**Министерство образования и науки Российской Федерации   
Федеральное государственное бюджетное образовательное учреждение высшего образования   
«Восточно-Сибирский государственный университет технологий и управления»**

Алтаев А.А., Будаев Е.С.

Практикум по дисциплине   
«Компьютерные сети. Курсовая работа»

**МЕТОДИЧЕСКОЕ ПОСОБИЕ**

**для студентов направления подготовки   
09.03.04 - Программная инженерия**

Издательство ВСГУТУ

Улан-Удэ, 2022

УДК 004.7(075)

ББК 32.971.35

Практикум по дисциплине  
«Компьютерные сети. Курсовая работа»/ Сост. Алтаев А.А., Будаев Е.С. – Улан-Удэ, Изд-во ВСГУТУ, 2022. – 109 с.

Методическое пособие включает краткую теорию компьютерных сетей, задание по курсовой работе и указания по ее выполнению. Предназначено студентам направления подготовки 09.03.04 - Программная инженерия

Рецензент: *Дармаев Т.Г.*, к.ф.-м.н., доц. БГУ

Печатается по решению редакционно-издательского совета ВСГУТУ

 ВСГТУ, 2022 г.

 Алтаев А.А., Будаев Е.С.

# Оглавление

[Оглавление 3](#_Toc93554452)

[Введение 5](#_Toc93554453)

[Общий обзор теории компьютерных сетей 6](#_Toc93554454)

[**Базовые понятия** **6**](#_Toc93554455)

[**Классификация компьютерных сетей** **7**](#_Toc93554456)

[Классификация компьютерных сетей по территории, по размерам 7](#_Toc93554457)

[Классификация компьютерных сетей по топологии 8](#_Toc93554458)

[Классификация компьютерных сетей по типу среды передачи 9](#_Toc93554459)

[**Структура сети** **9**](#_Toc93554460)

[**Сетевые протоколы** **10**](#_Toc93554461)

[Совместная работа сетевых протоколов 13](#_Toc93554462)

[Инкапсуляция и декапсуляция 14](#_Toc93554463)

[**Транспортный уровень модели OSI** **15**](#_Toc93554464)

[Порты 16](#_Toc93554465)

[Протокол TCP 16](#_Toc93554466)

[Протокол UDP 19](#_Toc93554467)

[**Сетевой уровень модели OSI** **21**](#_Toc93554468)

[Связь сетевого уровня с транспортным 21](#_Toc93554469)

[Типы рассылок 23](#_Toc93554470)

[Типы IP адресов 23](#_Toc93554471)

[Двоичная и десятичная системы исчисления 24](#_Toc93554472)

[Структура IP адреса 25](#_Toc93554473)

[Методика вычисления сетевого и широковещательного адресов 26](#_Toc93554474)

[Примеры расчета адресов 26](#_Toc93554475)

[Экскурс в историю IP протокола 28](#_Toc93554476)

[Классовая адресация 28](#_Toc93554477)

[Служебные адреса 30](#_Toc93554478)

[Бесклассовая адресация 31](#_Toc93554479)

[Subnetting 32](#_Toc93554480)

[Supernetting 33](#_Toc93554481)

[Публичные и приватные адреса, технология NAT 35](#_Toc93554482)

[Выполнение контрольного теста по IP адресации 37](#_Toc93554483)

[**DNS** **37**](#_Toc93554484)

[**Сетевые устройства** **40**](#_Toc93554485)

[**Канальный уровень модели OSI** **44**](#_Toc93554486)

[Предназначение канального уровня 44](#_Toc93554487)

[Протоколы канального уровня 45](#_Toc93554488)

[Frame Relay и MPLS 47](#_Toc93554489)

[Логика работы Ethernet коммутатора 49](#_Toc93554490)

[Обнаружение и исправление ошибок 52](#_Toc93554491)

[**Физический уровень** **53**](#_Toc93554492)

[Предназначение физического уровня 53](#_Toc93554493)

[Типы обмена трафиком 55](#_Toc93554494)

[Классификация сред передачи данных 56](#_Toc93554495)

[Сетевые устройства Cisco 61](#_Toc93554496)

[Моделирование компьютерных сетей 63](#_Toc93554497)

[**Базовая настройка устройств Cisco** **64**](#_Toc93554498)

[Маршрутизатор 64](#_Toc93554499)

[Коммутатор 66](#_Toc93554500)

[**Маршрутизация** **67**](#_Toc93554501)

[Таблицы маршрутизации 68](#_Toc93554502)

[Маршрут по умолчанию 72](#_Toc93554503)

[Динамическая маршрутизация 76](#_Toc93554504)

[Задание 77](#_Toc93554505)

[Пример выполнения задания 83](#_Toc93554506)

[**Построение модели сети** **83**](#_Toc93554507)

[**Планирование IP адресации сети** **83**](#_Toc93554508)

[Приватная сеть 83](#_Toc93554509)

[Публичная сеть 85](#_Toc93554510)

[**Базовая настройка маршрутизаторов и коммутаторов и реализация IP адресации** **86**](#_Toc93554511)

[**DHCP** **88**](#_Toc93554512)

[**Реализация динамической маршрутизации** **90**](#_Toc93554513)

[RIP 90](#_Toc93554514)

[OSPF 92](#_Toc93554515)

[**Настройка NAT на пограничном маршрутизаторе** **95**](#_Toc93554516)

[**Настройка DNS серверов** **98**](#_Toc93554517)

[Обеспечение доступности публичного сайта 98](#_Toc93554518)

[Обеспечение доступности корпоративного сайта извне 101](#_Toc93554519)

[Обеспечение доступности корпоративного сайта внутренним хостам по URL адресу 102](#_Toc93554520)

[Оформление отчета 105](#_Toc93554521)

[Литература 106](#_Toc93554522)

[Web ресурсы 106](#_Toc93554523)

[Приложение 1 109](#_Toc93554524)

# Введение

В рамках дисциплины «Компьютерные сети. Курсовая работа», читаемой студентам направления подготовки 09.03.04 - Программная инженерия, предусмотрено выполнение индивидуального задания «Настройка сети». Данное пособие содержит краткие сведения из теории компьютерных сетей, варианты задания и пример его выполнения. В ходе выполнения задания студенты получают навыки по сборке сети, базовой настройке сетевых устройств, планированию и реализации IP адресации, настройке DHCP и DNS серверов, реализации динамической маршрутизации, технологии NAT, фильтрации нежелательного трафика с применением ACL.

# 

# Общий обзор теории компьютерных сетей

## **Базовые понятия**

**Компьютерная сеть** — это совокупность компьютеров, которые могут обмениваться между собой информацией.

**Компоненты компьютерной сети:**

* компьютер;
* линии связи и коммуникационное оборудование, реализующие возможность обмена информацией;
* операционная система, в частности, модули ОС, реализующие сетевое взаимодействие;
* распределенные приложения - программы, работающие одновременно на разных компьютерах и взаимодействующие между собой через сеть.

**Сервер** - компьютер или программа, предоставляющая некоторые услуги.

**Выделенный сервер** — это компьютер, служащий только для обслуживания клиентских машин. Доступ к нему обычно получают администраторы системы для выполнения задач управления, мониторинга и поддержки работоспособности. Часто они выполняют только какую-то одну задачу.

**Клиент** — это компьютер или программа, запрашивающая услуги.

**Протокол** — это правила, позволяющие компьютерам осуществлять через сеть обмен данными. Сетевой протокол определяет:

* формат сообщений;
* очередность сообщений;
* действия, которые необходимо выполнять при получении, приеме сообщений или при наступлении иных событий.

**Пропускная способность канала связи** (англ. bandwidth) — наибольшая скорость передачи информации по каналу связи. Измеряется числом передаваемых двоичных символов в 1 секунду. Скорость передачи зависит от физических свойств канала связи, статистических свойств помех, способа передачи, приема сигналов и др.

Примеры:

* FastEthernet - 100 Mbps
* WiFi 802.11ac - 600 Mbps
* Gigabit Ethernet - 1000 Mbps

Особенности терминологии

* **Internet** – система объединенных компьютерных сетей, образующая глобальную сеть
* **internet** - несколько локальных сетей (сетевой комплекс)
* **intranet** – частная сеть организации, использующая механизмы Интернета
* **extranet** – объединение intranet-сетей различных компаний через Интернет
* **Ethernet** – технология локальных сетей

Интранет (англ. Intranet) — в отличие от сети Интернет, это внутренняя частная ceть организации. Как правило, Интранет — это Интернет в миниатюре, который построен на использовании протокола IP для обмена и совместного использования некоторой части информации внутри этой организации. Intranet допускает использование публичных каналов связи, входящих в Internet, (VPN — Virtual Private Network), но при этом обеспечивается защита передаваемых данных и меры по пресечению проникновения извне на корпоративные узлы.

## **Классификация компьютерных сетей**

### Классификация компьютерных сетей по территории, по размерам

* Local Area Network (LAN) — сети одной квартиры, дома, организации.
* Metropolian Area Network (MAN), городские — высокоскоростные каналы связи в пределах большого города.
* Региональные — объединяют компьютеры географической области.
* Wide Area Network[[1]](#footnote-2) (WAN), глобальные.

### Классификация компьютерных сетей по топологии

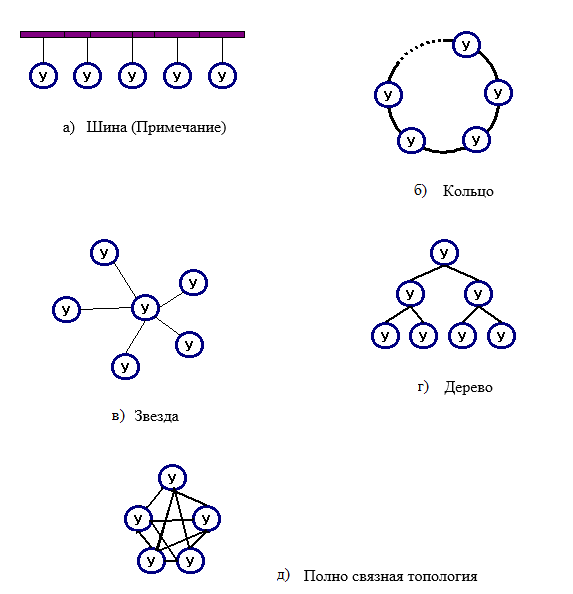
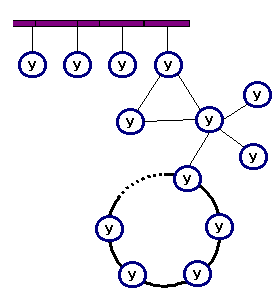


Рис. . Топологии сетей.

 e) Смешанные топологии

Продолжение Рис. 1. Топологии сетей.

### Классификация компьютерных сетей по типу среды передачи

**Проводные** (wire):

* витая пара;
* коаксиальный кабель;
* serial кабель;
* оптоволокно.

**Беспроводные** (wireless):

* радиосвязь (WiFi, WiMAX);
* спутниковый Интернет;
* СВЧ-связь (Bluetooth);
* инфракрасная связь.

## **Структура сети**

ISP (Internet Service Provider - Поставщик Интернет-услуг) – это компании, предоставляющие своим клиентам не только подключение к Интернету, но и другие сетевые услуги, такие как доступ к электронной почте, услуги центров обработки данных.

По степени охвата обслуживаемой территории ISP делятся на локальные, региональные и магистральные.

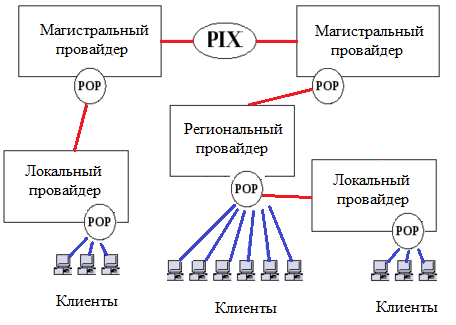


Рис. . Типы провайдеров

Для обслуживания своих клиентов (как физических, так и юридических) провайдеры используют оборудование POP (Point of Presence – Точка присутствия). Между собой провайдеры связаны через PIX (Point Internet eXchange - Точка обмена трафиком). Сами PIX связаны между собой магистральными каналами связи (backbone), представляющими из себя волоконно-оптические кабели.

Сети магистральных провайдеров (провайдеры первого уровня Tier-1) образуют основу Интернета, охватывая весь мир. А к их сетям через POP присоеденены сети региональных и локальных провайдеров (рис. 2).

## **Сетевые протоколы**

Под сетевым протоколом понимается некоторый свод общих правил, которых должны придерживаться все сетевые устройства и программы. Это гарантирует, что сообщение, отправленное в сеть одним устройством, будет получено и правильно обработано другим устройством. На начальном этапе эпохи компьютеризации в США научные лаборатории университетов и компаний в своих ранних локальных сетях применяли собственные протоколы. Когда министерство обороны США (U.S. Department of Defense, USDOD) поставило задачу объединения всех разрозненных локальных сетей в одну общую сеть, то выяснилось, что межсетевой трафик невозможен, так как в разных сетях применялись разные протоколы. Стало очевидным, что сетевые стандарты должны быть общими как для производителей оборудования, так и для разработчиков

Таблица 1. Модель OSI

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Номер (Number) | Уровень  (Layer) | Адреса  (Addresses) | Сообщения (Messages) | Протоколы  (Protocols) | Сетевые устройства (Network devices) |
| 7 | Уровень приложений  (Application layer) | Доменное имя (Domain name) | Данные (Data) | HTTP, FTP, SMTP, POP3, Telnet и др. |  |
| 6 | Уровень представления (Presentation layer) |  | MIME, JPEG, AVI, ZIP и др. |  |
| 5 | Уровень сессии (Session layer) |  |  |  |
| 4 | Транспортный уровень (Transport layer) | Порт (Port) | Сегмент (Segment) | UDP, TCP |  |
| 3 | Сетевой уровень (Network layer) | IPv4, IPv6 | Пакет (Packet) | IPv4, IPv6 | Маршрутизатор (Router), коммутатор 3-го уровня (Switch L3) |
| 2 | Канальный уровень  (Data link layer) | MAC, DLCI и др. | Кадр, фрейм (Frame) | , Wi-Fi,Ethernet, PPP, HDLC Frame Relay и др. | Коммутатор (Switch L2), мост (Bridge) |
| 1 | Физический уровень  (Physical layer) |  | Поток битов (Flow of bits) | Концентратор (Hub), повторитель (Repeater) |

Примечание к таблице: Модель OSI (Open System Interconnection - Взаимодействие открытых систем) была разработана ISO (International Organization for Standardization - Международная организация по стандартизации) [13].

программного обеспечения. Стандарты, выработанные в ходе создания этой сети, легли в основу будущего Интернета. Отметим, что USDOD в то время уделяло большое внимание повышению отказоустойчивости создаваемой сети, так чтобы в гипотетической войне с СССР при обмене ядерными ударами сеть сохраняла бы свою работоспособность.

Часть протоколов являются проприетарными и считаются собственностью компаний, их разработавших. Не проприетарные (открытые) сетевые протоколы размещены на общедоступных серверах в виде документов RFC (Request For Comments). Каждый RFC документ имеет свой уникальный номер. Протоколы включены в состав двух реестров: в эталонную модель OSI (табл. 1) и в стек протоколов TCP/IP

На рисунке 3 сопоставлены оба реестра протоколов

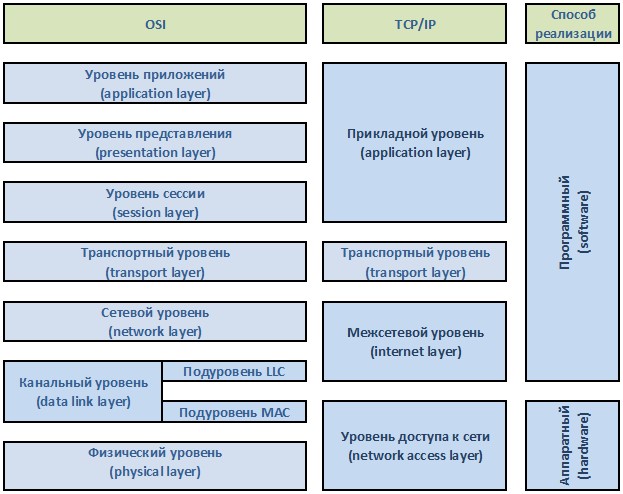


Рис. 3. Модель OSI и стек TCP/IP

Канальный уровень OSI состоит из двух подуровней:

* нижнего MAC подуровня (Media Access Control), регулирующего доступ к физической среде передачи данных;
* верхнего LLC подуровня (Logical Link Control), обеспечивающего обслуживание сетевого уровня.

