Предположим, что доступ к тому же серверу (адрес 192.168.1.2) должны получить все компьютеры подсети 192.168.0.0/24. В этом случае на маршрутизаторе R1 необходимо выполнить команды:

Router1(config)#access-list 10 permit 192.168.0.0 0.0.0.255

Router1(config)#interface fa 0/0

Router1(config-if)#ip access-group 10 out

Необходимо отметить, что, если нужно разрешить какому-либо одному узлу из другой подсети (например, 192.168.2.2/24) доступ к этому же серверу, создаваемый список необходимо дополнить командой

Router1(config)#access-list 10 permit 192.168.2.2 0.0.0.0

или, что то же самое, командой

Router1(config)#access-list 10 permit host 192.168.2.2

Создание списков доступа очень похоже на создание «белых» и «черных» списков на телефоне. При создании «белого» списка принимать разрешено только вызовы от источников, номера которых внесены в «белый» список, остальные вызовы отбрасываются. При использовании «черного»

списка отбрасываются только вызовы от источников, внесенных в список, остальные вызовы принимаются. Основное отличие от телефонных вызовов состоит в том, что в списки **permit** и **deny** вносятся не телефонные номера, а значения заголовков различных уровней.

Используя эту аналогию с «белыми» и «черными» списками телефона, можно отметить, что рассмотренные способы аналогичны созданию в телефоне «белых» списков – указанные в списке доступа адреса являются разрешенными, остальные – запрещенными.

В ряде случаев более удобным является использование аналогии «черного» списка – разрешено передавать данные от всех, кроме тех, кто указан в черном списке.

Предположим, что к тому же серверу необходимо обеспечить доступ всем компьютерам, кроме одного, имеющего адрес 192.168.0.15. Конфигурирование такого списка будет иметь вид:

Router1(config)#access-list 11 deny host 192.168.0.15

Router1(config)#access-list 11 permit any

Router1(config)#interface fa 0/0

Router1(config-if)#ip access-group 11 out

Напомним, что по умолчанию у создаваемых списков доступа неявно присутствует заключительная строка **deny any** – запретить все. В данном случае мы заменили эту строку на **permit any** – разрешить все. Соответственно, доступ к серверу будет разрешен всем, кроме компьютера с адресом 192.168.0.15.

Рассмотрим теперь применение расширенного списка для конфигурирования маршрутизатора R1 (рисунок 3.2), при этом должны быть выполнены следующие условия:

- компьютеру 192.168.0.2/24 необходимо предоставить доступ к web-серверу с адресом 192.168.1.2 по протоколу WWW;

- всем компьютерам подсети 192.168.0.0/24 необходимо предоставить доступ к FTP-серверу с адресом 192.168.1.3 по протоколу FTP.

Команды конфигурирования в этом случае будут выглядеть следующим образом:

Router1(config)#access-list 110 permit tcp host 192.168.0.2 host 192.168.1.2 eq www
Router1(config)#access-list 110 permit tcp 192.168.0.0 0.0.0.255 host 192.168.1.3 eq ftp
Router1(config)#interface fa 0/0
Router1(config-if)#ip access-group 110 out

Очевидно, что указанный способ аналогичен созданию «белого» списка в телефоне, так как третье неявное условие, находящееся в конце списка, блокирует все, что не разрешено.

Рассмотрим пример, когда удобнее использовать аналогию «черного» списка в телефоне.

На маршрутизаторе R1 должны быть выполнены следующие условия:

- компьютеру 192.168.0.2/24 необходимо запретить доступ к серверу с адресом 192.168.1.2 по протоколу WWW, но разрешить доступы к другим сервисам;

- всем компьютерам подсети 192.168.0.0/24 необходимо запретить доступ к серверу с адресом 192.168.1.3 по протоколу FTP, но разрешить доступ к другим сервисам.

Команды конфигурирования в этом случае будут иметь вид:

Router1(config)#access-list 110 deny tcp host 192.168.0.2 host 192.168.1.2 eq www
Router1(config)#access-list 110 deny tcp 192.168.0.0 0.0.0.255 host 192.168.1.3 eq ftp
Router1(config)#access-list 110 permit ip any any
Router1(config)#interface fa 0/0
Router1(config-if)#ip access-group 110 out

Запись **permit ip any any** означает, что весь остальной трафик от любого источника к любому получателю должен передаваться.

Просмотреть созданные на маршрутизаторе списки доступа можно по команде **show access-list**, а списки, настроенные на конкретных интерфейсах, командами **show ip interfaces** или **show running-config**. На рисунке 10.3 показаны настроенные списки доступа для рассмотренного здесь примера.

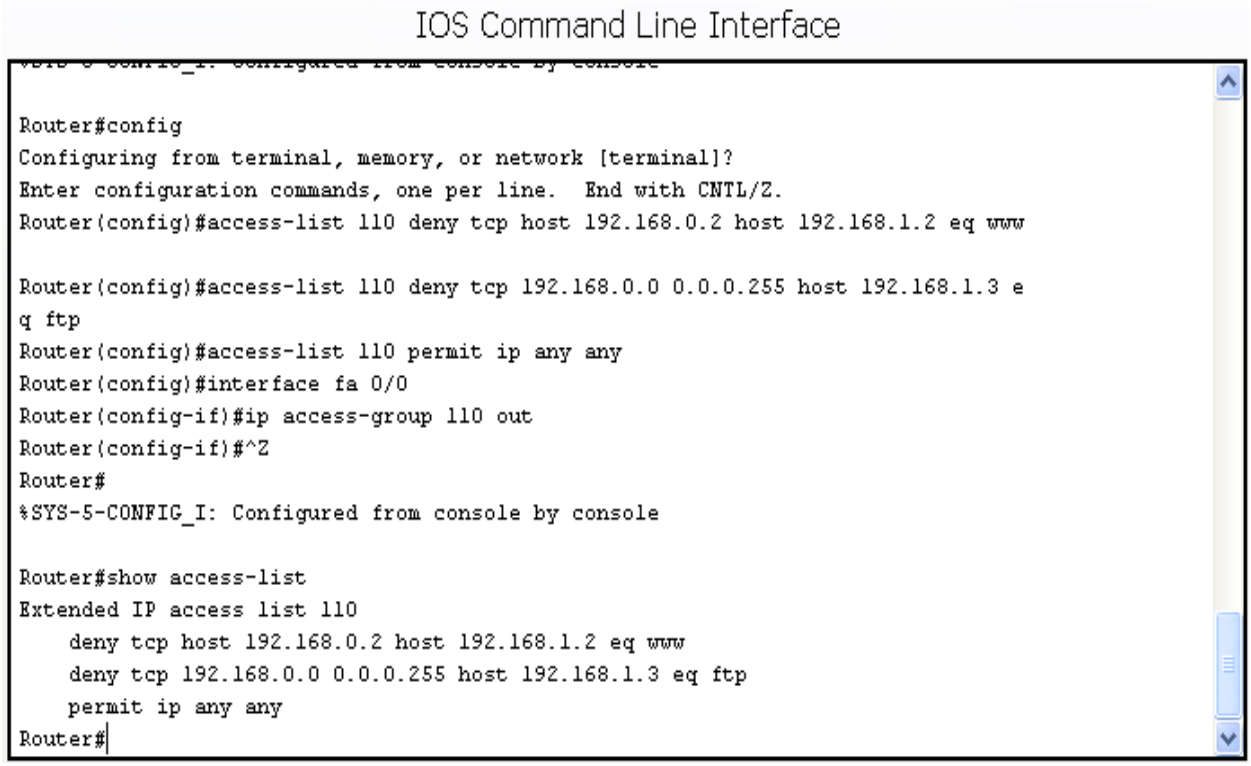

Рисунок 10.3 – Конфигурирование списка доступа и его просмотр

Списки доступа также желательно использовать и для конфигурирования удаленного доступа к устройствам (глава 1).