Алгоритмы протоколов физического уровня и подуровня MAC реализуются в микросхемах (hardware), а подуровня LLC и других уровней (с 3-го по 7-й) - с помощью соответствующего программного обеспечения (software). Например, чип сетевой платы компьютера относится к подуровню MAC, а его драйвер - к подуровню LLC. Следовательно, IP адрес (3-й уровень - software) этой платы может быть легко изменен, а ее MAC адрес (подуровень MAC канального уровня - hardware) «вшит» в микросхему. По этой причине MAC адрес называется аппаратным или физическим адресом.

Стек TCP/IP состоит из четырех уровней и исторически он появился раньше, чем модель OSI. Сетевые модули операционных систем реализованы в соответствии с данным стеком, но при этом стоит отметить, что вся терминология сетевых технологий тесно увязана с моделью OSI.

### Совместная работа сетевых протоколов

Протоколы вышестоящих уровней пользуются услугами протоколов нижележащих

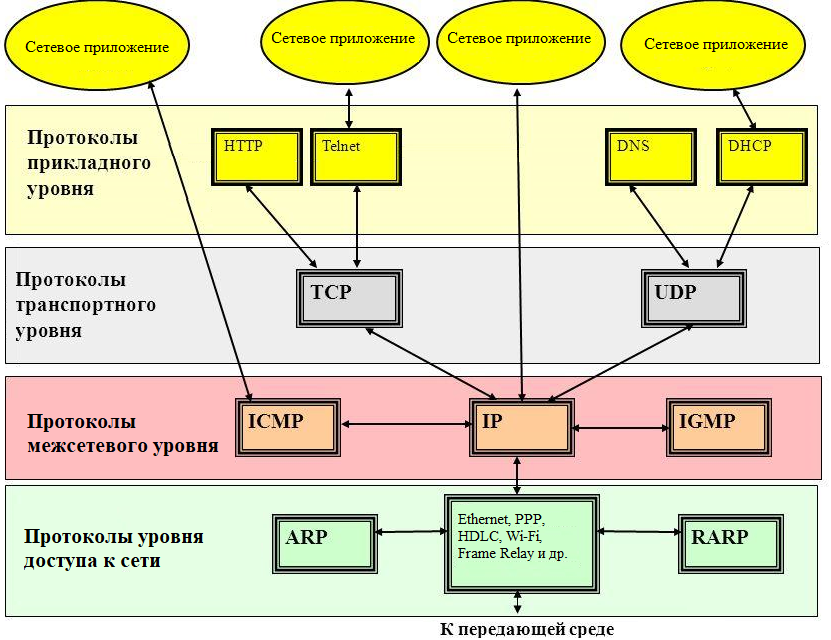


Рис. 4. Взаимодействие сетевых протоколов

уровней. Так, например, HTTP и Telnet для передачи своих данных применяют протокол TCP (рис. 4), DNS и DHCP – протокол UDP. В свою очередь, протоколы TCP и UDP свои данные передают по протоколу межсетевого уровня IP. Протокол IP использует Ethernet, Wi-Fi, PPP, HDLC и другие протоколы уровня доступа к сети.

### Инкапсуляция и декапсуляция

Когда данные перемещаются по уровням сверху вниз, то они оснащаются соответствующими заголовками. На рисунке 5 демонстрируется передача данных между приложениями **Application1** и **Application2**. **Application1** свое сообщение **user data**, адресованное **Application2**, передает 7-му уровню. На этом уровне находится множество протоколов. К примеру, если **Application1** является браузером, то он является HTTP

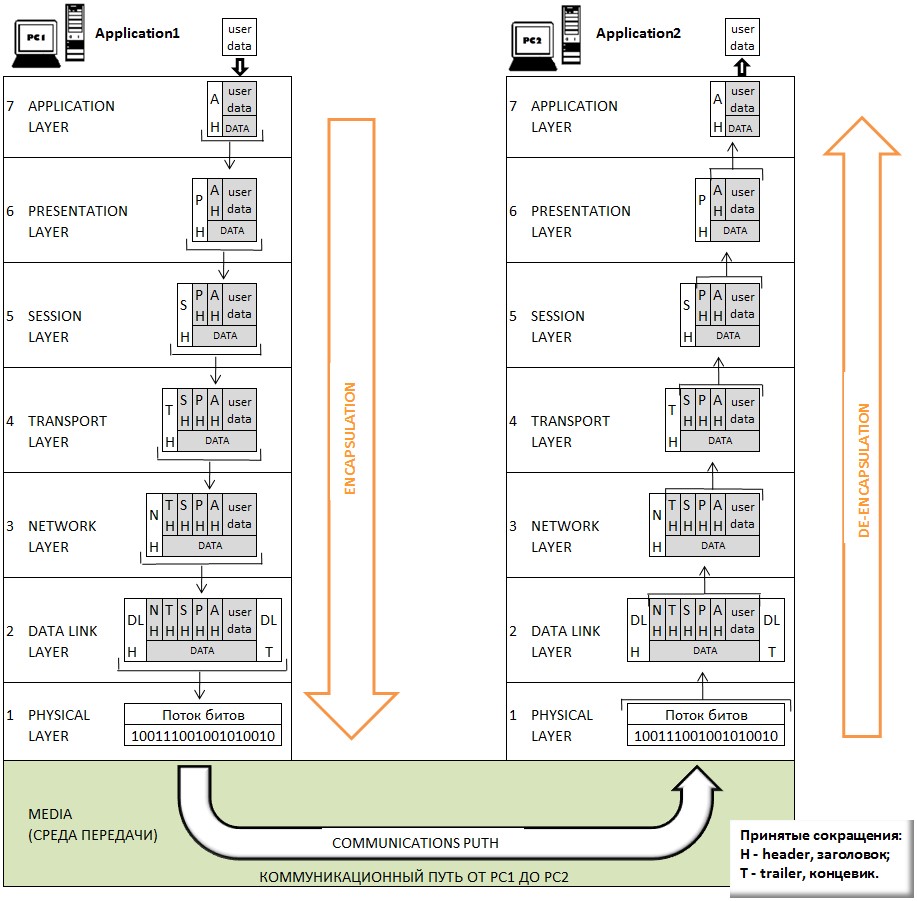


Рис. 5. Инкапсуляция и декапсуляция данных

клиентом. В таком случае сообщение **user data** оснащается HTTP заголовком. Так как HTTP в качестве своего транспорта использует протокол FTP, то на 4-м уровне к сообщению будет добавлен FTP заголовок. Полученный FTP сегмент будет направлен на 3-й уровень и помещен в IP пакет.

Вся рассмотренная процедура передачи данных с вышележащих уровней на нижележащие с добавлением сопутствующих заголовков получила название инкапсуляции данных (сообщения верхнего уровня вкладываются в сообщения нижнего уровня).

Таким образом, сегмент вкладывается в пакет, пакет - во фрейм. На 2-м уровне помимо заголовка в конец фрейма добавляется трейлер. На 1-м уровне передаваемые по сети данные представлены в виде потока битов.

После доставки сообщения на другой компьютер, оно проделывает обратный путь по OSI уровням. Каждый уровень считывает служебную информацию из своего заголовка для соответствующей обработки, удаляет заголовок и передает сообщение вышележащему уровню. Сам процесс передвижения сообщения на верхние уровни получил название декапсуляции. По завершении декапсуляции сетевое приложение **Application2** получит данные **user data**.

## **Транспортный уровень модели OSI**

Рассмотрение наиболее значимых уровней модели OSI начнем с транспортного. О его важности говорит то обстоятельство, что он присутствует в виде отдельного уровня как в модели OSI, так и в стеке протоколов TCP/IP. На этом уровне используются два протокола TCP и UDP, а в качестве адреса применяется порт. Особенностью уровня является его способность проводить разбиение передаваемого сообщения на более мелкие фрагменты и восстановление на другом компьютере из прибывших фрагментов исходного сообщения. Добавлением заголовка к фрагменту будет получен сегмент. В заголовок вносятся позиции начального и конечного байтов фрагмента сообщения, значения портов получателя и отправителя и другая служебная информация.

Предназначение транспортного уровня: **передача сообщения от одного сетевого приложения другому приложению**. Приложения используют в качестве своих адресов порты. Не может быть так, чтобы на компьютере два приложения использовали одинаковый порт. Следовательно, порты являются сетевыми идентификаторами приложений.

### Порты

Порт представляет собой двухбайтовое число, лежащее в интервале от 0 до 65535. Данный интервал разбит на три диапазона:

1. общеизвестные порты (0 - 1023);
2. зарегистрированные порты (1024 – 49151);
3. динамические порты (49152 – 65535).

Общеизвестные порты предназначены для использования серверными программами. При организации клиент-серверного взаимодействия клиент, как правило, является инициатором установления соединения с сервером, и клиент всегда должен знать, какой порт использует его сервер. Для решения данной проблемы за сетевыми протоколами были закреплены порты первого диапазона [6] (например, за протоколом HTTP - порт 80, за HTTPS - 443, SSH – 22, Telnet - 23). Например, браузер, являясь HTTP клиентом, всегда пытается отправить свой запрос на установление соединения с web сервером на 80-й порт. Сервера же, работая в фоновом режиме на своих компьютерах, должны прослушивать свои порты в ожидании приема входящих запросов от клиентов.

Порты второго и третьего диапазонов используются как серверами, так и клиентами. Когда клиент обращается на удаленный сервер, то он от сетевого модуля операционной системы своей машины получает любой свободный на данный момент порт из диапазона выше, чем 1023. И этот выделенный клиенту порт будет занят на период всей сессии связи с сервером и по ее завершении будет вновь возвращен в список свободных портов и может быть повторно использован в сессиях других сетевых приложений. Отличие зарегистрированных портов от динамических состоит в том, что порты второго диапазона могут быть закреплены за серверами на постоянной основе. В таком случае исключается возможность конфликта серверов разных приложений за право использования одного и того же порта. Регистрацией портов занимается организация IANA (Internet Assigned Numbers Authority - Администрация адресного пространства Интернет).

### Протокол TCP

Основное достоинство протокола TCP (Transmission Control Protocol – Протокол управления передачей) – надежность доставки сообщения. Само сообщение прибывает на узел назначения в виде фрагментов. TCP в случае отсутствия среди прибывших какого-либо фрагмента отправляет запрос на повторную пересылку недостающего фрагмента, что обеспечивает полную доставку данного сообщения. Алгоритм передачи сообщений протоколом TCP состоит из трех этапов: установления соединения между двумя удаленными приложениями, обмена сообщениями и завершения соединения. При установлении соединения оба приложения обмениваются «трехкратным рукопожатием» (рис. 6).

Первое «рукопожатие» состоит в отправке инициатором соединения (это клиент) SYN запроса. Сервер в случае своей готовности отвечает SYN–ACK сообщением. Установление TCP соединения завершается присылкой клиентом ACK сообщения с данными протокола прикладного уровня (например, HTTP запросом, если клиентом является браузер).

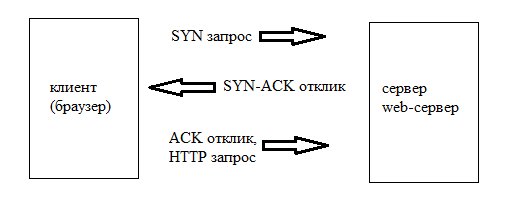


Рис. 6. Установление TCP соединения

На втором этапе сторона, пересылающая свои данные, отправляет фреймы порциями. В примере, рассмотренном на рисунке 7, браузер выгружает свой файл на сервер. Передача идет порциями сегментов. После передачи очередной порции браузер переходит в режим ожидания прихода квитанции от сервера. В квитанции сервер подтверждает получение всей порции и сообщает о своей готовности к приему следующей порции. Объем порции называется скользящим окном. Если при передаче не наблюдаются потери сегментов, то принимающая сторона может уведомить отправителя о возможности увеличения размера данного окна, что приводит к уменьшению числа периодов ожидания и, соответственно, к увеличению скорости передачи данных. При плохой связи размер окна, напротив, может быть уменьшен.

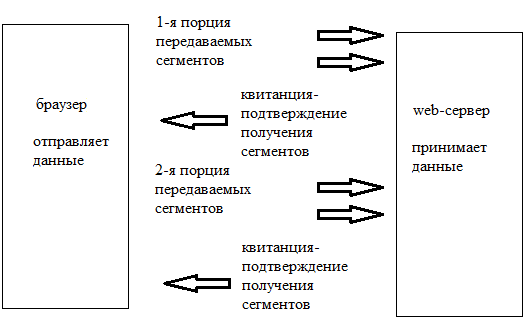


Рис. 7. Передача данных

По завершении передачи оба приложения закрывают соединение, обменявшись FIN и ACK сообщениями (рис. 8). Инициатором разрыва может быть любая сторона.

Таким образом, особенности протокола TCP заключаются в следующем:

1. гарантированность доставки сообщения;
2. установление соединения;
3. применение квитанций;
4. низкая скорость передачи, обусловленная наличием периодов ожидания.

Более подробную информацию об особенностях работы протокола TCP можно узнать на ресурсе [8].

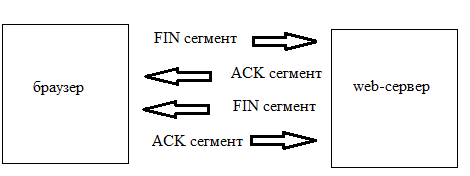


Рис. 8. Завершение TCP соединения

На особенностях установления TCP соединения основана известная DOS атака[[2]](#footnote-3) на сервер. Зараженные хосты отправляют в определенное время SYN запросы на атакуемый сервер. Сервер для каждого подключения создает отдельную нить исполнения и отправляет хосту SYN–ACK ответ и переходит в режим ожидания сообщения от хоста. Но хост свое «третье рукопожатие» не выполняет. У сервера настроено время ожидания, по истечении которого его нить исполнения удаляется из оперативной памяти машины. Так как атака выполняется большим количеством хостов, то у сервера заканчивается лимит на число одновременных подключений, и он начинает отказывать другим хостам в установлении соединений. Среди хостов, получивших отказ, могут быть и незараженные машины. Для них сервер становится недоступным. Администратор атакуемого сервера для частичного восстановления его работоспособности может увеличить лимит на число одновременных подключений и уменьшить время ожидания прихода ACK сообщения от хоста. Такая атака имеет второе название: DDOS attack (Distributed DOS attack – распределенная DOS атака).

### Протокол UDP

Рассмотрим работу протокола UDP (User Datagram Protocol - Протокол пользовательских дейтаграмм). Особенности данного протокола прямо противоположные свойствам протокола TCP: отсутствие соединения и пересылки квитанций, что не дает гарантию полной доставки всего сообщения. Отправленные сегменты могут теряться при пересылке. Приложения, использующие данный протокол, должны иметь встроенный механизм компенсации потерь некоторой части переданного сообщения. Единственным плюсом данного протокола является высокая скорость доставки данных. Применяется протокол тогда, когда важна скорость доставки, а не надежность. В качестве примера, где используется UDP, можно привести передачу голосового трафика, онлайн игры, видеоконференции в режиме реального времени.

Рассмотрим упрощенный пример передачи голоса. Аналоговый сигнал, представляющий собой волны сжатия или расширения воздуха и вызывающие колебания мембраны микрофона (рис. 9, а), преобразуется в цифровой сигнал (рис. 9, б). В одном из вариантов оцифровки уровень сигнала измеряется через равные промежутки времени. Сам процесс получения значений измеренного сигнала получил название аналогово-цифрового преобразования (АЦП). Полученные значения сигнала помещаются в разные UDP сегменты. Например, значения сигнала, стоящие на нечетных позициях, будут помещены в первый сегмент, на четных - во второй сегмент. При пересылке через сеть некоторые сегменты могут быть утеряны. В нашем примере пусть будет утерян второй сегмент. На стороне получателя по значениям, извлеченным из первого сегмента (рис. 10, а),

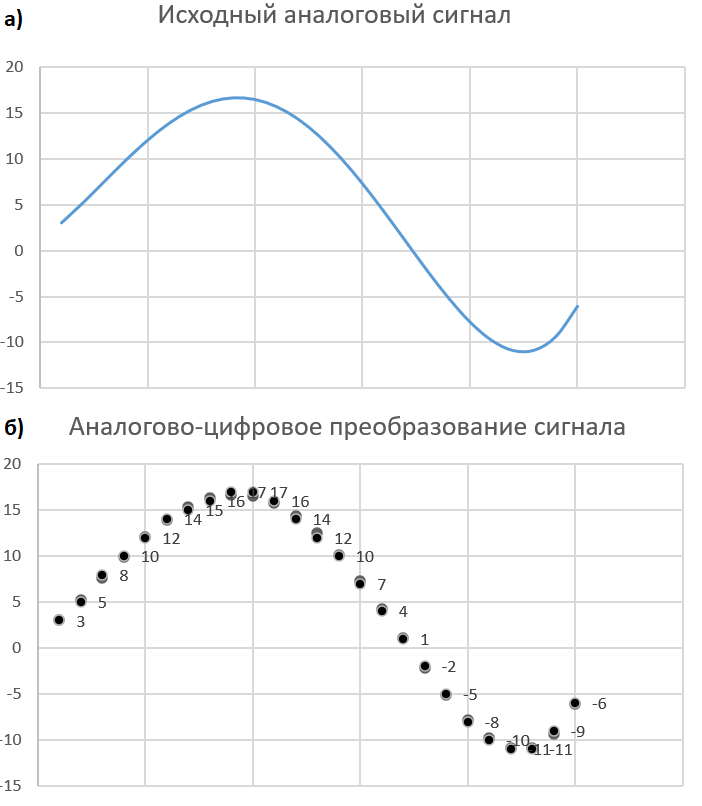


Рис. 9. Аналогово-цифровое преобразование

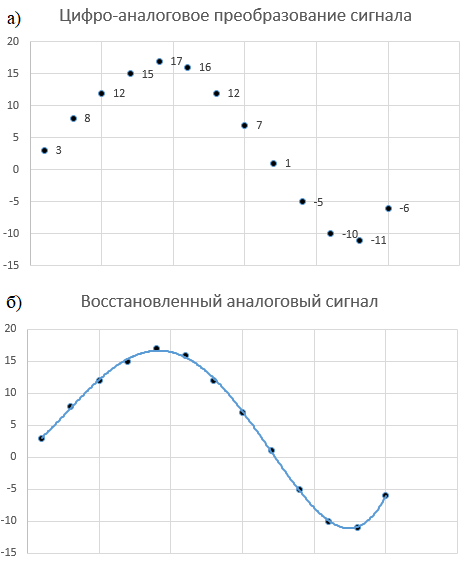


Рис. 10. Цифро-аналоговое преобразование

аналоговый сигнал будет восстановлен, но с некоторой потерей точности (рис. 10, б). Таким образом, тембр голоса собеседника будет передан с некоторым искажением. Но при этом собеседник будет вовремя услышан. Обратное преобразование цифрового сигнала в аналоговый называется цифро-аналоговым преобразованием (ЦАП).

## **Сетевой уровень модели OSI**

### Связь сетевого уровня с транспортным

Сетевой уровень является третьим в стеке OSI и отвечает он **за доставку сообщения от устройства к устройству** (например, от компьютера до сервера). На данном уровне используется протокол IP (Internet Protocol – Протокол межсетевого взаимодействия). Его предназначение – это адресация сетевых устройств. Когда на компьютере формируется пакет, то в его заголовок вносятся два IP адреса: отправителя и получателя. Это позволяет пакету пройти множество сетей и достичь своего адресата. Получатель пакета при этом будет знать от кого он получил данное сообщение и, соответственно, иметь возможность отправить ему ответ.

Адресация, примененная в компьютерных сетях, имеет много схожего с адресацией, используемой в обычной почтовой корреспонденции. Пакет представляет собой аналог почтового конверта, в который вложено само письмо. В нашем случае в пакет вкладывается TCP или UDP сегмент и задачей пакета является доставка сегмента на узел назначения. На почтовом конверте указываются адреса отправителя и получателя, так же и у пакета в заголовке присутствуют два IP адреса. По прибытии письма в дом, оно вкладывается в почтовый ящик квартиры адресата. Роль номера квартиры в нашем случае играет номер порта получателя, который указан в заголовке сегмента. Сама пара адресов носит название сокета (socket – розетка) и обозначается с использованием двоеточия: **IP адрес:порт**. Пример сокета: 192.168.17.175:50332, где 192.168.17.175 - IP адрес хоста, 50332 – порт, который использует сетевое приложение. Таким образом, взаимодействующие сетевые приложения с помощью двух сокетов образуют виртуальный канал связи, позволяющий им обмениваться между собой данными. Сообщение, отправленное в один сокет одним приложением, будет получено вторым приложением через другой сокет. На рисунке 11 показан пример канала связи, установленного между двумя приложениями. Сокет 192.168.17.175:50332 создан на компьютере пользователя, второй сокет 44.192.201.143:4244 – на удаленном сервере. Приложение пользователя имеет идентификатор PID (Process identifier - Идентификатор процесса) 4232. Состояние ESTABLISHED означает, соединение между приложениями установлено. Два процесса (с идентификаторами PID, равными 1016 и 4) прослушивают порты 135 и 445 в ожидании подключения (состояние LISTENING).

В языках программирования есть соответствующие классы, позволяющие сетевым приложениям создавать свои сокеты. Так, в Java есть два класса Socket и ServerSocket, используемые при создании клиент-серверного приложения. В конструкторе класса Socket

public Socket(String host, int port) throws UnknownHostException, IOException

первый параметр host задает доменный или IP адрес компьютера, на котором развернут сервер приложения, второй параметр port указывает порт, который использует сервер.

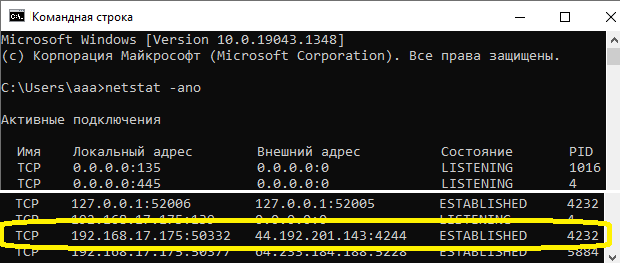


Рис. . Виртуальный канал связи

Конструктор класса ServerSocket

public ServerSocket(int port) throws IOException

задает значение порта, который сервер будет прослушивать.

В терминологии IP протокола устройства, обменивающиеся между собой пакетами, принято называть хостами. Сами хосты группируются в сети. Если следовать аналогии, приведенной выше, то сети соответствуют улицам. Внутри сети все хосты должны обладать своими уникальными, не повторяющими адресами, такое же требование касается и сетей: они должны иметь разные адреса.

На данный момент используются две версии протокола IP: версия 4, в которой для адреса выделено 32 бита, и версия 6 с 128 битовым адресом. 6-я версия имеет ограниченное применение и в курсовой работе она не рассматривается.

### Типы рассылок

Сообщения, передаваемые по сети, могут относиться к одной из четырех рассылок:

1. одноадресная (unicast). Такое сообщение адресовано одному узлу;
2. широковещательная (broadcast). Сообщение адресовано всем узлам сети;
3. групповая (multicast). Сообщение адресовано узлам одной группы;
4. unicast. Сообщение адресовано одному узлу одной группы.

### Типы IP адресов

Используются следующие типы адресов:

1. Хостовый адрес. Хостовые адреса назначаются сетевым картам компьютеров, портам маршрутизаторов, виртуальным интерфейсам устройств. В данном пособии хостовые адреса условно обозначаются как hIP (h – от слова host, хост).
2. Адрес сети или сетевой адрес. Принятое обозначение: nIP (network, сеть).
3. Широковещательный адрес с обозначением bIP (broadcast address).
4. Групповой адрес (mIP, multicast address).

Адреса получателя и отправителя будут обозначаться как IPs (source, источник) и IPd (destination, назначение, целевой адрес). Так как адрес отправителя относится к хостовому типу, то его можно обозначить в виде hIPs, соответственно адрес получателя – hIPd, bIPd или mIPd в зависимости от вида рассылки (unicast, broadcast или multicast).

### Двоичная и десятичная системы исчисления

В компьютерах информация хранится и обрабатывается в двоичном виде (в битовом представлении) и в таком же виде пересылается по сети. Следует хорошо владеть способами перевода числа из десятичной системы счисления в двоичную и правилами обратного перевода.

Рассмотрим вначале правило перевода двоичного числа в десятичное представление. Для этого вначале рассмотрим более знакомую нам десятичную систему. Возьмем в качестве примера число 20510. Его можно представить в виде суммы из трех чисел: . Число 10, возводимое в степень, является основанием десятичной системы. Аналогично двоичное число, например 10012 можно представить в виде суммы: , или , что дает число 910. В таблице 1 даны степени основания 2 с показателем от 0 до 7.

Таблица

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 |
| 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

Если рассматривается число со значением не более 255, то при переводе из десятичной в двоичную систему его можно разложить на сумму из чисел, представленных в таблице 1. Примеры перевода десятичных чисел в 8-битовое представление:

;

.

В таблице 2 приведены наиболее часто используемые 8-битовые комбинации.

Таблица

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Двоичное число | Десятичный эквивалент |  | Двоичное число | Десятичный эквивалент |
| 00000000 | 0 |  |  | |
| 00000001 | 1 |  | 11111110 | 254 |
| 00000011 | 3 |  | 11111100 | 252 |
| 00000111 | 7 |  | 11111000 | 248 |
| 00001111 | 15 |  | 11110000 | 240 |
| 00011111 | 31 |  | 11100000 | 224 |
| 00111111 | 63 |  | 11000000 | 192 |
| 01111111 | 127 |  | 10000000 | 128 |
| 11111111 | 255 |  |  | |

Самый младший бит определяет нечетность числа. Если он единичный, то число нечетное. Максимальное значение, которое дают 8 битов, равно 255.

### Структура IP адреса

IP адрес состоит из сетевой и хостовой составляющих (табл. 3).

Таблица

|  |  |  |
| --- | --- | --- |
| Адрес,  32 бита | Сетевая составляющая,  n битов | Хостовая составляющая,  32-n битов |
| hIP | xxx…xxx | xxx…xxx |
| nIP | xxx…xxx | 000…000 |
| bIP | xxx…xxx | 111…111 |
| SM | 111…111 | 000…000 |

В таблице 3 через x обозначены произвольные биты, т.е. вместо x можно подставлять либо 0, либо 1. Если хостовый (hIP), сетевой (nIP) и широковещательный (bIP) адреса относятся к одной сети, то их сетевые биты должны полностью совпадать. Все три типа отличаются друг от друга лишь хостовой частью. У хостового адреса в его хостовой части могут встречаться как нули, так и единицы. У сетевого адреса хостовая часть состоит полностью из нулевых битов, а у широковещательного адреса в хостовой части нулевые биты отсутствуют. Для отделения сетевой и хостовой частей друг от друга используется вспомогательное 32-битовое число, называемое маской подсети (subnet mask, SM). Маска адресом не является. Его сетевая часть (табл. 3) состоит из единичных битов, а хостовая – из нулевых битов. Число единичных битов маски называется длиной маской подсети (length subnet mask) и условно обозначается в виде /n.

### Методика вычисления сетевого и широковещательного адресов

При настройке сетевой карты компьютера или порта маршрутизатора им назначается IP адрес с указанием его маски. Как было указано выше этот адрес является хостовым (hIP). По этим двум двоичным числам вычисляется адрес сети (nIP) за счет побитового их умножения друг на друга (табл. 4).

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| & | hIP | xxx…xxx | xxx…xxx |
| SM | 111…111 | 000…000 |
|  |  | | |
|  | nIP | xxx…xxx | 000…000 |

Маска позволяет сохранить биты сетевой части хостового адреса неизменными и обратить в ноль биты хостовой части. Широковещательный адрес вычисляется из сетевого инвертированием битов его хостовой части (табл. 5).

Таблица

|  |  |  |
| --- | --- | --- |
| nIP | xxx…xxx | 000…000 |
| bIP | xxx…xxx | 111…111 |

### Примеры расчета адресов

Для представления адресов в четвертой версии протокола IP используется десятично-точечная нотация: четыре десятичных числа и три точки для отделения этих чисел друг от друга. Пример такого адреса был рассмотрен выше: 192.168.17.175.

Рассмотрим первый пример расчета адресов. Пусть сетевой карте назначен адрес 172.16.169.33/22. Вычисления начинаем с длины маски, которая представляет количество единиц, содержащихся в самой маске. Так как единицы располагаются в старших разрядах маски, они располагаются в левых октетах[[3]](#footnote-4): . Первые два октета полностью заполнены единицами, третий октет частично, а четвертый октет состоит из нулевых битов. По таблице 2 определяем маску подсети, как 255.255.252.0. Было бы ошибкой маску записать как 255.255.**63**.0. В третьем октете 6 единиц располагаются слева: 11111100, что дает число 252. Вариант 00111111 был бы ошибочен!

Определим адрес сети, в которой находится наш компьютер. Первые два октета относятся к сетевой части адреса. Это означает, что числа 172 и 16 одинаковы и для сетевого, и для хостового адресов. Четвертый октет входит в хостовую часть и для адреса сети данный октет заполняется нулями. Таким образом, nIP выглядит как 172.16.X.0, где X нам предстоит найти. В третьем октете хостового адреса находится число 169, битовое представление которого выглядит как 010101001. Старшие 6 битов относятся к сетевой части: 010101001. В сетевом адресе данный октет выглядит как: 010101000. Пояснение: два младших бита находятся в хостовой части адреса и в случае сетевого адреса должны быть нулевыми. Таким образом, мы получаем значение X в виде числа 168. Итак, адрес сети найден: 172.16.168.0.

Определим широковещательный адрес bIP. Его хостовая часть должна быть заполнена единицами. Следовательно, bIP = 172.16.Y.255. Значение Y находим по его битовому представлению 010101011. Здесь два младших бита представлены единицами. Отсюда следует, что значение Y равно 171, а широковещательный адрес равен 172.16.171.255.

Важно также уметь находить границы диапазона варьирования хостовых адресов [hIPmin; hIPmax]. Минимальный хостовый адрес очевидно будет получен, если его хостовая часть будет заполнена нулями кроме самого младшего бита, который будет единичным. Отсюда следует первое правило: для получения первого хостового адреса (hIPmin) берется сетевой адрес (nIP) и младший его бит инвертируется в единицу. В нашем примере hIPmin будет равен 172.16.168.1. Аналогично для получения максимального хостового адреса его хостовую часть заполняем единицами кроме самого младшего бита, который будет нулевым. Следовательно, второе правило гласит: максимальный хостовый адрес образуется из широковещательного заменой его младшего бита на ноль. Для нашего примера hIPmax будет равен 172.16.171.254.