В настоящее время чаще используются не нумерованные, а именованные списки доступа. Удобство использования именованных списков доступа заключается прежде всего в том, что названию списка можно придать определенный смысл (INTERNET, ADMIN, FTP, и т.д.). Так как именованный список не имеет номера, который однозначно определяет его вид (таблица 3.1), при создании такого списка необходимо явно указать, какой именно список создается – стандартный или расширенный. Команда создания именованного списка доступа имеет вид:

ip access-list <standard/extended> <имя>

<правило 1>

<правило 2>

<правило n>

Параметр **standard/extended** указывает на вид создаваемого списка, а правила прописываются аналогично нумерованным спискам.

10.5 Отчет по работе:

Демонстрация корректной работы созданных списков доступа.

Практическое занятие №11

Конфигурирование межсетевых экранов с сохранением состояний

11.1 Цель работы: Привитие навыков конфигурирования межсетевых экранов с сохранением состояний.

11.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Windows пакетом Cisco Packet Tracer или GNS3.

11.3 Задание:

- Используя Cisco Packet Tracer, построить сеть, показанную на рисунке 11.1. Произвести конфигурирование сетевых устройств для обеспечения доступа всех серверов из внутренней сети;
- Убедиться в возможности доступа во внутреннюю сеть извне с использованием произвольного протокола;
- Настроить инспектирование TCP-трафика и сконфигурировать список доступа. Убедиться в доступности серверов из внутренней сети и недоступности ресурсов внутренней сети извне.

11.4 Порядок выполнения работы

Такие межсетевые экраны еще называют экранами с сохранением сессий (statefull firewall). Суть заключается в том, что при запросе на установление соединения (например, TCP-сессии) маршрутизатор запоминает эту сессию и при поступлении извне пакета сверяет его со всеми текущими сессиями. Если принятый извне пакет относится к какой-либо текущей сессии, он продвигается во внутреннюю сеть, в противном случае – отбрасывается.

Для конфигурирования межсетевого экранирования на устройствах Cisco необходимо в явном виде указать, трафик каких протоколов должен отслеживаться (инспектироваться). Для этого используется команда

ip inspect name <имя правила> <название протокола>

Данная команда выполняется в режиме глобального конфигурирования.

Аналогично спискам доступа, созданное правило необходимо привязать к интерфейсу с указанием направления передачи:

R1(config)#int fa0/0 – переход в режим конфигурирования интерфейса fa 0/0;

R1(config-if)#ip inspect <имя правила> <in/out> - привязка правила к интерфейсу с указанием направления передачи.

Необходимо отметить, что правило можно привязывать как к внутреннему, так и ко внешнему интерфейсу маршрутизатора, однако направление передачи должно соответствовать направлению запросов из внутренней сети ко внешней. Соответственно, если правило привязывается к внутреннему интерфейсу, направление передачи – входящее (in), если к внешнему – исходящее (out).

Приведем пример межсетевого экранирования для сети, показанной на рисунке 11.1. Для удобства разместим маршрутизатор R1 во внешней сети, в которой также располагаются два сервера – web-сервер и ftp-сервер. Произведем настройку всего оборудования таким образом, чтобы из внутренней сети были доступны оба сервера.

Создадим правило для инспектирования запросов к web-серверу (протокол HTTP) с именем HTTP и привяжем его к внутреннему интерфейсу fa0/0 с входящим направлением передачи (рисунок 11.2).

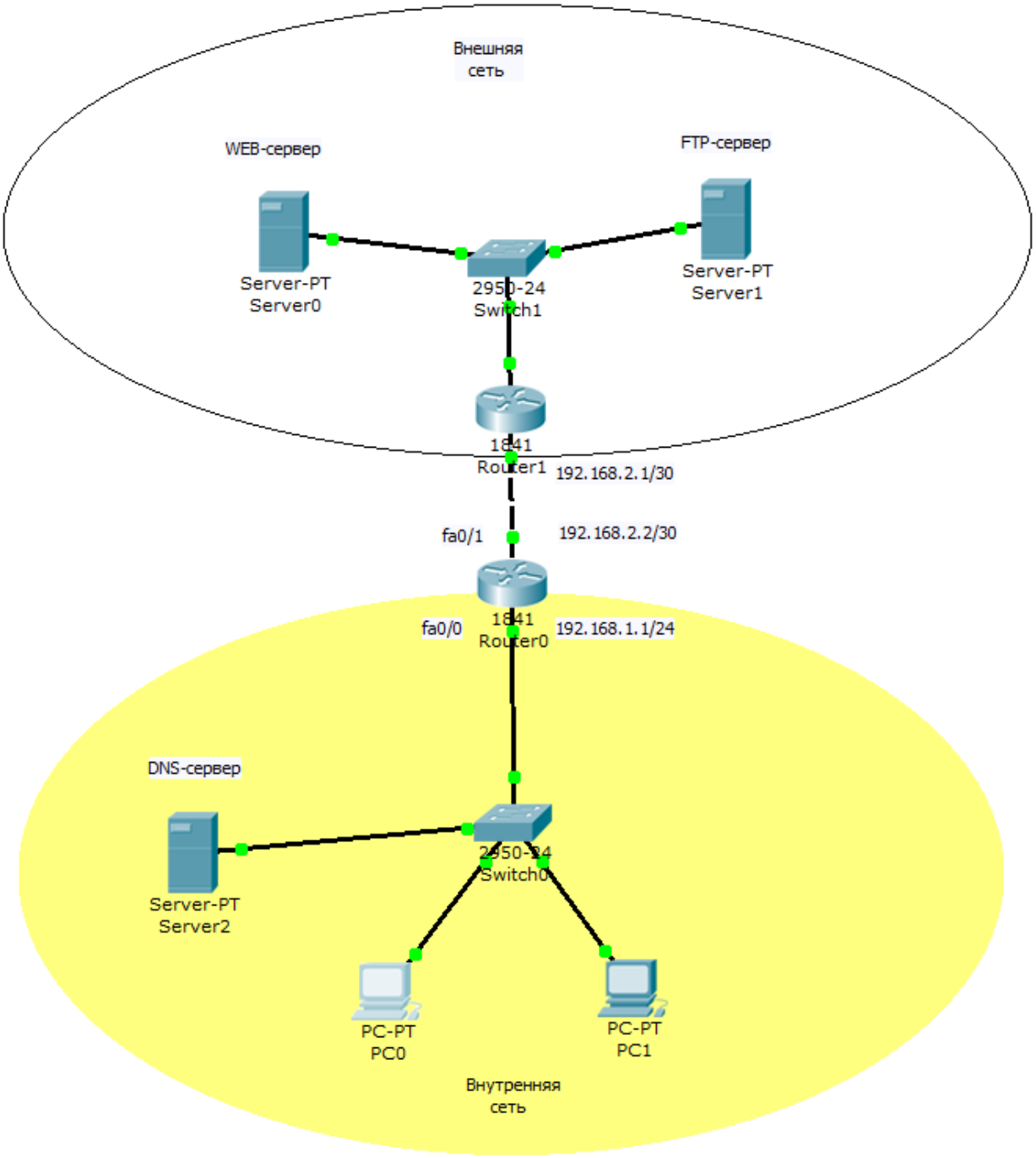

Рисунок 11.1 – Пример сети

```
Router>en
Router#conf
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip inspect name HTTP http
Router(config)#int fa0/0
Router(config-if)#ip inspect HTTP in
Router(config-if)#
```

Рисунок 11.2 – Конфигурирование инспектирования протокола HTTP

Следует отметить (и это очень важно!), что инспектирование трафика необходимо применять совместно со списками доступа. В нашем примере, когда списки доступа не были созданы, http-запросы, поступающие из внутренней сети, будут инспектироваться. Однако если из внешней сети также поступит http-запрос, он пройдет во внутреннюю сеть, так как не существует списка доступа, обеспечивающего фильтрацию этого запроса.

Для проверки этого разместим во внешней сети ПК с адресом 213.80.65.4 и попробуем соединиться с сервером внутренней сети по протоколу HTTP (рисунок 11.3).

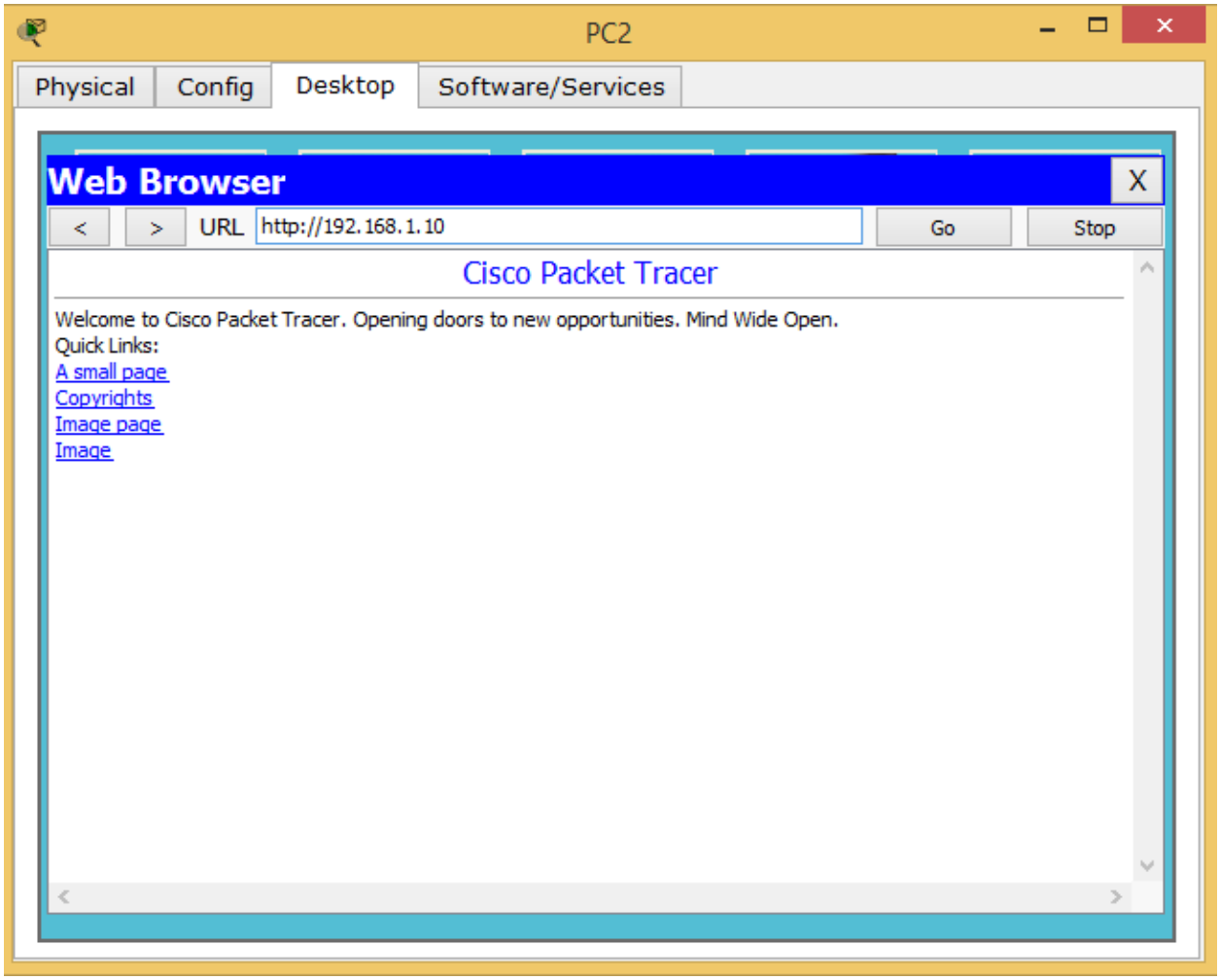

Рисунок 11.3 – Соединение с сервером внутренней сети по протоколу HTTP

В качестве внутреннего сервера мы использовали DNS-сервер с адресом 192.168.1.10/24. Как видно из рисунка 11.3, соединение прошло успешно.

Для защиты внутренней сети создадим на маршрутизаторе R0 список доступа, запрещающий передачу всех IP-пакетов, и привяжем его к внешнему интерфейсу fa0/1 с указанием входящего направления (рисунок 11.4).

```
Router>en
Router#conf
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip access-list ex FRW
Router(config-ext-nacl)#deny ip any any
Router(config-ext-nacl)#
```

Рисунок 11.4 – Создание списка доступа

Теперь снова попытаемся послать HTTP-запрос из внешней сети (рисунок 11.5).