Максимальное число хостовых адресов для данной сети определится по формуле .

Перейдем ко второму примеру. Пусть помимо хостового адреса 137.53.66.53 известна его маска в виде 255.255.255.224. Начинаем решение данной задачи с определения длины маски подсети (табл. 2): . Таким образом, первые три октета входят в сетевую составляющую адреса. Также старшие три бита четвертого октета являются сетевыми, и они в наших расчетах изменяться не будут. Адрес сети будет 137.53.66.X. Для вычисления X рассмотрим битовое представление числа 53: 00110101. В адресе сети младшие пять битов должны быть обращены в ноль: 00100000, что дает число 32. Мы определили сетевой адрес как 137.53.66.32. В широковещательном адресе эти же биты должны быть обращены в 1, в итоге имеем 00111111, а в десятичном представлении - число 63. Таким образом, широковещательный адрес имеет значение 137.53.66.63. Находим остальные искомые величины:

1. первый хостовый адрес - 137.53.66.33;
2. последний хостовый адрес - 137.53.66.62;
3. число хостов .

### Экскурс в историю IP протокола

Разработчики протокола IP изначально не смогли правильно оценить масштабы будущей глобальной сети и предусмотрели для адреса всего лишь 32 бита. Также в первой концепции протокола (1980 год) лишь старшие 8 битов (первый октет) отвели под сетевую составляющую, а остальные 24 бита – под хостовую часть. Т.е. было предусмотрено создание 28 (256) сетей, но каждая из этих сетей имела бы огромные размеры - по 224 (16 777 216) адресов. Позже по мере лавинообразного увеличения в мире количества сетей начали предприниматься меры по увеличению количества сетевых адресов:

1. использование классовой адресации (1981 год);
2. переход к бесклассовой адресации (1993 год);
3. разделение адресов на публичные и приватные (1996 год) и использование NAT (Network address translation – Трансляция сетевых адресов);
4. создание 6-й версии протокола IP (1998 год).

### Классовая адресация

В сентябре 1981 года появилась четвёртая версия протокола IP (IPv4, документ RFC 791), согласно которой все адреса были распределены по 5 классам: A, B, C, D и E [9]. Первые три класса (A, B, C) предназначены для хостовых (hIP), сетевых (nIP) и широковещательных (bIP) адресов, класс D – для групповых (mIP) адресов. Адреса класса E были зарезервированы для использования в будущем. Выяснить принадлежность адреса тому или иному классу можно по первым битам первого октета (табл. 6).

Таблица

|  |  |  |
| --- | --- | --- |
| Класс | Биты первого октета | Диапазон варьирования первого числа адреса |
| A | 0XXXXXXX | 0 - 127 |
| B | 10XXXXXX | 128 - 191 |
| C | 110XXXXX | 192 - 223 |
| D | 1110XXXX | 224 - 239 |
| E | 1111XXXX | 240 - 255 |

В данной таблице через X обозначен произвольный бит. Маска подсети применяется в первых трех классах и имеет фиксированное значение (табл. 7): 8 – в классе A, 16 – в классе B и 24 – в классе D.

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| Класс | Адреса | | |
| сетевые, nIP | Хостовые, hIP | Широковещательные, bIP |
| A | N.0.0.0 | N.H.H.H | N.255.255.255 |
| B | N.N.0.0 | N.N.H.H | N.N.255.255 |
| C | N.N.N.0 | N.N.N.H | N.N.N.255 |

Через N (Network - сеть) обозначено число, относящееся к сетевой части адреса, через H – число хостовой части (Host). Наибольшее число сетей имеет класс C. В данном классе число изменяемых сетевых битов равно , что дает 221 (2097152) сетевых адресов. Но размеры сетей данного класса по сравнению с классами A и B относительно небольшие: 28 – 2 = 254 хостовых адреса.

Распределение адресов суперсетей с маской /8 можно просмотреть на ресурсе [10]. Большие блоки адресов выделены военному ведомству США. Распределением адресов внутри некоторых суперсетей занимаются региональные регистраторы [11]:

1. AFRINIC (Африка);
2. ARIN (Северная Америка);
3. APNIC (Азиатско-Тихоокеанский регион);
4. LACNIC (Латинская Америка);
5. RIPE NCC (Европа, Центральная и Западная Азия, Россия).

### Служебные адреса

Некоторые IP-адреса выполняют особые функции. Два диапазона адресов из класса A отданы под служебные цели: 0.X.X.X и 127.X.X.X. Так, например, 127.0.0.1 (его доменный аналог - localhost) является адресом обратной петли и используется для проверки настроек сетевой карты (рис. 12).

Ping является сетевой утилитой операционной системы и используется для проверки доступности удаленного узла. Его алгоритм основан на протоколе ICMP (Internet control message protocol - Протокол межсетевых управляющих сообщений). В примере, рассмотренном на рисунке 12, пингом проверяется доступность самой сетевой карты. Она настроена верно.

Адрес 255.255.255.255 из класса E является широковещательным. Он используется, например, в протоколе DHCP (Dynamic Host Configuration Protocol – Протокол динамического конфигурирования хоста). Данный протокол позволяет автоматизировать раздачу узлам их сетевых настроек. В сети на сервере или маршрутизаторе запускается и настраивается DHCP сервер. На хосте, на котором сетевая карта еще не настроена и не имеет IP адреса, запускается DHCP клиент. Клиент, не зная в какой сети находится его узел, отправляет в сеть широковещательный DHCP запрос, в котором адрес hIPd равен 255.255.255.255, а в качестве адреса hIPs указывается 0.0.0.0. В ответ на полученный запрос DHCP сервер вышлет узлу его IP адрес, маску и другие сетевые настройки. IP адрес выдается хосту на определенный срок, называемый сроком аренды (lease time). Незадолго

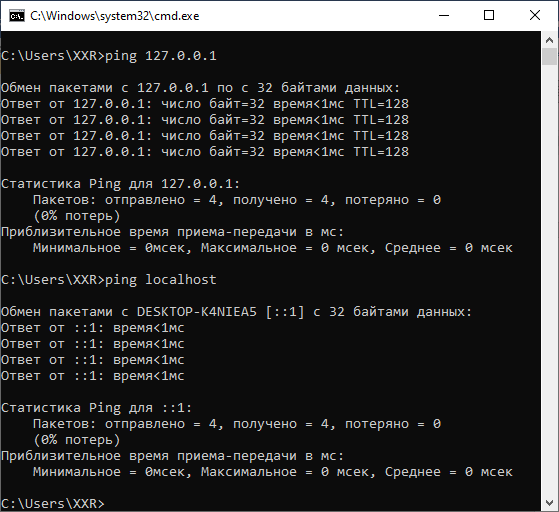


Рис. . Проверка командой ping настроек сетевой карты компьютера

до истечения данного срока хост должен новым запросом к серверу продлить срок аренды адреса.

### Бесклассовая адресация

В классовой адресации были предусмотрены сети очень большого размера, но самих сетей было относительно мало. Особенно это касалось сетей класса A. Было принято решение увеличить адресное пространство для сетей за счет уменьшения хостовой составляющей адреса. С этой целью часть битов передали из хостовой части в сетевую (рис. 13). Длина маски при этом увеличилась, т.е. длина маски перестала быть фиксированной величиной (8, 16 или 24). Такой прием получил название VLSM (Variable length subnet mask - Маска подсети переменной длины).

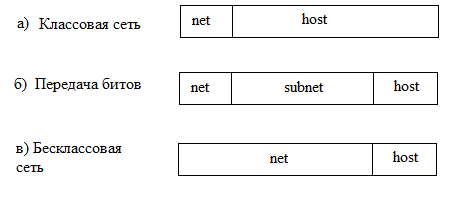


Рис. . Разбиение классовой сети на подсети

Выше были рассмотрены два примера определения адресов сетей. В первом примере был найден адрес сети 172.16.168.0/22. На самом деле эта сеть является подсетью классовой сети 172.16.0.0/16, так как число 172 входит в диапазон значений первого октета для класса B (таблицы 6 и 7). Во втором примере подсеть 137.53.66.32/27 является частью классовой сети 137.53.0.0/16 (также класс B).

### Subnetting

Термин **subnetting** означает деление сети на подсети. Делить на части можно не только классовую сеть. Любую полученную из классовой сети подсеть, в свою очередь, можно также подвергнуть дальнейшему разбиению на более мелкие подсети. Рассмотрим алгоритм разбиения на подсети более подробно. Биты, переданные из хостовой части в сетевую, позволяют создать новые сетевые адреса. Так, один переданный бит дает 2 адреса (21 = 2 комбинации), если передано 2 бита, то получаем 4 новых адреса (22 = 4 комбинации). В качестве примера рассмотрим классовую сеть 192.168.0.0/24. Увеличим маску на один бит. Четвертый октет сетевого адреса будет либо в виде 00000000, либо 10000000. Первый вариант соответствует подсети 192.168.0.0/25, второй вариант подсети 192.168.0.128/25. Ту же исходную сеть (192.168.0.0/24) попробуем разбить на четыре подсети. На этот раз сетевой части передаем 2 бита, что дает четыре битовых варианта четвертого октета: 00000000 (010), 01000000 (6410), 10000000 (12810), 1100000 (19210). Следовательно, мы создали подсети со следующими адресами: 192.168.0.0/26, 192.168.0.64/26, 192.168.0.128/26, 192.168.0.192/26. Нумерацию созданных подсетей принято начинать с нуля, т.е. 192.168.0.0/26 будет нулевой подсетью (zero subnet), 192.168.0.64/26 – 1-й подсетью и т.д.

Проведем дальнейшее разделение одной из полученных подсетей. В качестве примера возьмем подсеть 192.168.0.128/26 и ее поделим на две подсети. Четвертый октет примет вид 10000000 и 10100000, что даст адреса 192.168.0.128/27 и 192.168.0.160/27. В таблице 8 приведено общее число адресов и число хостовых адресов. При необходимости данные таблицы можно легко экстраполировать за пределы отрезка [24; 27].

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| Длина маски | Число битов в хостовой части адреса | Общее число адресов | Число хостовых адресов |
| 24 | 8 | 28 = 256 | 254 |
| 25 | 7 | 27 = 128 | 126 |
| 26 | 6 | 26 = 64 | 62 |
| 27 | 5 | 25 = 32 | 30 |

### Supernetting

Для упрощения навигации в Интернете сети объединены в суперсети. Такая операция обратна по смыслу термину **subnetting**, т.е. в данном случае ставится задача объединения нескольких сетей в одну общую сеть (supernet). Для выполнения этой операции должно выполняться следующее условие: у объединяемых сетей должны совпадать старшие биты сетевой части их адресов. В самом лучшем варианте у объединяемых сетей должны быть смежные адреса. В качестве примера можно привести сети 172.16.0.0/27, 172.16.0.32/27, 172.16.0.64/27, …. Здесь адреса меняются с шагом 32. Но в общем случае у объединяемых сетей даже их длины масок могут не совпадать.

Supernetting (называемый также Aggregation - Агрегирование) проще понять, если сами сети условно представить в виде хостов, а суперсеть в виде обычной сети. У адресов хостов и адреса самой сети биты сетевой части одинаковы, а биты хостовой - разные. На рисунке 14 показан общий принцип агрегирования сетей. У объединяемых сетей (рис. 14, а) выявляются совпадающие между собой биты **n** (рис. 14, б), и различающиеся биты **h** переводятся в хостовую часть, что приводит к их обнулению. Маска полученной суперсети (рис. 14, в) должна быть короче, чем у вложенных сетей.

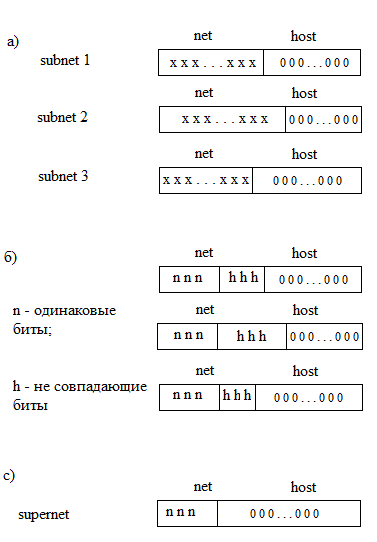


Рис. . Агрегирование сетей

Рассмотрим пример. Пусть даны три смежные сети: 10.200.0.0/16, 10.201.0.0/16, 10.202.0.0/16. Биты у них различаются во втором октете: 011001000 (20010), 011001001 (20110), 011001010 (20210). Одинаковыми являются первые шесть битов: 011001000 (20010), 011001001 (20110), 011001010 (20210). Эти 6 битов оставляем сетевыми, а остальные два младших бита, переводим в хостовую часть и обращаем в ноль: 011001000 (20010), и в итоге получаем адрес суперсети: 10.200.0.0/14.

К понятию «суперсеть» близок термин CIDR (Classless Inter-Domain Routing - Бесклассовая междоменная маршрутизация) [12], позволяющий объединять маршруты к нескольким сетям в один общий маршрут. Тема маршрутизации будет рассмотрена в разделе «Маршрутизаторы».

### Публичные и приватные адреса, технология NAT

Переход к бесклассовой адресации лишь ненадолго решил проблему нехватки сетевых адресов, ибо количество сетей возрастало в геометрической прогрессии и в 1996 году было принято решение о разделении адресов на публичные (второе, обиходное название - белые) и приватные (серые). Три пула адресов (табл. 9), которые в то время еще никому не были выделены, были переведены в разряд приватных адресов. До этого времени все хосты были доступны с любого узла интернета. Адреса таких хостов стали называть публичными. Приватные же адреса отличны тем, что они доступны лишь внутри своей локальной сети и извне, с интернета, они не видны. Выигрыш от использования приватных адресов заключается в том, что в мире миллионы сетей могут иметь один и тот же приватный адрес. Совпадение же их адресов не приводит к конфликтной ситуации, так как такие сети обособлены друг от друга.

Таблица

|  |  |
| --- | --- |
| Класс | Приватные адреса |
| A | 10.X.X.X |
| B | 172.Y.X.X |
| C | 192.168.X.X |

Примечание к таблице: 0 ≤ X ≤ 255, 16 ≤ Y ≤ 31.

На данный момент весь компьютерный мир окрашен в серый цвет, т.е. подавляющее большинство персональных компьютеров, ноутбуков, планшетов, смартфонов и других IP устройств имеют приватные адреса. Но если взглянуть на сам Интернет, то он будет полностью белым, ибо в глобальной сети доступными являются лишь те устройства, которые имеют публичные адреса. Но как же все устройства, имея приватные адреса, общаются с внешним миром? В этом им помогает NAT (Network Address Translation – Трансляция сетевых адресов).