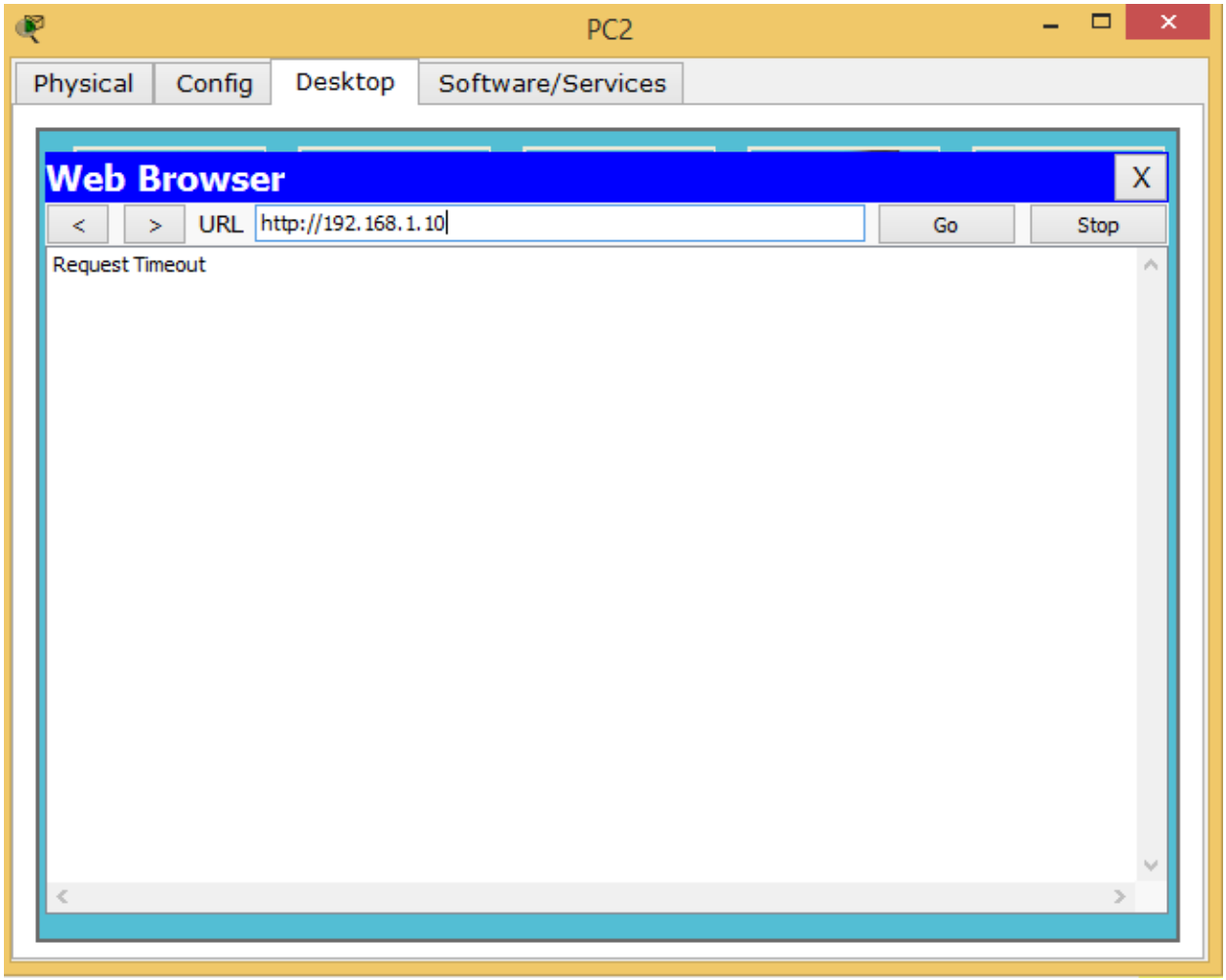

Рисунок 11.5 – Отсутствие соединения с сервером внутренней сети по протоколу HTTP

Как видно из рисунка 11.5, из внешней сети сервер не доступен. При этом из внутренней сети web-сервер остается доступным.

С учетом созданного списка доступа FRW остальные протоколы (кроме HTTP) не инспектируются. Следовательно, если послать из внутренней сети запрос по какому-либо другому протоколу, ответ получен не будет, так как его заблокирует список доступа на внешнем интерфейсе маршрутизатора. Попробуем получить из внутренней сети доступ к ftp-серверу (рисунок 11.6).

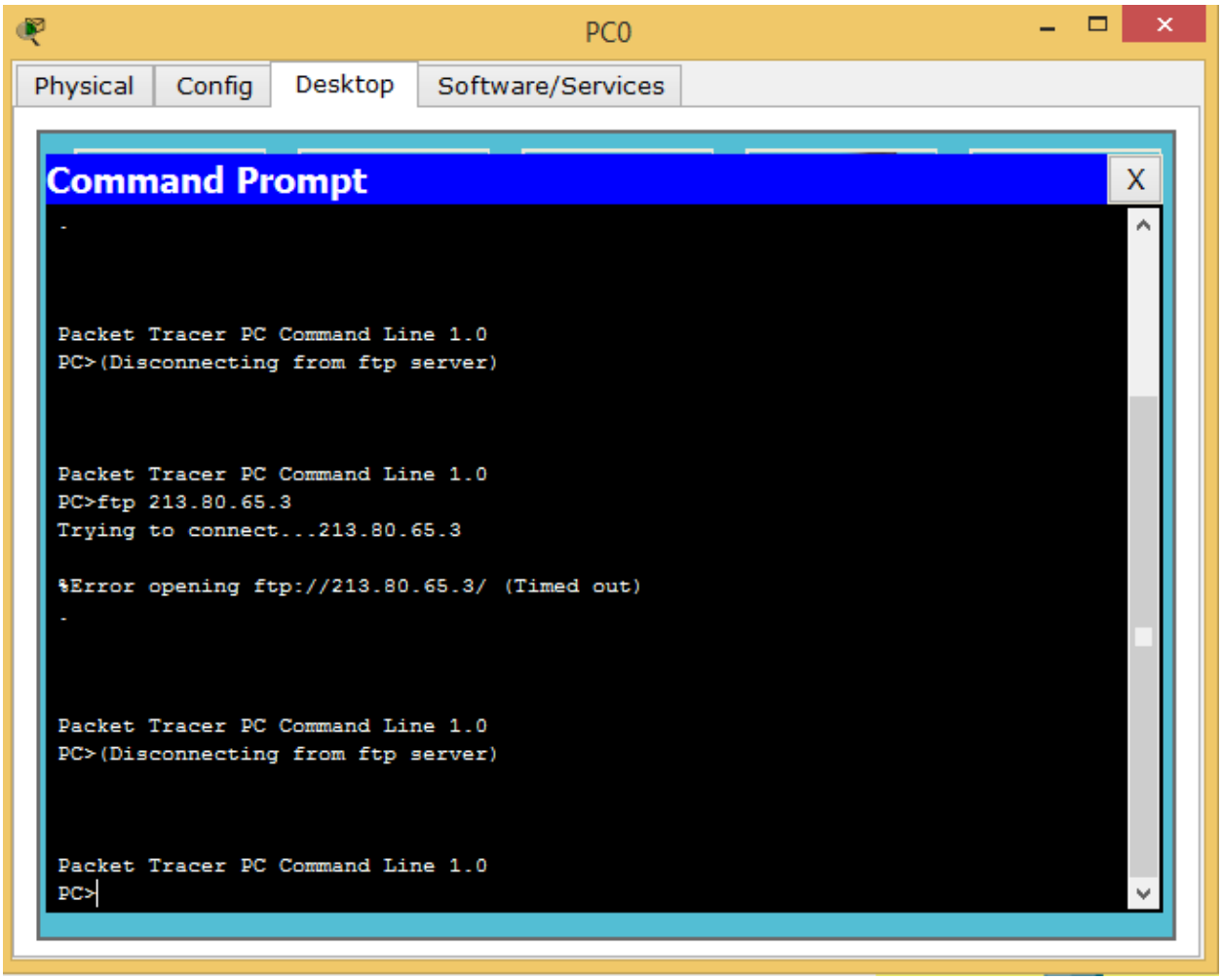

Рисунок 11.6 – Попытка соединения с ftp-сервером

Как видим, попытка не увенчалась успехом. Если же настроить инспектирование FTP-трафика, доступ к ftp-серверу будет возможен. Иллюстрировать здесь это не будем, так как Cisco Packet Tracer в силу своей ограниченной функциональности это не поддерживает. Поэтому

инспектирование разных видов трафика необходимо производить либо на реальном оборудовании, либо с использованием симулятора GNS3.