Рассмотрим в упрощенном виде механизм работы технологии NAT. Как правило, на границе приватной и публичной областей находится пограничный маршрутизатор (border router) провайдера. Провайдер, выделив своим клиентам (как юридическим, так и физическим лицам) пулы приватных адресов, на своем маршрутизаторе настраивает NAT. В примере, приведенном на рисунке 15, клиент пытается установить TCP соединение с сервером, находящимся в Интернете. Поскольку у сервера публичный адрес, клиент имеет возможность отправить ему свой SYN запрос. Когда сообщение проходит через маршрутизатор, то он подменяет приватный адрес клиента hIP1 на свой публичный hIP2

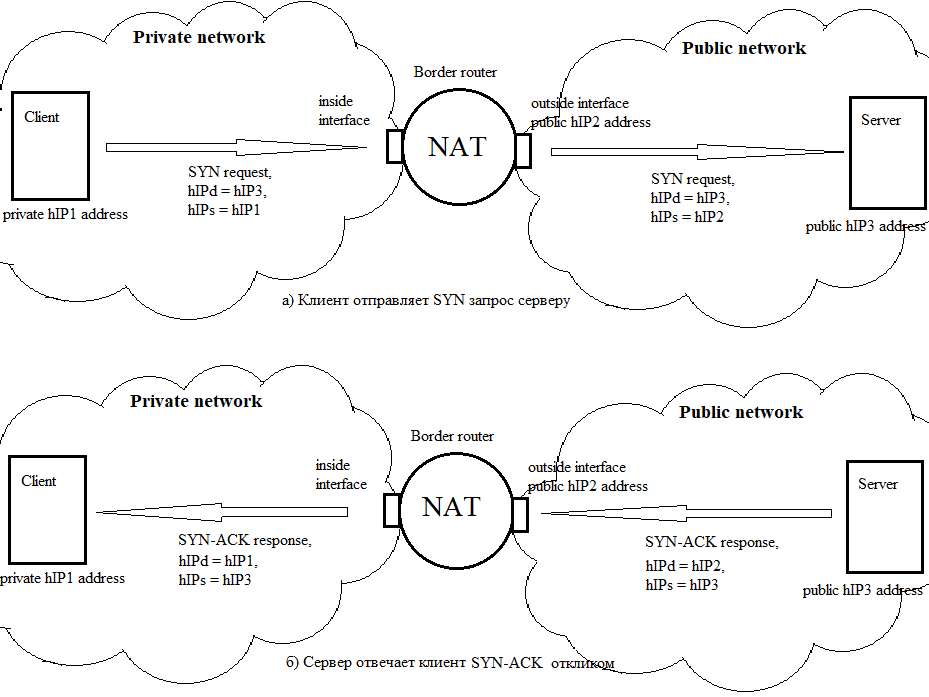


Рис. . Упрощенная схема работы NAT.

(рис.15, а). При прохождении ответа от сервера маршрутизатор делает обратную подмену, подставляя в качестве hIPd адрес клиента (рис.15, б). Для клиента NAT работает в прозрачном режиме, сервер же считает, что он взаимодействует с хостом, у которого адрес hIP2.

Рассмотрим вариант, когда клиент находится в public сети, а сервер в приватной зоне. Поскольку у сервера приватный адрес, то клиент его не видит. Единственным решением может быть отправка клиентом своего запроса на публичный адрес пограничного маршрутизатора. Как маршрутизатор определит, что полученный запрос следует перенаправить внутреннему серверу? В этой ситуации NAT позволяет использовать адрес транспортного уровня, а именно порт. Допустим, что сервером является web сервер, который всегда прослушивает порт 80. NAT можно настроить на пропуск внешних SYN запросов с destination портом, равным 80, на внутренний сервер (рис. 16).

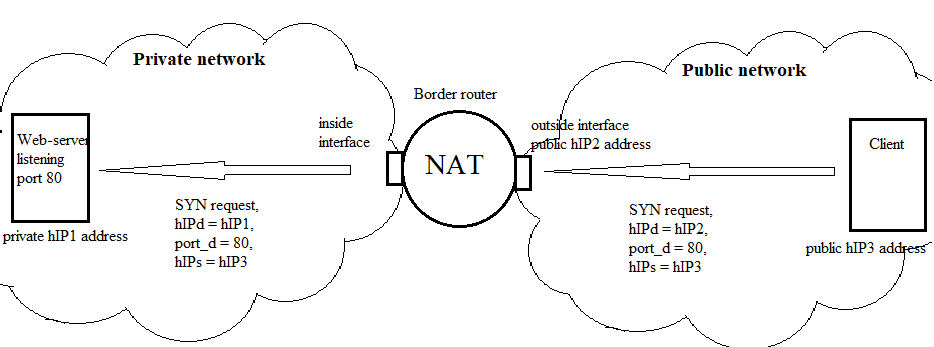


Рис. . Внешний SYN запрос

Приватные адреса и NAT позволили надолго решить проблему нехватки публичных адресов. К минусу технологии NAT можно отнести повышенную нагрузку, накладываемую на пограничный маршрутизатор. У провайдера приватная сеть может достигать больших размеров и число одновременных подключений «клиент-сервер», которые маршрутизатор должен поддерживать, может быть очень велико. К таким пограничным маршрутизаторам предъявляются повышенные требования как по количеству и мощности процессоров, так и по объему оперативной памяти.

### Выполнение контрольного теста по IP адресации

Тест выполнен в виде Java апплета и находится по адресу http://www.altaev-aa.narod.ru/net/ip. Предварительно следует к браузеру на основе Chromium добавить расширение CheerpJ Applet Runner с адреса: chrome.google.com/webstore/detail/cheerpj-applet-runner/bbmolahhldcbngedljfadjlognfaaein. Для выполнения апплета следует запустить данное расширение (рис. 17).

## **DNS**

Доменный адрес применяется на 7-м, прикладном, уровне модели OSI (табл. 1). Необходимость в использовании подобных адресов стала очевидной еще тогда, когда сети только появились. Пользователям сети было проще запомнить буквенно-числовые обозначения узлов, нежели их IP адреса. Технология DNS (Domain Name System - Система доменных имён) позволяет по URL[[4]](#footnote-5) адресу какого-либо сетевого ресурса определить

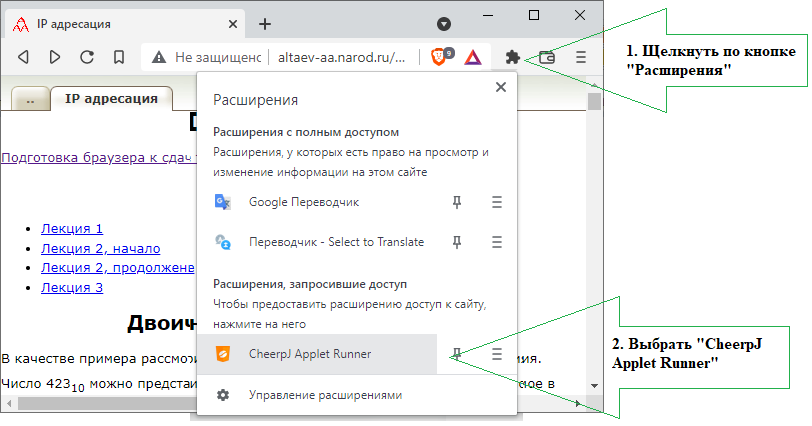


Рис. 17. Запуск апплета

его IP адрес. Примером может служить отправка эхо сообщения на ресурс esstu.ru (рис. 18).

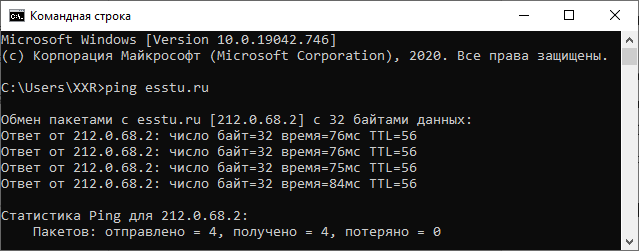


Рис. 18. Доменный и IP адреса.

В данном примере мы видим, что DNS хранит информацию о ресурсе esstu.ru и закрепленным за ним IP адресе 212.0.68.2.

На начальном этапе развития DNS службы его записи о ресурсах сети хранились в текстовых файлах hosts на компьютерах пользователей. Пользователям приходилось обмениваться записями, что было не совсем удобно. Позже стали использовать выделенные машины с запущенными на них DNS серверами. В настоящее время сервера образуют иерархическую структуру, в вершине которой расположены серверы доменов высшего уровня [24]. Сами адреса отображают структуру (рис. 19, [25]) иерархии доменов.

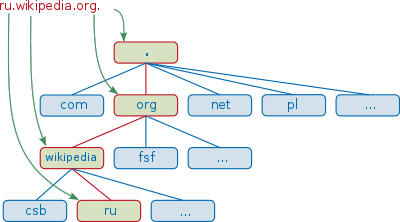


Рис. 19. Структура доменного имени.

Завершающая точка в имени обозначает корень дерева иерархии. В DNS запросах она не указывается, но обязательна в настройках конфигурации серверов.

Файл hosts сохранил свое присутствие в современных операционных системах [26]. В ОС Windows он хранится по адресу C:\Windows\System32\drivers\etc. Один из примеров его содержания:

**# Copyright (c) 1993-2009 Microsoft Corp.**

**#**

**# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.**

**#**

**# This file contains the mappings of IP addresses to host names. Each**

**# entry should be kept on an individual line. The IP address should**

**# be placed in the first column followed by the corresponding host name.**

**# The IP address and the host name should be separated by at least one**

**# space.**

**#**

**# Additionally, comments (such as these) may be inserted on individual**

**# lines or following the machine name denoted by a '#' symbol.**

**#**

**# For example:**

**#**

**# 102.54.94.97 rhino.acme.com # source server**

**# 38.25.63.10 x.acme.com # x client host**

**# localhost name resolution is handled within DNS itself.**

**# 127.0.0.1 localhost**

**# ::1 localhost**

Доменное имя localhost применяется для проверки доступности самой сетевой карты (рис. 12, раздел «Служебные адреса»). При поиске запрашиваемого доменного адреса вначале проверяется файл hosts, далее DNS запрос идет последовательно на DNS сервер компании (если он был настроен), DNS сервер провайдера, сервер домена высшего уровня.

## **Сетевые устройства**

Различие между коммутаторами (switch) и маршрутизаторами (router) заключается в том, первые используются для создания сетей, а вторые – для соединения сетей друг с другом. Коммутаторы имеют большое количество портов (рис. 20 ), к каждому из которых кабелем может быть присоединен отдельный компьютер. И все подключенные к этому коммутатору компьютеры будут относиться в одной сети[[5]](#footnote-6).

Маршрутизаторы же имеют относительно мало портов (рис. 21), но все они принадлежат разным сетям. Эти порты фактически являются IP хостами, на которых указывается их хостовый IP адрес. У коммутаторов их порты не предусматривают настройки IP адреса[[6]](#footnote-7). Таким образом, маршрутизаторы за счет того, что их порты расположены в разных сетях, осуществляют трафик из одной сети в другую (т.е., отвечают за **межсетевой трафик**). Обязанность же коммутаторов – это передача **внутрисетевого трафика**.



Рис. . Коммутатор Cisco SG500-28P-K9-G5

У коммутатора все порты относятся к одному типу (как правило, это Fast Ethernet или Gigabit Ethernet порты). Маршрутизаторы же соединяют между собой сети, передача данных в которых может быть реализована с помощью разных сред: витая пара, оптоволокно, Wi-Fi и т.д. По этой причине у них порты могут быть разных типов. Так, например, на рисунке 21 маршрутизатор имеет пару fastethernet портов (наклейки желтого цвета) и один serial порт (синяя наклейка). Wi-Fi маршрутизатор (рис. 22) предназначен для передачи трафика между Wi-Fi и Ethernet сетями.



Рис. . Маршрутизатор Cisco 1841

И маршрутизаторы, и коммутаторы являются специализированными компьютерами со своей операционной системой. Так, например, у коммутаторов и маршрутизаторов компании Cisco [36] операционная система носит название IOS (Internetwork Operating System - Межсетевая операционная система[[7]](#footnote-8)).



Рис. . Wi-Fi маршрутизатор Cisco C819H-K9.

Так как оба сетевых устройства не имеют своего периферийного оборудования в виде монитора и клавиатуры, то для их настройки используют консольный кабель, обычный компьютер или ноутбук (рис. 23) с запущенной на них терминальной программой (PuTTY, KiTTY и т.п.). На рисунках 24, 25 и 26 показана настройка консольного подключения через PuTTY. Консольное подключение необходимо для базовой настройки сетевого устройства.

|  |  |
| --- | --- |
| Рис. . Консольный кабель | Рис. . Подключение по консольному кабелю к устройству через PuTTY. Шаг 1. |
| Рис. 25 Подключение по консольному кабелю к устройству через PuTTY. Шаг 2. | Рис. 26. Подключение по консольному кабелю к устройству через PuTTY. Шаг 3. |

На рисунке 27 показано терминальное окно KiTTY в момент открытия SSH сессии. Заранее, на этапе базовой настройки коммутатора или маршрутизатора, на них должен быть запущен SSH сервер, а их интерфейсам (портам) должны быть назначены IP адреса и сами порты должны быть активированы. В таком случае, режим работы программы KiTTY можно перевести в формат SSH клиента с указанием параметров сокета, используемого SSH сервером, а именно одного из адресов сетевого устройства и номера порта (SSH использует порт 22). На рисунке пока еще не введен IP адрес устройства.

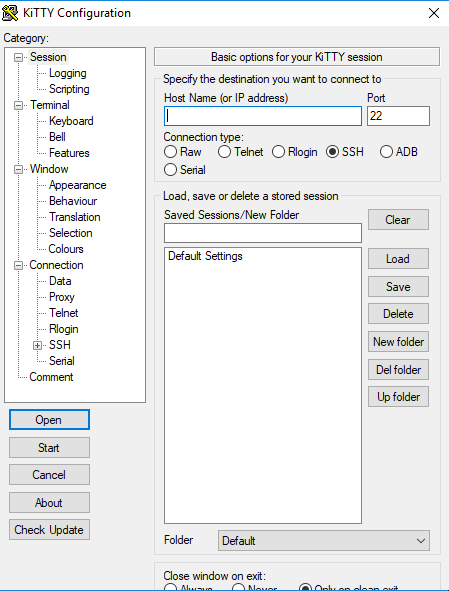


Рис. . KiTTy в роли SSH-клиента

Концентраторы (hub, device L1) не относятся к интеллектуальным устройствам, так как их основным предназначением является передача потока битов. Двухпортовые концентраторы (репитеры) применяются в магистральных оптоволоконных линиях связи для регенерации световых сигналов. Они, последовательно выполняя аналого-цифровое и обратное цифро-аналоговое преобразование сигнала, восстанавливают форму и мощность светового импульса, тем самым позволяя добиться передачи информации на очень большие расстояния.

## **Канальный уровень модели OSI**

### Предназначение канального уровня

Канальный уровень **отвечает за доставку сообщения от одного устройства к другому устройству, причем они оба должны быть в одной сети**. Рассмотрим передачу сообщения на примере сети, состоящей из двух конечных (end device) и двух промежуточных (intermediary device) устройств (рис. 28). Конечные устройства находятся на периферии сети, к ним, как правило, подведен лишь один сетевой кабель, и они являются источником сетевого трафика и его потребителем. Это компьютеры, ноутбуки, сервера, смартфоны, принтеры и другие устройства пользователя. В нашем примере их роль играют два компьютера.

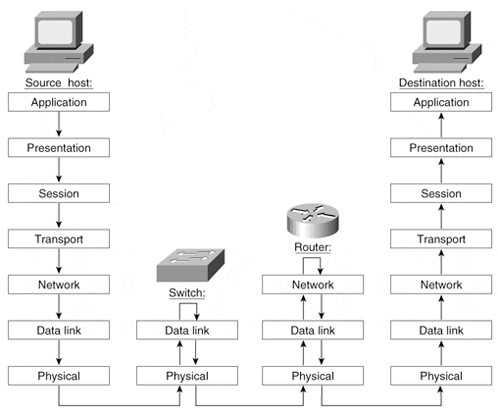


Рис. . Иллюстрация прохождения данных по маршруту: исходный хост, коммутатор, маршрутизатор, хост назначения.