Однако Cisco Packet Tracer поддерживает инспектирование TCP-трафика. Так как и протокол HTTP, и протокол FTP использует для передачи TCP-сегменты, после настройки TCP-инспектирования доступны окажутся и http и ftp серверы (рисунки 11.7, 11.8).

```
Router(config)#ip inspect name HTTP tcp
Router(config)#
Router(config)#^Z
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

Рисунок 11.7 – Настройка инспектирования TCP-трафика

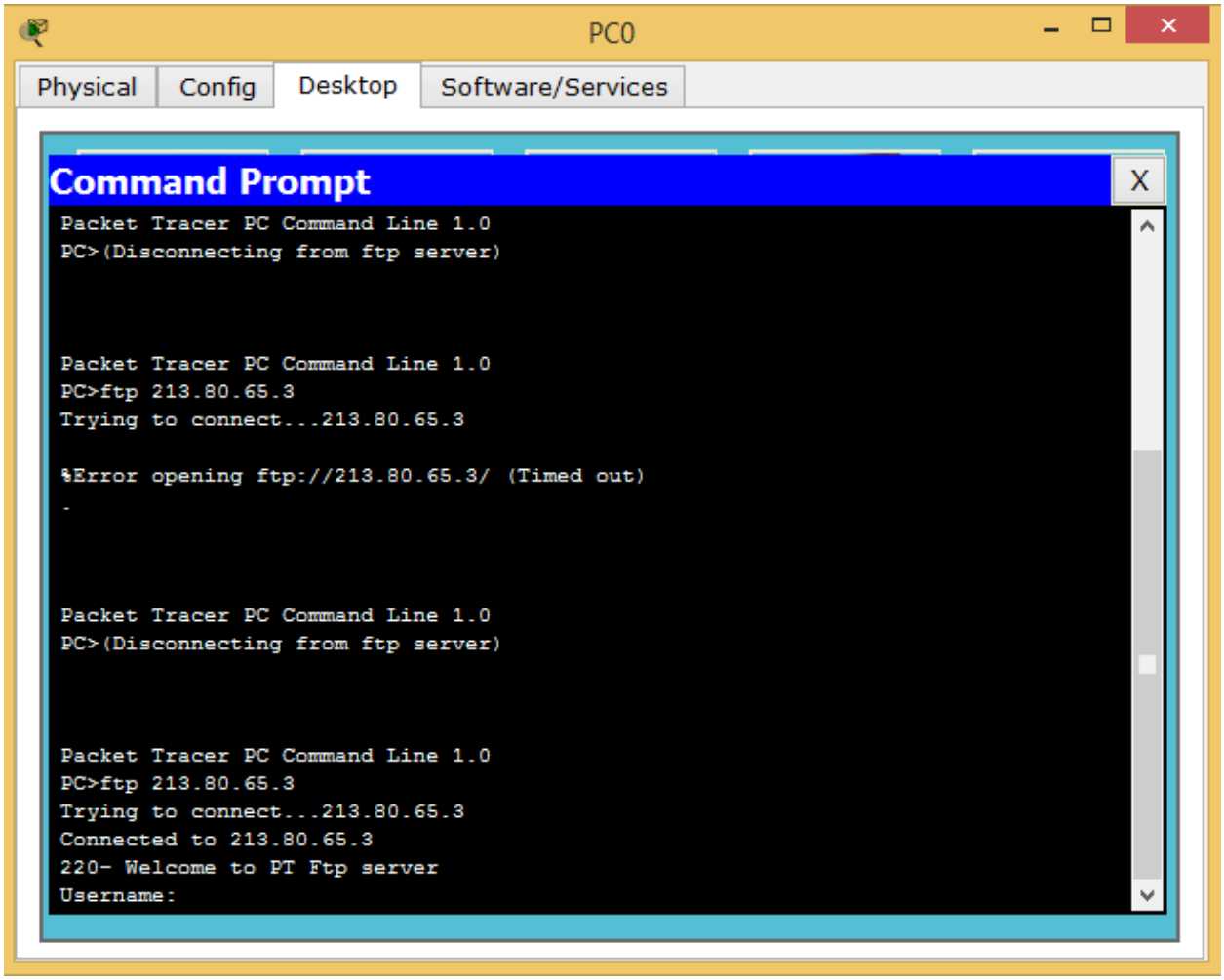

Рисунок 11.8 – Доступ к ftp-серверу после настройки инспектирования TCP-трафика

Рассмотренные настройка межсетевого экрана являются базовыми, а более тонкие настройки применяются в случае, если производится разделение сети с различными зонами безопасности, которые рассмотрим на следующем занятии.

11.5 Отчет по работе:

Демонстрация корректной работы межсетевого экрана.

Практическое занятие №12

Конфигурирование Zone-Based Policy Firewall

12.1 Цель работы: Привитие навыков конфигурирования межсетевых экранов Zone-Based Policy Firewall.

12.2 Перечень оборудования:

- Стенд «Инфокоммуникационные сети»;
- Рабочие станции под управлением ОС Windows пакетом Cisco Packet Tracer или GNS3.

12.3 Задание:

- Собрать сеть, показанную на рисунке 12.1, настроить адресацию по следующим исходным данным:
  - внутренняя сеть – 192.168.X.0;
  - демилитаризованная зона – 192.168.X+10.0;
  - внешняя сеть – 213.80.X.0.
- Проверить связность сети;
- Сконфигурировать на маршрутизаторе межсетевой экран ZBFW, обеспечивающий выполнение следующих правил:
  - из внутренней сети разрешены все запросы к внешней сети;
  - к демилитаризованной зоне разрешены запросы из внутренней сети по протоколам Telnet и SSH и только с адресов, принадлежащих внутренней сети;
  - из внешней сети запросы во внутреннюю сеть запрещены;
  - из внешней сети запросы в демилитаризованную зону разрешены только по протоколу HTTP.

12.4 Порядок выполнения работы

Zone-Based Policy Firewall (ZBFW) – относительно новое направление на маршрутизаторах под управлением операционной системы Cisco IOS для конфигурирования правил доступа между сетями. До появления этой

технологии трафик фильтровался с помощью списков доступа ACL (рассмотренных раньше) и динамической инспекции трафика Context-Based Access Control (CBAC) (в настоящем пособии не рассматривается). И ACL и правила CBAC применяются непосредственно на физические интерфейсы, что во многих случаях не способствует масштабируемости и гибкости сетевых решений. Такая модель ограничивает степень детализации политик межсетевого экрана и вызывает путаницу правильного применения политики межсетевого экранирования, особенно в случаях, когда политика межсетевого экрана должна применяться между несколькими интерфейсами. Zone-Based Policy Firewall меняет конфигурацию межсетевого экрана от старой интерфейсной модели на более гибкую модель, основанную на зонах безопасности.

Зона безопасности состоит из набора различных интерфейсов, которые должны иметь одинаковую политику сетевой безопасности или, иначе говоря, одинаковый уровень доверия. В каждую из зон может входить один или несколько интерфейсов. После создания зон настраиваются правила для взаимодействий между зонами. Такой подход облегчает настройки правил межсетевого экрана, так как правила определяются не для отдельных интерфейсов, а для множества интерфейсов, входящих в одну зону. Кроме того в Zone-Based Policy Firewall используется язык Cisco Policy Language (CPL), который позволяет более гибко, чем в предыдущих версиях межсетевого экрана, настраивать правила фильтрации трафика.

В большинстве случаев сеть делится, как минимум, на три зоны:

- внутренняя зона, где расположены пользователи (inside);

- внешняя зона (Интернет – outside);

- демилитаризированная зона, где расположены серверы, к которым должен быть обеспечен доступ извне (dmz).

Важно, что по умолчанию весь трафик между различными зонами будет запрещен, весь трафик внутри зоны – разрешен.

На первом этапе настройки межсетевого экрана необходимо создать зоны. Зоны создаются командой, выполняемой в режиме глобального конфигурирования:

zone security <имя зоны>

После этого необходимо создать пары зон, между которыми будет передаваться трафик:

zone-pair security <имя пары> source <имя зоны> destination <имя зоны>