Промежуточные устройства имеют несколько портов, т.е. они посредством кабелей связаны с несколькими соседями и способны через себя пропускать трафик. В нашем случае ими являются коммутатор (switch, устройство канального уровня) и маршрутизатор (router, устройство сетевого уровня). Рассматриваемая топология содержит две сети, так как присутствует лишь один маршрутизатор. На хосте, который сообщение отправляет, формируется пакет, на уровне Data link пакет помещается во внутрь фрейма. Далее фрейм в виде потока битов по кабелю отправляется на коммутатор. Коммутатор принимает фрейм на один из своих портов и передает его свой второй порт, далее фрейм уходит в сторону маршрутизатора. Как видим, коммутатор сам фрейм не изменяет. Маршрутизатор является устройством 3-го уровня (device L3). По прибытии сообщения на порт маршрутизатора, оно, поднимаясь по стеку OSI, проходит декапсуляцию, т.е. на уровне Data link заголовок фрейма вместе с трейлером удаляются и на уровень Network передается лишь пакет. Пакет передается с порта прибытия на порт убытия и там происходит обратный процесс – инкапсуляция. Пакет перемещается на канальный уровень, на котором порт убытия для пакета создает новый фрейм. И фрейм отправляется в сторону хоста destination. Какой вывод можно сделать из рассмотренного примера? Фреймы пределы своей сети не покидают, т.е. второй уровень (layer 2, Data link layer), отвечает за доставку сообщения только внутри сети. Фрейм через маршрутизатор не передается, в отличие от пакета. Пакет, являясь сообщением уровня 3 и будучи сформированным на одном конечном устройстве, без изменений, пройдя через несколько сетей, доставляется на другое конечное устройство. Исключением является лишь пограничный маршрутизатор, на котором NAT подменивает IP адреса в заголовке пакета.

### Протоколы канального уровня

На втором уровне различают два типа сетей: широковещательные и двухточечные. Двухточечную сеть образуют, например, два маршрутизатора, соединенные между собой serial кабелем. В сети такого типа используются протоколы HDLC (High-Level Data Link Control - Высокоуровневая процедура управления каналом данных) и PPP (Point-to-Point Protocol - Двухточечный протокол). У Cisco маршрутизаторов на их serial интерфейсах по умолчанию настроен протокол HDLC. Реализация данного протокола у разных вендоров различается и если двухточечную сеть образуют маршрутизаторы разных вендоров, то в такой сети следует использовать протокол PPP.

В широковещательных сетях могут присутствовать сотни и тысячи устройств. В то время как в двухточечных сетях отсутствует необходимость в адресах канального уровня, технологии широковещательных сетей: Wi-Fi и Ethernet, используют MAC адреса (Media Access Control – Контроль за доступом к среде). MAC адреса представляют из себя 48-битовые числа, и они, как было сказано выше, являются аппаратными. Вендорам выделены определенные пулы, из которых они выбирают адреса при изготовлении чипов своих интерфейсных карт. Как и в IP протоколе есть широковещательный MAC адрес, состоящий из 48 единичных битов и используемый в качестве адреса назначения во фрейме.

Когда интерфейсная карта устройства (например, Fast Ethernet порт маршрутизатора, сетевая карта или Wi-Fi адаптер компьютера), готовится отправить пакет в сеть, то ему необходимо сформировать фрейм (Fast Ethernet или Wi-Fi фрейм, соответственно) с указанием в нем адресов MACd[[8]](#footnote-9) и MACs. Как карта выясняет адрес получателя MACd его фрейма? В решении данной задачи карте помогает протокол ARP (Address Resolution Protocol - Протокол определения адреса). Рассмотрим подробно алгоритм работы этого протокола.

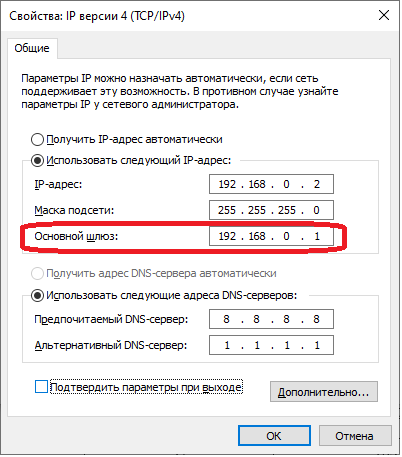


Рис. . Настройки сетевой карты компьютера.

Сам передаваемый пакет может относиться либо к внутрисетевому, либо к межсетевому трафику. В первом случае следует найти MACd узла, который находится в одной сети с отправителем, во втором случае - адрес Fast Ethernet порта маршрутизатора или Wi-Fi адаптера точки беспроводного доступа (Wi-Fi маршрутизатора). Пояснение: если пакет необходимо передать в другую сеть, то его можно туда перенаправить лишь через маршрутизатор. Сетевая карта хранит в своей памяти адрес своей сети (nIPs), вычисленную через ее хостовый IP адрес и маску подсети путем побитового перемножения (см. раздел «Методика вычисления сетевого и широковещательного адресов»). ARP вычисляет nIPd, т.е. адрес сети, в которой находится получатель пакета, побитовым перемножением адреса получателя пакета hIPd на маску своей карты. Оба сетевых адреса сравниваются. Если nIPs и nIPd совпадают, то пакет относится к внутрисетевому трафику, если нет - то к межсетевому. Затем карта отправляет широковещательный ARP запрос в сеть с указанием в нем или hIPd адреса (если пакет принадлежит к внутрисетевому трафику) или адреса основного шлюза (если пакет принадлежит к межсетевому трафику). Пояснение: основной шлюз – это адрес порта маршрутизатора (рис. 29). Все узлы сети получают данный запрос и лишь один из них, чей IP адрес совпал с адресом в запросе, отвечает ARP откликом, сообщая в нем свой MAC адрес. На этом работа протокола ARP завершается, и сетевая карта формирует фрейм, помещая в нее пакет, и отправляет его либо на другой узел своей сети, либо на маршрутизатор.

Какой вывод можно сделать из анализа работы протокола ARP? Если при настройке сетевой карты не был указан основной шлюз, то узел теряет возможность отправлять свои пакеты в другие сети, ибо ARP не сможет определить MAC адрес порта маршрутизатора.

### Frame Relay и MPLS

В данном разделе рассмотрим технологии, относящиеся к канальному уровню, но предназначенные для работы не в локальных сетях, а в сетях гораздо большего масштаба. В качестве первого примера можно привести Frame Relay. Отметим сразу, что она в Российской Федерации провайдерами не поддерживается. Frame Relay использует адреса канального уровня, называемые DLCI (Data Link Connection Identifier - Идентификатор соединения канала данных). DLCI позволяют создать виртуальный канал связи через сеть провайдера. Такие каналы позволяют объединить разные части одной распределенной корпоративной сети организации, например сеть головного офиса с сетями филиалов.

На рисунке 30 приведен пример распределённой корпоративной сети, построенной с применением технологии Frame Relay. Сеть провайдера состоит из пяти маршрутизаторов R1 – R5, переведенных в режим коммутации Frame Relay. Это им позволяет работать лишь с заголовками Frame Relay кадров. Такой способ пересылки пакетов работает быстрее, так как маршрутизаторами не просматриваются заголовки 3-го уровня. Корпоративная сеть состоит из сети головного офиса (network HQ[[9]](#footnote-10)) и сетей филиалов (network Branch1 и network Branch2). Виртуальные каналы создаются администратором провайдера. Первый канал соединяет сети HQ и Branch1 и представляет собой набор значений DLCI: 301, 100, 200, 103. Второй канал соединяет HQ и Branch2 и имеет набор DLCI: 302, 16, 22, 203.

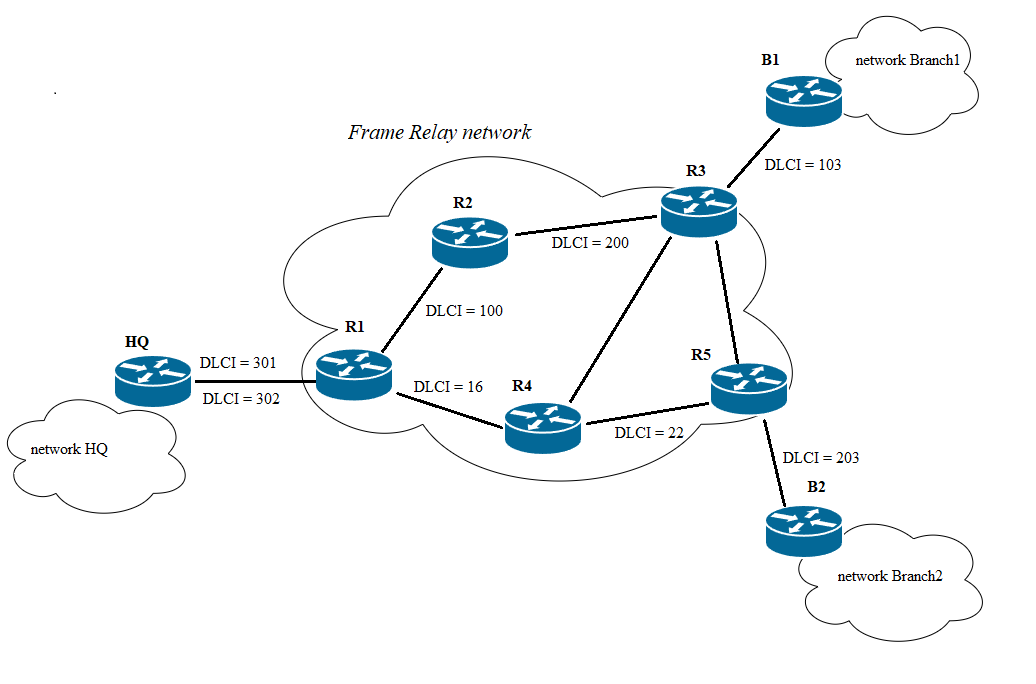


Рис. . Топология распределенной корпоративной сети, построенной с применением технологии Frame Relay.

Администратор корпоративной сети, настраивая маршрутизаторы HQ, B1 и B2, указывает на них в качестве входных значений DLCI 301, 103, 302 и 203. Таким образом, в итоге образуется Frame Relay сеть из внутренних маршрутизаторов R1 – R5 и трех пограничных маршрутизаторов HQ, B1 и B2.

Рассмотрим, как работает такая сеть. Если пакет идет из сети network HQ в сеть network Branch1, то он попадает на маршрутизатор HQ. Маршрутизатор создает Frame Relay кадр, в заголовок которого помещает DLCI равное 301. Фрейм по кабелю перемещается на маршрутизатор R1. Когда фрейм входит в интерфейс (порт) S0 (рис. 31), то из его заголовка извлекается его DLCI. Далее проверяется таблица коммутации (табл. 10) и определяется выходные интерфейс и DLCI: S1 и 100. Таким образом, фрейм через выходной интерфейс S1 будет перенаправлен соседнему маршрутизатору R2 (рис. 30). В его заголовке будет обновленное значение DLCI, равное 100.

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| Интерфейс | DLCI | Интерфейс | DLCI |
| S0 | 301 | S1 | 100 |
| S0 | 302 | S2 | 16 |

Если же из сети network HQ пакет необходимо направить в сеть network Branch1, то маршрутизатор HQ в заголовок кадра поместит DLCI, равный 302. Маршрутизатор R1 в таком случае перенаправит кадр в сторону R4, поменяв DLCI на 16.

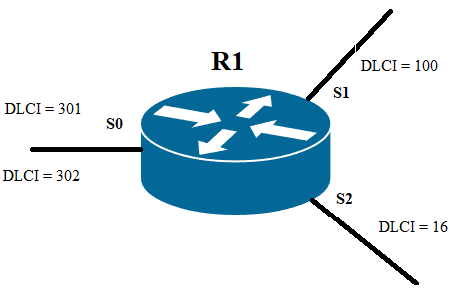


Рис. . Маршрутизатор провайдера, настроенный на Frame Relay коммутацию.

Второй технологией, позволяющей создавать в глобальной сети виртуальные каналы связи, является MPLS (Multiprotocol Label Switching - Многопротокольная коммутация по меткам). Роль адреса канального уровня играет метка. Но MPLS можно лишь условно отнести к канальному уровню, так как метка располагается межу заголовками 2-го и 3-го уровней [15]. Но принцип работы MPLS в общих чертах похож на технологию Frame Relay. Также как и у Frame Relay, имеется область из внутренних маршрутизаторов MPLS, играющих роль коммутаторов, а по периферии этой области располагаются пограничные маршрутизаторы, добавляющие/удаляющие метки. Виртуальные каналы создаются во время создания TCP сессии. Прокладываются каналы на этапе создания TCP соединения в период прохождения SYN и SYN-ACK сегментов. Удаляются каналы при прохождении FIN сообщений.

### Логика работы Ethernet коммутатора

Коммутатор умеет выполнять три типа действий с пришедшим на один из его портов фреймом:

1. пересылку (forwarding);
2. затопление (flooding);
3. фильтрацию (filtering).

В первом случае фрейм из порта прибытия перемещается в порт убытия и через него покидает коммутатор. При затоплении фрейм копируется во все активные порты и полученные копии фрейма также покидают коммутатор. В третьем случае фрейм никуда не передается и на интерфейсе прибытия уничтожается. Выбор нужного варианта действия коммутатор делает по таблице коммутации CAM (Content Addressable Memory). Данная таблица на Cisco коммутаторах выглядит так

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| Vlan | Mac Address | Type | Ports |

В данной работе виртуальные локальные сети (VLAN) не рассматриваются. Отметим лишь, что на коммутаторах по умолчанию всегда присутствует VLAN 1. Как правило тип записей в таблице динамический (DYNAMIC).

Коммутацию фреймов рассмотрим на примере небольшой сети (рис. 32), состоящей из коммутатора (switch), концентратора (hub) и четырех компьютеров (PC1 – PC4). Примечание: в современных топологиях концентраторы не используются. В нашем примере он нужен лишь для наглядности. Концентратор умеет фреймы лишь широковещательно рассылать всем подключенным устройствам, то есть выполнять операцию flooding. MAC адреса компьютеров условно обозначены как MAC1 – MAC4.

CAM таблица заполняется по MACs[[10]](#footnote-11) фреймов, отправляемых конечными устройствами. Например, фрейм отправляется компьютером PC1 и его MACs будет равен MAC1. Фрейм прибудет в порт Fa0/1 коммутатора и в CAM таблице появится запись:

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | MAC1 | DYNAMIC | Fa0/1 |

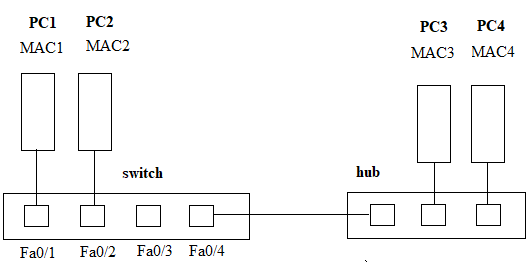


Рис. . Исходная коммутируемая сеть.

Фреймы, отправленные компьютерами PC3 и PC4, будут обнаружены со стороны порта Fa0/4, фрейм от PC2 – со стороны Fa0/2. Таким образом, таблица CAM иметь четыре записи

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| Vlan | Mac Address | Type | Ports |
| 1 | MAC1 | DYNAMIC | Fa0/1 |
| 1 | MAC2 | DYNAMIC | Fa0/2 |
| 1 | MAC3 | DYNAMIC | Fa0/4 |
| 1 | MAC4 | DYNAMIC | Fa0/4 |

Если администратор кабель от PC2 перекинет в порт Fa0/3, то первый же фрейм от компьютера внесет соответствующее изменение в таблицу, поменяв имя порта. Таким образом, CAM таблица будет всегда находиться в актуализированном состоянии, не требуя при этом вмешательства сетевого администратора. Записи в таблице хранятся в течении определенного времени и, если за это время, устройство больше не присылало свои фреймы, то запись о его MAC адресе будет из таблицы удалена.

Рассмотрим все три типа коммутации на примере нашей сети (рис. 32).

**Forwarding.**

Данный вариант коммутации будет выполнен, когда, например, фрейм будет отправлен компьютером PC1 в сторону PC2. Адресом назначения MACd[[11]](#footnote-12) будет MAC2. Фрейм от PC1 прибывает в порт Fa0/1 и его MACd в таблице CAM соответствует второй записи (табл. 11). Пересылка фрейма заключается в переброске фрейма с порта Fa0/1 на порт Fa0/2.