Необходимо отметить, что пары зон являются однонаправленными. То есть, если в нашей сети предполагается двунаправленная передача данных между внутренней и внешней зонами, необходимо создать две пары зон. Например (считаем, что внутренней зоне было присвоено имя IN, а внешней – OUT, имя пар IN_OUT и OUT_IN):

zone-pair security IN_OUT source IN destination OUT

zone-pair security OUT_IN source OUT destination IN

Как указывалось выше, по умолчанию передача трафика между созданными парами зон запрещена. Для формирования разрешенного для передачи трафика необходимо определить критерии, по которым отсортировывается нужный трафик. Для этого используется так называемый class-map (дословно – классовая карта). Class-map определяет, какой именно трафик будет инспектироваться (проходить между зонами, также проходить будут ответы на этот трафик). Фильтроваться трафик может по критериям с 3-го по 7-й уровней модели OSI (т.е. начиная от IP-адреса и заканчивая трафиком определенного приложения или сервиса прикладного уровня). Определяться трафик может списками доступа, значением CoS, типом протокола и еще рядом других параметров. Критериев может присутствовать одновременно несколько. При этом можно указать, должен ли трафик попадать под все эти критерии (match-all) или под любой из них (match-any). Таким образом, основная задача class-map – отфильтровать необходимый тип трафика.

Для создания class-map используется команда:

class-map type inspect match-all/match-any <имя class-map>

После этого мы попадаем в конфигурирование созданного class-map, и далее необходимо использовать команду **match** с указанием критериев, по которым отсортировывается трафик:

- **access-group** - стандартный, расширенный или именованный список доступа, который может фильтровать трафик на основании IP адреса и порта источника и приемника. Это единственный способ выделить трафик от конкретного источника к конкретному получателю;

- **protocol** - это протоколы уровня 4 (TCP, UDP, ICMP), а также прикладные сервисы, такие как HTTP, SMTP, DNS, и т.д. Может быть указан любой известный или определяемый пользователем сервис;

- **class-map** - подчиненный класс, который предоставляет дополнительные критерии соответствия;

- **not** - определяет, что любой трафик, который не соответствует указанному сервису или протоколу, или листу доступа, будет выбран в данном class-map.

Важно, что критерии вводятся списком, и порядок обработки списка последовательный, как и у списков доступа. Например, если при конфигурировании class-map match-any мы использовали команды

match protocol http

match protocol tcp

то при обработке пакета сначала будет проверено его соответствие протоколу HTTP. Если найдено соответствие, то далее будет инспектироваться этот трафик, и следующее условие не будет проверяться. Если же команды поменять местами, то пакет сначала попадет под инспектирование трафика TCP.

Политики межсетевого экранирования определяются командой **policy-map**. Команда **policy-map** определяет действие, которое будет произведено с отфильтрованным с помощью команды **class-map** трафиком. Существует три основных действия, которые применимы к классифицированному трафику:

Drop – Трафик, обрабатываемый этим действием, отбрасывается и никакого уведомления на удаленный хост не высылается (в противоположность классическим листам доступа (ACL), когда высылается ICMP-сообщение Host Unreachable). Каждая карта политик имеет скрытый класс class-default, для которого сконфигурировано действие **Drop** (аналогично строке deny any any в любом списке доступа).

Pass – Пропускает трафик, не включая инспекцию протокола. Это действие позволяет маршрутизатору пересылать трафик из одной зоны в другую, при этом он не отслеживает состояние соединений или сессий. Это действие разрешает прохождение трафика только в одном направлении. Чтобы обратный трафик был передан, должна быть соответствующая политика и для него. Это действие полезно для таких протоколов, как IPSec ESP, IPSec AH, ISAKMP и других по своей сути безопасных протоколов с предсказуемым поведением.