**Flooding.**

Затопление происходит в двух случаях: либо фрейм является широковещательным (MACd = bMAC), либо адрес назначения фрейма MACd в таблице CAM не найден. Фрейм будет доставлен всем устройствам, присоединенным к коммутатору.

**Filtering.**

Фильтрация произойдет, если, например, PC3 отправит свой фрейм PC4. Данный фрейм через концентратор (hub) будет доставлен адресату, но его копия окажется в порту Fa0/4. Поскольку MACd равен MAC4, что соответствует четвертой записи таблицы CAM, то фрейм с порта FA0/4 никуда передавать не нужно и его следует уничтожить. Фильтрация может произойти и в случае, если вычисленная контрольная сумма не совпадет с FCS[[12]](#footnote-13) трейлера фрейма. Такой вариант фильтрации возможен, если коммутатор относится к типу коммутаторов с полной буферизацией. Если коммутатор относится к типу «с коммутацией на лету», то проверка FCS не выполняется и ошибочные фреймы попадут в другие сегменты коммутируемой сети.

### Обнаружение и исправление ошибок

Выявление ошибок передачи данных по сети является одной из функций канального уровня. Алгоритмы, осуществляющие обнаружение ошибочных сообщений, реализуются на аппаратном подуровне (подуровень MAC). Для решения этой задачи перед отправкой фрейма к информационным битам добавляются контрольные биты за счет которых размер сообщения увеличивается. Добавляемые биты могут располагаться вперемежку с информационными или могут во фрейме размещены отдельно. При обнаружении факта наличия ошибки получатель может попытаться исправить ошибку, или без исправления уничтожить полученный фрейм. В Ethernet и Wi-Fi применяется алгоритм CRC32[[13]](#footnote-14), позволяющий вычислить 32 контрольных бита, которые и создают значение FCS фрейма. Более подробно о CRC32 изложено в [16].

## **Физический уровень**

### Предназначение физического уровня

Данный уровень отвечает за передачу битов через среду передачи данных (media). Биты в зависимости от типа media кодируются в радиоволны, в световые импульсы и электрические сигналы. Существуют разные варианты кодирования битов. Самым простым и очевидным является интерпретация единичного бита плюсовым напряжением (например, +1 В), а нулевого - отсутствием в сети напряжения (рис. 33). Через T обозначен битовый интервал (отрезок времени, в течение которого передается один бит информации).

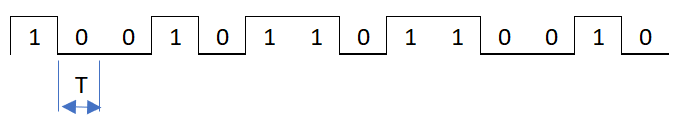


Рис. . Кодирование битов по методу NRZ[[14]](#footnote-15)

Такой вариант кодирования не является самосинхронизирующим. Какую роль играет указанное свойство у используемых кодов? Объясняется это тем, что применяемые в микросхемах портов и сетевых карт кварцевые генераторы частоты в виду их очень малых размеров трудно откалибровать по точности. И, как следствие этого, величина битового интервала T у разных микросхем различается. Если отправитель отправляет в линию связи длинную последовательность нулевых битов или, напротив, единичных битов, то получатель в виду разности значений их битовых интервалов может неправильно интерпретировать принятые биты. Например, было отправлено 10 нулевых битов, а получатель может посчитать, что ему прислали 9 или 11 нулевых битов. Поэтому важно при передаче чаще менять уровень сигнала, чтобы получатель мог правильно фиксировать начало следующего бита и выполнять синхронизацию отправки/приема битов с отправителем.

Более оптимальным является код NRZI, позволяющий передавать длинные последовательности единиц (рис. 34). Если кодируется единичный бит, то уровень сигнала меняется на противоположный, при нулевом бите уровень остается прежним. Таким образом, NRZI не обеспечивает самосинхронизацию.

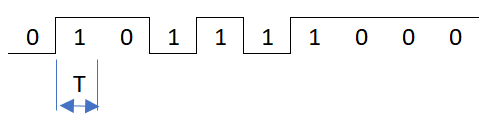


Рис. 34. Код NRZI

В манчестерском коде (рис. 35) в середине битового интервала сигнал меняет свое значение, что обеспечивает хорошую синхронизацию. К недостатку данного кода следует отнести нерациональное использование пропускной способности канала, так как по сравнению с NRZ требует удвоения частоты сигнала.

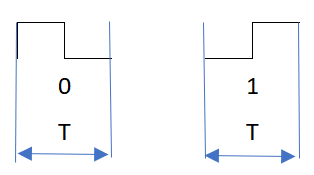


Рис. . Манчестерский код

Хорошим компромиссом между указанными выше способами является применение кода 4B/5B (рис. 36), в котором исходные четырехбитовые последовательности заменяются на пятибитовые. Этот код используется совместно с кодом NRZI, так как в отобранных для передачи пятибитовых комбинациях отсутствуют длинные последовательности нулевых битов.

В целом, завершая эту тему, отметим, что помимо рассмотренных выше примеров кодирования в сетях применяются и другие их варианты [18].

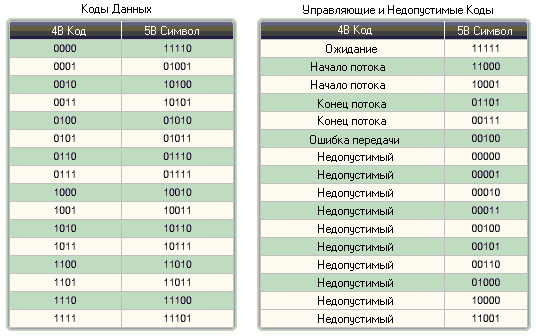


Рис. . Код 4B/5B.

### Типы обмена трафиком

Существуют три типа обмена трафиком: симплексный (рис. 37, а), полудуплексный (рис. 37, б) и дуплексный (полнодуплексный) (рис. 37, в).

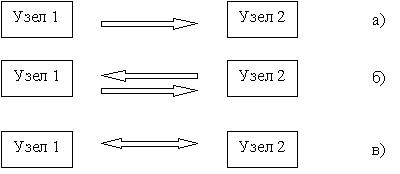


Рис. . Типы обмена трафиком

При симплексном (simplex) варианте обмена трафиком узел 1 всегда является передатчиком, узел 2 – приемником. Пример: пульт телевизора (передатчик) и сам телевизор (приемник). Полудуплексный (half duplex) режим предполагает поочередную смену ролями обеих узлов, т.е. в какой-то момент времени, например, узел 1 является передатчиком, а второй – приемником, а в следующий момент – второй становится передатчиком, а первый начинает принимать данные. Полнодуплексный (full duplex или просто duplex) вариант означает, что оба узла одновременно отправляют и получают данные. Последний вариант более производителен и, следовательно, в основном и применяется в современных компьютерных сетях.

### Классификация сред передачи данных

Среды, через которые передаются сигналы, могут быть беспроводные (wireless) и проводные (wire). К беспроводным относятся:

* Передача информации с помощью света, излучаемого в невидимом инфракрасном диапазоне. В качестве примера можно привести тот же телевизионный пульт и телевизор. Пульт в данном случае является, по сути, обычным фонарем, излучающим невидимый свет, а на лицевой панели телевизора имеется фотоприемник, принимающий свет от пульта. Недостаток такого вида передачи данных: между приемником и передатчиком не должно быть никаких непрозрачных преград.
* Передача информации с помощью радиоволн. Данный вид передачи в настоящее время интенсивно развивается. Преимущества его очевидны, так как нет необходимости прокладывать кабели, сами узлы имеют свободу перемещения без нарушения связи с другими узлами. Основной недостаток: зависимость от радиопомех. В качестве примера можно привести Wi-Fi, Bluetooth, спутниковые связь и телевидение.

Проводной вид связи классифицируется на кабели, основанные на применении медной (Cooper) жилы и оптоволокна (Fiber Optic Cable). Оптоволоконные кабели, в свою очередь, разделяются на многомодовые (рис. 38, а) и одномодовые (рис. 38, б).

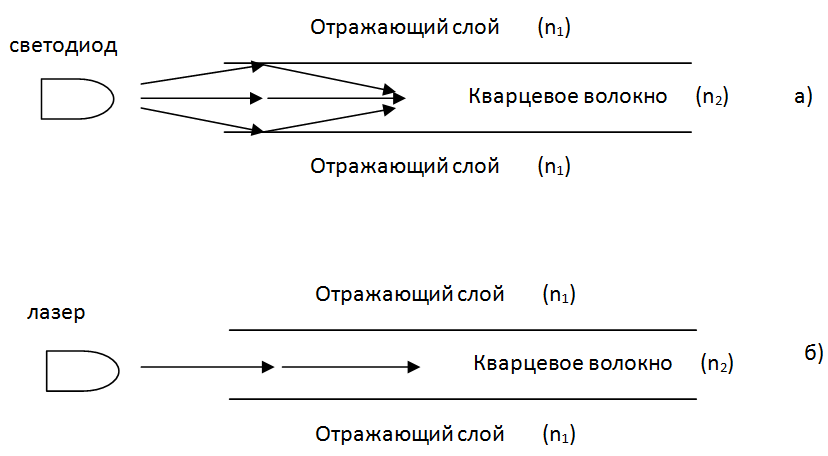


Рис. . Типы оптоволокна

Свое название волокна получили от способа распространения светового луча в них. Волокно состоит из сердцевины (кварц особо высокой чистоты) и оболочки, имеющих разные показатели преломления n1 и n2. Различие показателей преломления позволяет лучам отражаться от границы раздела обеих сред, не теряя своей мощности. В многомодовых волокнах сердцевина имеет относительно большой диаметр и, соответственно, световые фотоны перемещаются по разным траекториям, образуя разные моды. В одномодовом волокне диаметр сердцевины сопоставим с длиной световой волны, и фотоны движутся в одном направлении.

Источниками света в оптоволоконных кабелях являются:

* светодиоды, генерирующие инфракрасный свет с длиной волны 850 или 1310 нм. Светодиоды используются для передачи сигналов по многомодовому волокну на расстояние до 2000 м. Применяются в локальных сетях;
* лазерные диоды, генерирующие инфракрасный луч света с длиной волны 1310 или 1550 нм. Лазеры, обладая более высокой мощностью излучения, чем светодиоды, используются совместно с одномодовым волокном для передачи сигналов на большие расстояния в локальных и глобальных сетях (магистральные каналы связи, Internet backbone).

Медные кабели получили применение лишь в локальных сетях (LAN). В ранних версиях LAN применялась такая разновидность медных кабелей, как коаксиальный кабель (Coaxial Cable) рис. 39, а). Существовало две его модификации: тонкий (Thin) и толстый (Thick) коаксиал.

Коаксиал похож на обычный телевизионный кабель и обеспечивает лишь полудуплексный способ передачи данных. Сети, созданные на основе коаксиала, имели шинную топологию. На данный момент коаксиальный кабель используется редко, а наибольшее распространение получила витая пара (Twisted Pair, TP).

Витая пара состоит из изолированных жил медной проволоки, скрученных вместе. Есть две версии витой пары: неэкранированная пара (Unshielded Twisted Pair, UTP) (рис. 39, б) и экранированная (Shielded Twisted Pair, STP) (рис. 39, в). Экранированная пара имеет дополнительную защиту от внешних помех в виде экрана из фольги и дороже неэкранированной.



Рис. . Медные кабели.

Также витая пара различается по категориям: Cat.5, Cat.5e, Cat.6, Cat.6a, Cat.7 и Cat.7a. Категории зависят от степени скрутки пар, наличия или отсутствия экрана, скорости передачи данных. Максимальная длина используемого в сети кабеля должна быть не более 100 метров.

Первый вариант витой пары (рис. 40, а) применялся в телефонных сетях. Два провода позволяли создать замкнутый электрический контур (рис. 40, б) между двумя устройствами. Свивка проводов позволяет уменьшить влияние внешних радиопомех на передаваемые сигналы. Позже появился двупарный кабель (рис. 40, в), в котором были сохранены позиции кабелей первой пары для обеспечения совместимости с первым вариантом. В современных компьютерных сетях чаще применяется 4-парный вариант витой пары (рис. 40, г). Пины являются частью коннекторов RJ-45, предназначены для физического контакта с проводами кабеля.

Существуют разные схемы обжима витой пары в коннекторе RJ-45 [17]:

1. прямой;
2. кроссоверный;
3. ролловерный.

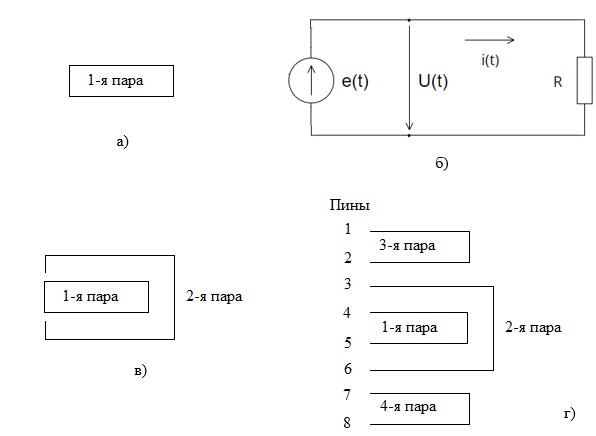


Рис. 40. Электрический контур и типы витых пар.

Прямой обжим является основным. Первый провод соединяет 1-й пин одного коннектора с 1-м пином второго коннектора. Остальные 7 проводов соединены по аналогичной схеме: 2-2, 3-3, 4-4, 5-5, 6-6, 7-7, 8-8.

Ролловерный вариант используется в консольных кабелях. В такой схеме обжима провод, присоединенный к 1-му пину одного коннектора, на втором коннекторе присоединен к 8-му пину. Другие 7 проводов соединены по схеме 2-7, 3-6, 4-5, 5-4, 6-3, 7-2 и 8-1.

Кроссоверный обжим в настоящее время не применяется. Согласно этой схемы обжима, 3-я пара одного коннектора на втором коннекторе интерпретируется как 2-я, и, наоборот, 2-я превращается во 3-ю. Использование кроссоверного варианта витой пары было обусловлено тем обстоятельством, что 3-я пара используется для отправки сообщения (фрейма), а 2-я - для его приема (рис. 41). Такая схема обжима позволяет обеспечить полнодуплексный режим обмена фреймами через витую пару. Из рисунка видно, что провода соединяют 1-й пин первого коннектора с 3-м пином второго, 2-й пин – с 6-м, 3-й – с 1-м и 6-й – со 2-м. Такую схему условно назовем (1-3, 2-6).

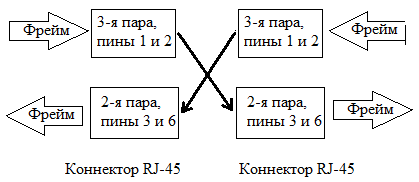


Рис. 41. Реализация полнодуплексного режима в витой паре

1-я и 4-я пары в FastEthernet (100 Mbps) технологии не используются, но в PoE (Power over Ethernet) используются для подачи напряжения конечным устройствам, например, к камерам наблюдения.

В каких случаях применяются кроссоверные и в каких прямые кабели? Сетевые устройства можно разделить на два типа. В портах устройств первого и второго уровней (hub и switch) скрутка (1-3, 2-6) уже выполнена. Ко второму типу относятся маршрутизаторы и компьютеры, в интерфейсах которых такой скрутки нет. Это означает, что при соединении устройств разных типов следует использовать прямой кабель, если же устройства относятся к одному типу – то кроссоверный кабель. Например, компьютер к коммутатору следует присоединять прямым кабелем, компьютер к маршрутизатору или два коммутатора между собой – кроссоверным кабелем. Впоследствии была разработана спецификация, согласно которой в сетевых картах и портах была реализована функция, позволяющая устройствам определять тип соединяющего их кабеля и соответствующим образом перестраивать свой режим работы с соединением. Таким образом, необходимость в кроссоверных кабелях отпала. Но если в сети используются устройства старого типа, в которых данная функция не поддерживается, то следует выбирать тип витой пары следуя указанным выше рекомендациям.