Inspect - Включает динамическую инспекцию для трафика, который проходит от зоны источника к зоне приемника, и автоматически разрешает обратный трафик даже для сложных протоколов, таких как H.323. Например, если трафик передается из зоны IN в зону OUT, маршрутизатор поддерживает информацию о соединениях или сеансах для TCP и UDP трафика. Поэтому маршрутизатор разрешает обратный трафик из зоны OUT в зону IN в качестве ответов на запросы соединений из IN в OUT.

Формат команды:

policy-map type inspect <имя policy-map >

После этого мы попадаем в режим конфигурирования созданного policy-map, в котором указываем, какой именно class-map должен обрабатываться, и затем указываем необходимое действие:

class type inspect <имя class-map>

inspect/pass/drop

Теперь созданные политики необходимо применить к парам зон, которые уже были созданы ранее (пары зон можно создать и на этом шаге):

zone-pair security <имя пары> source <имя зоны> destination <имя зоны>

service-policy type inspect <имя policy-map >

Осталось в явном виде указать маршрутизатору, какие его интерфейсы относятся к какой зоне:

interface <имя интерфейса>

zone-member security <имя зоны>

Приведем пример конфигурирования межсетевого экранирования на примере сети, показанной на рисунке 12.1.

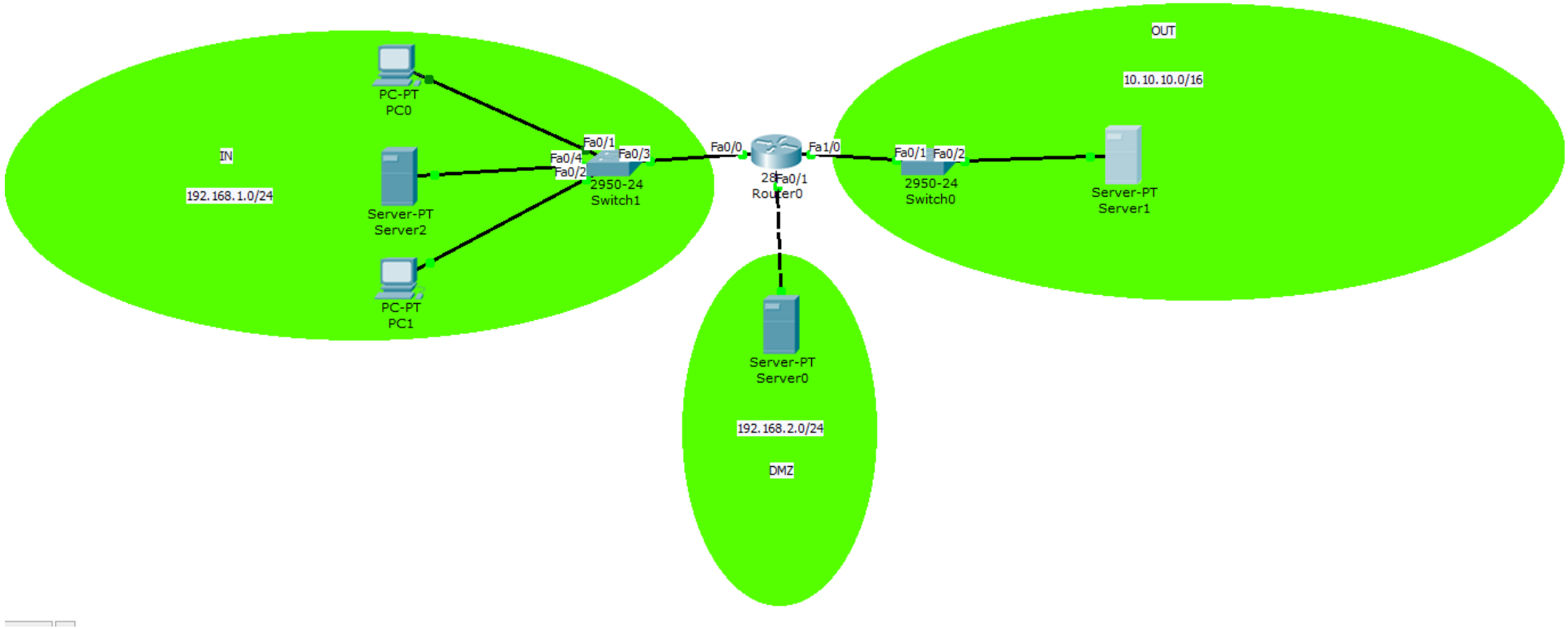

Рисунок 12.1 – Пример сети

Для простоты будем считать, что интерфейсам маршрутизатора присвоены первые адреса из адресного пространства подсетей (192.168.1.1/24 в подсети IN, 10.10.10.1/16 в подсети OUT, и т.д.)

Сначала необходимо определить зоны:

R0(config)#zone security IN

R0(config)#zone security OUT

R0(config)#zone security DMZ

Так как созданные зоны пока не привязаны ни к каким интерфейсам, никакого ограничения в передаче трафика после создания зон не произойдет.

Сначала сконфигурируем межсетевой экран для информационного обмена между зонами IN и OUT. Считаем, что из внутренней сети разрешены любые запросы к серверу, находящемуся во внешней сети. Таким образом, при создании class-map в этом направлении удобно использовать стандартный список доступа, разрешающий передавать данные от всех рабочих станций подсети 192.168.1.0/24:

R0(config)#access-list permit 10 192.168.1.0 0.0.0.255

Создаем class-map для трафика, удовлетворяющего условию списка доступа 10, присвоив самому class-map имя FROM-IN:

R0(config)#class-map type inspect match-any FROM-IN

и указываем, какой трафик входит в созданный class-map:

R0(config-cmap)#match access-group 10

Создаем policy-map с именем FROM-INSIDE (в принципе, имена class-map и policy-map могут совпадать, здесь специально выбраны разные имена):

R0(config)#policy-map type inspect FROM-INSIDE

указываем, какой class-map должна обрабатывать политика:

R0(config-pmap)#class type inspect FROM-IN

и указываем нужное действие – инспектировать:

R0(config-pmap-c)#inspect

Теперь обращаемся к интерфейсам для указания, к каким зонам они относятся. Одновременно можно назначить интерфейсам IP-адреса и включить их, если этого не было сделано раньше:

R0(config)#int fa0/0

R0(config-if)#ip address 192.168.1.1 255.255.255.0

R0(config-if)#no shutdown

R0(config-if)#zone-member security IN

R0(config)#int fa1/0

R0(config-if)#ip address 10.10.10.1 255.255.0.0

R0(config-if)#no shutdown

R0(config-if)#zone-member security OUT

Так как интерфейсы маршрутизатора теперь принадлежат к разным зонам, передача трафика между ними запрещена. Для разрешения передачи между ними трафика в соответствии с созданной политикой создадим зонную пару с именем IN-TO-OUT:

R0(config)#zone-pair security IN-TO-OUT source IN destination OUT

и применим к ней созданную политику:

R0(config-sec-zone-pair)#service-policy type inspect FROM-INSIDE

Заметим, что пару для обратного направления OUT-IN мы не создавали. Это связано с тем, что по условиям задачи любой трафик извне запрещен, за исключением ответов на запросы, которые поступили из внутренней сети (они инспектируются). Проверить работоспособность сконфигурированного межсетевого экрана достаточно просто – если из внутренней сети послать любой запрос (например, ping или http-запрос), то внутренний компьютер должен получить ответ (рисунок 12.2). Если же послать запрос с внешнего сервера, ответа не будет – запрос будет отброшен маршрутизатором (рисунок 12.3).

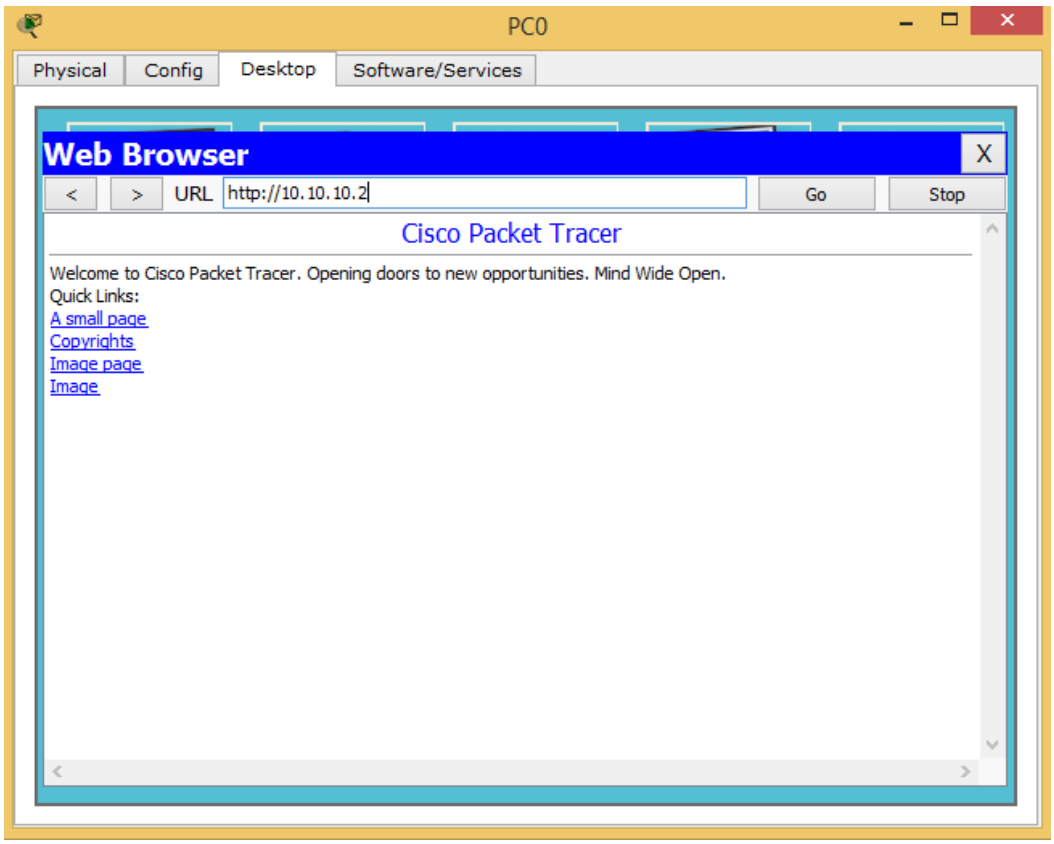

Рисунок 12.2 – Получение ответа при запросе из внутренней сети

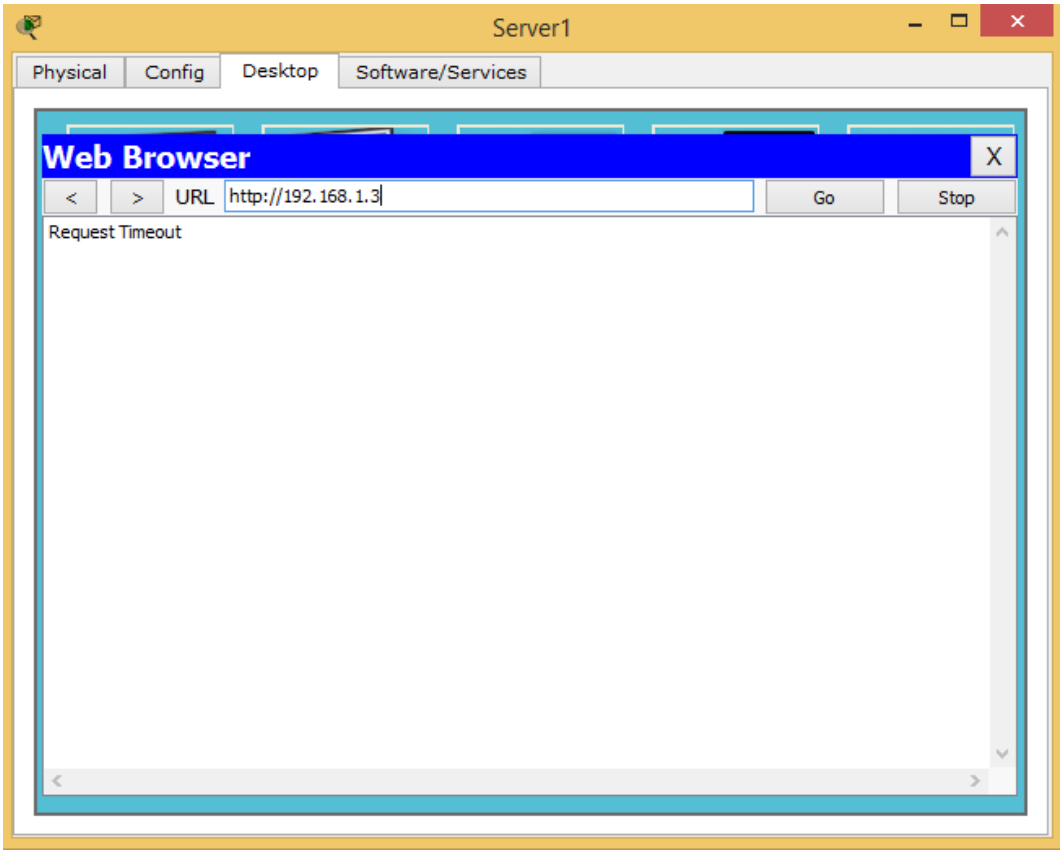

Рисунок 12.3 – Отсутствие ответа при запросе из внешней сети

Перейдем теперь к настройке демилитаризованной зоны. Здесь ситуация несколько иная – к серверам демилитаризированной должен быть обеспечен доступ как извне (например, по протоколу HTTP, если это web-сервер), так и изнутри (например, по протоколу SSH или Telnet для конфигурирования сервера). Соответственно, в этом случае необходимо создавать как минимум две пары зон – OUT-DMZ, IN-DMZ – с различными политиками. Если же предполагается наличие доступа из демилитаризованной зоны, необходимо создавать пары DMZ-OUT или DMZ-IN.

Предположим, что для нашей сети необходимо обеспечить следующие условия:

1. Доступ из внешней сети к серверу, находящемуся в DMZ, возможен только по протоколу HTTP;

2. Доступ из внутренней сети к серверу, находящемуся в DMZ, возможен по протоколам SSH, Telnet, HTTP, и только с тех IP-адресов, которые относятся к адресному пространству внутренней сети 192.168.1.0/24.

Привяжем интерфейс маршрутизатора fa0/1 к созданной ранее зоне DMZ, одновременно назначим ему IP-адрес и включим его:

R0(config)#int fa0/1

R0(config-if)#ip address 192.168.2.1 255.255.255.0

R0(config-if)#no shutdown

R0(config-if)#zone-member security DMZ

Очевидно, что для передачи трафика из внутренней сети к серверу DMZ необходимо создать несколько class-map, так как в каждом из них одновременно должно выполняться как минимум два условия:

- источником трафика является внутренняя сеть и используется протокол SSH;

- источником трафика является внутренняя сеть и используется протокол Telnet;

- источником трафика является внутренняя сеть и используется протокол HTTP.

Создадим class-map для передачи трафика из внутренней сети к серверу DMZ по протоколу SSH с именем IN-DMZ-SSH:

R0(config)#class-map type inspect match-all IN-DMZ-SSH

R0(config-cmap)#match access-group 10

R0(config-cmap)#match protocol ssh

Создадим class-map для передачи трафика из внутренней сети к серверу DMZ по протоколу Telnet с именем IN-DMZ-TLN:

R0(config)#class-map type inspect match-all IN-DMZ-TLN

R0(config-cmap)#match access-group 10

R0(config-cmap)#match protocol telnet

Создадим class-map для передачи трафика из внутренней сети к серверу DMZ по протоколу HTTP с именем IN-DMZ-HTTP:

R0(config)#class-map type inspect match-all IN-DMZ-HTTP

R0(config-cmap)#match access-group 10

R0(config-cmap)#match protocol http

Создадим policy-map с именем IN-DMZ, в которой укажем на необходимость инспектирования трафика, удовлетворяющего созданным class-map:

R0(config)#policy-map type inspect IN-DMZ

R0(config-pmap)#class type inspect IN-DMZ-SSH

R0(config-pmap-c)#inspect

R0(config-pmap-c)#exit

R0(config-pmap)#class type inspect IN-DMZ-TLN

R0(config-pmap-c)#inspect

R0(config-pmap-c)#exit

R0(config-pmap)#class type inspect IN-DMZ-HTTP

R0(config-pmap-c)#inspect

R0(config-pmap-c)#exit

R0(config-pmap)#

Для доступа к серверу DMZ из внешней сети должен использоваться только протокол HTTP, поэтому создадим один class-map с именем OUT-DMZ:

R0(config)#class-map type inspect match-any OUT-DMZ

R0(config-cmap)#match protocol http

Создадим policy-map с таким же названием и с указанием инспектировать трафик:

R0(config)#policy-map type inspect OUT-DMZ

R0(config-pmap)#class type inspect OUT-DMZ

R0(config-pmap-c)#inspect

Осталось создать пары зон и применить к ним созданные политики. Пара IN-TO-DMZ:

R0(config)#zone-pair security IN-TO-DMZ source IN destination DMZ

R0(config-sec-zone-pair)#service-policy type inspect IN-DMZ

Пара зон OUT-TO-DMZ:

R0(config)#zone-pair security OUT-TO-DMZ source OUT destination DMZ

R0(config-sec-zone-pair)#service-policy type inspect OUT-DMZ

12.5 Отчет по работе:

Демонстрация корректной работы межсетевого экрана

Перечень использованных источников

1. Манин А.А., Сосновский И.А. Системы коммутации. Принципы и технологии пакетной коммутации. Учебное пособие. Издание 2-е, переработанное и дополненное. Ростов-на-Дону: СКФ МТУСИ, 2018.

2. Ожиганов А.А. Криптография. Учебное пособие. СПб: Университет ИТМО, 2016.

3. https://www.intuit.ru/studies/courses/2/2/lecture/50

4. ttps://www.gns3.com/

5. https://www.wireshark.org/

6. https://tacacsgui.com/

7. Малюха В.А., Новопашенный А.Г., Подгурский Ю.Е., Заборовский В.С. Методы и средства защиты компьютерной информации. Межсетевое экранирование: Учебное пособие. СПб: Изд-во СПбГПУ, 2010.