# Сетевые устройства Cisco

Более подробную информацию об американской компании Cisco можно получить со страницы Википедии [14]. Имеет разветвленную сеть академий, позволяющих получить признанные в отрасли профессиональные сертификаты «Сертифицированный Cisco сетевой специалист» (CCNA) и «Сертифицированный Cisco сетевой профессионал» (CCNP). Программы академии Cisco относятся к одним из лучших стандартов методик преподавания основ сетевых технологий и реализации дистанционного обучения. Проверка умений работы с сетевым оборудованием Cisco входит в компетенцию «Сетевое и системное администрирование» движения WorldSkills [19].

Операционная система у коммутаторов и маршрутизаторов Cisco называется одинаково: IOS. Хранится IOS во flash памяти. Настройки представляют собой макрос, хранимый в памяти NVRAM в виде файла startup-config. Этот же файл в оперативной (RAM) памяти обозначается как – running-config. Еще один вид памяти – ROM[[15]](#footnote-16) предназначен для хранения программ самотестирования POST (Power-On Self Test) и начальной загрузки (Bootstrap), значения конфигурационного регистра и т.д.

IOS предусматривает работу в разных режимах. Приглашение командной строки ко вводу ваших команд в разных режимах различается (табл. 14).

Таблица 14.

|  |  |
| --- | --- |
| Приглашение командной строки | Режим работы |
| Router> | Пользовательский режим |
| Router# | Привилегированный режим |
| Router(config)# | Режим глобального конфигурирования |
| Router(config-if)# | Режим конфигурирования интерфейса |
| Router(config-line)# | Режим конфигурирования консольной и виртуальных линий |

В каждом режиме предусмотрен свой набор команд. Вызов помощи по командам с помощью ввода символа «?». Если какая-либо команда не распознается, то возможно делается попытка ее выполнения в режиме, в котором ее выполнение не предусмотрено. Команды привилегированного режима выполняются из других режимов вводом:

do <команда привилегированного режима>

Пользовательский режим считается относительно безопасным режимом, так как команды, допустимые в этом режиме, не позволяют просмотреть настройки устройства и, тем более, их изменить. В привилегированный режим посторонних допускать крайне не рекомендуется, так здесь доступна команда show, позволяющая просмотреть настройки и получить представление о самой сети, в которой находится устройство. Из этого режима имеется возможность перейти в следующий режим, а именно в режим глобального конфигурирования с последующим переходом во все другие режимы, что позволит выполнять несанкционированное изменение всех настроек данного устройства. Настоятельно рекомендуется защищать паролем сам консольный доступ к устройству, удаленный доступ через сеть и также установить парольную защиту на сам переход из пользовательского режима в привилегированный. Также одним из методов защиты является использование предупреждающих сообщений[[16]](#footnote-17), что может впоследствии служить обоснованием для вынесения наказания за игнорирование запрета.

# Моделирование компьютерных сетей

Отметим, что для изучения сетевых технологий используются симуляторы и эмуляторы. Симуляторы позволяют создать модель компьютерной сети. Эмуляторы отвечают за аппаратную часть. Например, известен эмулятор Dynamips, позволяющий на компьютере, на котором он запущен, создать маршрутизатор Cisco. Но для запуска такого маршрутизатора необходим образ ОС IOS. С помощью эмуляторов можно ознакомиться с возможностями реальной ОС. Но запуск одновременно нескольких маршрутизаторов на компьютере возможен, если у него хорошая память и мощный процессор. По этой причине в изучении сетевых технологий используются симуляторы, которые менее требовательны к ресурсам компьютера. Одним из популярных симуляторов является Cisco Packet Tracer (сокращенно - CPT). Cisco разработала данную программу для нужд своей

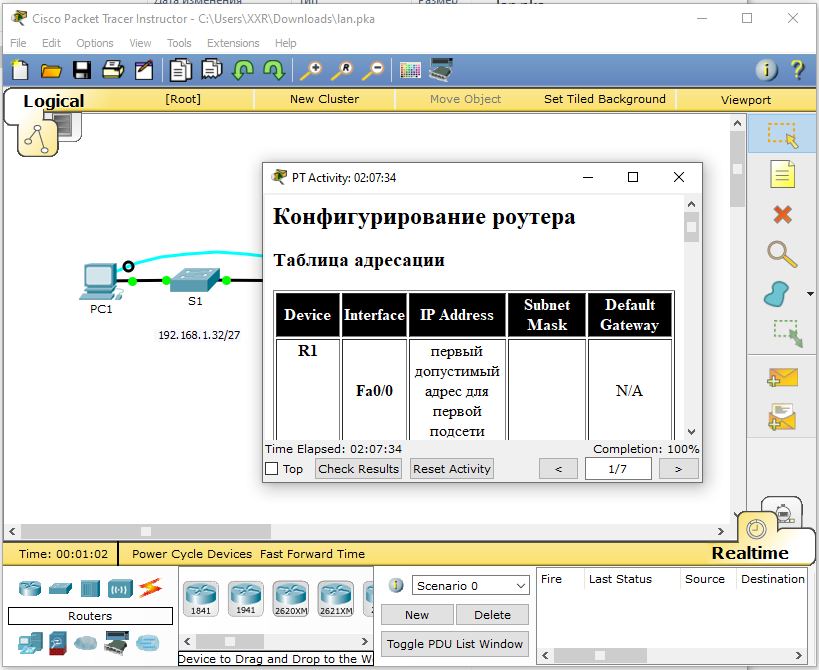


Рис. 42. Pka файл, открытый в CPT[[17]](#footnote-18).

сетевой академии, но дистрибутив симулятора может скачать любой желающий [20], необязательно являющийся слушателем академии. Общие сведения о приемах работы с данным симулятором сети можно узнать в [21].

CPT позволяет работать с двумя типами файлов: pkt и pka. Pka файлы применяются для проверки выполнения задания по моделированию сети. CPT при открытии pka файла создает два окна (рис. 42). В окне Activity размещается текст задания, в правом нижнем углу окна указывается процент выполнения задания. С помощью кнопки Check Results можно проверить какие пункты задания выполнены и какие нет. Второе окно – окно модели сети.

## **Базовая настройка устройств Cisco**

Базовая настройка проводится с помощью консольного кабеля и терминальной программы. Заключается в наделении устройства его сетевым идентификатором, IP конфигурировании его интерфейсов и обеспечении защиты от несанкционированного доступа извне на данное устройство.

### Маршрутизатор

Рассмотрим вначале настройку маршрутизатора[[18]](#footnote-19). На начальном этапе после загрузки в RAM операционной системы появится приглашение входа в режим диалога для настройки:

**--- System Configuration Dialog ---**

**Continue with configuration dialog? [yes/no]:**

Следует отказываться от диалога, вводя «n» и нажав «Enter». Далее переходим в привилегированный режим командой enable (en – сокращенный вариант данной команды):

**Press RETURN to get started!**

**Router>en**

Переход в режим изменения настроек:

**Router#conf t**

Изменение системного имени

**Router(config)#hostname R2**

Ввод маршрутизатора в домен esstu.ru

**R2(config)#ip domain-name esstu.ru**

Назначение пароля class на команду enable[[19]](#footnote-20)

**R2(config)#enable secret class**

Назначение пароля cisco на консольное подключение и виртуальные линии[[20]](#footnote-21)

**R2(config)#line con 0**

**R2(config-line)#password cisco**

**R2(config-line)#login**

**R2(config-line)#line vty 0 4**

**R2(config-line)#password cisco**

**R2(config-line)#login**

**R2(config-line)#exit**

Шифрование паролей[[21]](#footnote-22)

**R2(config)#service password-encryption**

IP конфигурирование интерфейсов

**R2(config)#interface FastEthernet0/0**

**R2(config-if)#ip address 180.119.21.129 255.255.255.248**

**R2(config-if)#no shutdown**

**R2(config-if)#interface Serial2/0**

**R2(config-if)#ip address 180.119.21.1 255.255.255.252**

**R2(config-if)#clock rate 1000000**

**R2(config-if)#no shutdown**

**R2(config-if)#interface Serial3/0**

**R2(config-if)#ip address 10.238.0.5 255.255.255.252**

**R2(config-if)#clock rate 800000**

**R2(config-if)#no shutdown**

**R2(config-if)#exit**

Пояснение по команде **clock rate**: на **FastEthernet** интерфейсах пропускная способность имеет значение 100 Mbps, а на **Serial** интерфейсах его нужно задавать. В данном примере на **Serial2/0** ее указали равной 1 Mbps, на Serial**3/0** – 0,8 Mbps. Сам Serial кабель соединяет порты двух типов: DTE и DCE. Команда **clock rate** подается на DCE интерфейсе.

Сохранение конфигурационного файла на внешнюю память

**R2(config)#exit**

**R2#**

**%SYS-5-CONFIG\_I: Configured from console by console**

**R2#write**

Проверка конфигурационных файлов в RAM и NVRAM

**R2#sh run**

<вывод пропущен>

**R2#sh start**

<вывод пропущен>

### Коммутатор

Базовые команды настройки коммутатора в целом похожи на команды, подаваемые на маршрутизаторе:

**switch>en**

**switch#conf t**

**switch(config)#hostname S5**

**S5(config)#ip domain-name esstu.ru**

**S5(config)#enable secret class**

**S5(config)#line con 0**

**S5(config-line)#password cisco**

**S5(config-line)#login**

**S5(config-line)#line vty 0 15**

**S5(config-line)#password cisco**

**S5(config-line)#login**

**S5(config-line)#exit**

**S5(config)#service password-encryption**

Интерфейсы (порты) коммутатора не получают IP адреса, но на виртуальном интерфейсе (чаще всего это интерфейс VLAN 1) назначается адрес. Также необходимо указать шлюз по умолчанию (default-gateway)[[22]](#footnote-23):

**S5(config)#interface Vlan1**

**S5(config-if)#ip address 10.236.0.2 255.255.0.0**

**S5(config-if)#no shutdown**

**S5(config-if)#exit**

**S5(config)#ip default-gateway 10.236.0.1**

Сохранение файла

**S5(config)#exit**

**S5#write**

## **Маршрутизация**

В данном разделе рассмотрим основную логику работы маршрутизатора. Его работа основана на использовании таблицы маршрутизации (routing table). Просмотреть таблицу можно командой

**R2#show ip route**

Данная таблица содержит IP адрес сети с ее маской и либо свой выходной интерфейс, либо адрес соседнего маршрутизатора, через который данная сеть доступна. Пояснение: основная функция маршрутизатора – это пересылка поступающих на его интерфейсы[[23]](#footnote-24) пакетов на соседние устройства: маршрутизаторы, коммутаторы и компьютеры. Источником для принятия решения является указанная таблица и IPd[[24]](#footnote-25) поступившего пакета. В правильно настроенной сети маршрутизаторы должны иметь маршруты ко всем сетям.

Прежде чем рассмотреть логику работы маршрутизатора, разберем различия межу коммутацией фреймов и маршрутизацией пакетов (рис. 43).

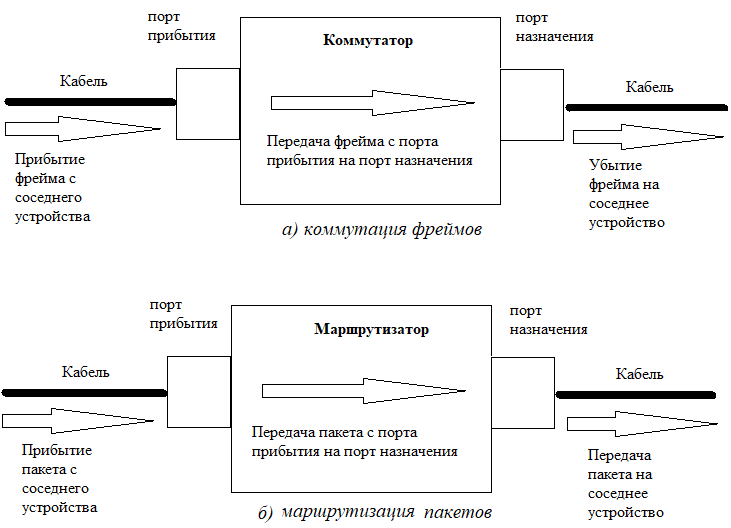


Рис. 43. Сравнение коммутации и маршрутизации.

При коммутации пакет, который находится внутри фрейма (рис. 43, а), пределы сети не покидает. Оба интерфейса коммутатора (и порт прибытия и порт назначения) относятся к одной и той же сети. При маршрутизации (рис. 43, б) пакет, перемещаясь с порта прибытия на порт назначения, переходит из одной сети в другую, ибо интерфейсы маршрутизатора относятся к разным сетям.

### Таблицы маршрутизации

Принцип работы маршрутизатора разберем на примере модели небольшой сети (рис. 44). Так как в сети присутствуют два маршрутизатора, то в нашей топологии присутствуют три подсети (табл. 15). Подсети получены из классовой сети 192.168.1.0/24 и их значения меняются с шагом 32. Каждая сеть содержит по два хоста, адреса которых приняты либо минимальны, либо максимальны для своей подсети.

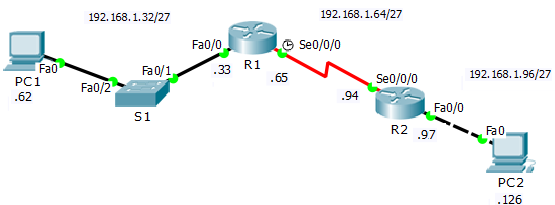


Рис. . Модель сети

Таблица 15

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес подсети | Широко-вещательный адрес подсети | Минимальный хостовый адрес | Максимальный хостовый адрес | Хосты подсети |
| 192.168.1.32 | 192.168.1.63 | 192.168.1.33 | 192.168.1.62 | PC1, порт Fa0/0 маршрутизатора R1 |
| 192.168.1.64 | 192.168.1.95 | 192.168.1.65 | 192.168.1.94 | порт Se0/0/0 маршрутизатора R1, порт Se0/0/0 маршрутизатора R2 |
| 192.168.1.96 | 192.168.1.127 | 192.168.1.97 | 192.168.1.126 | PC2, порт Fa0/0 маршрутизатора R2 |

Настройки портов маршрутизатора R1 имеют вид

**interface FastEthernet0/0**

**description R1 LAN**

**ip address 192.168.1.33 255.255.255.224**

**interface Serial0/0/0**

**description Link to R2**

**ip address 192.168.1.65 255.255.255.224**

Просмотр таблицы маршрутизации R1:

**R1#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 not set**

**192.168.1.0/27 is subnetted, 2 subnets**

**C 192.168.1.32 is directly connected, FastEthernet0/0**

**C 192.168.1.64 is directly connected, Serial0/0/0**

Интерес представляют три нижние строки вывода команды **show**. В них сообщается, что классовая сеть **192.168.1.0** разбита на подсети с маской **/27**, со стороны интерфейса **FastEthernet0/0** присутствует подсеть **192.168.1.32,** со стороны интерфейса **Serial0/0/0 - 192.168.1.64.** Символ «**C»**, присутствующий в начале двух маршрутов, обозначает connected (присоединенная сеть). Адреса сетей вычисляются побитовым перемножением адреса интерфейса на его маску.

Как видим, маршрутизатор R1 не знает о существовании сети **192.168.1.96/27**, которая для него является удаленной («спрятанной» за соседним маршрутизатором R2). Для обеспечения маршрутизаторов информацией о полной топологии сети применяются статическая и динамическая маршрутизация. В первом случае маршруты в удаленные сети администратор прописывает командой

**R1(config)#ip route …**

В нашем случае эта команда будет иметь вид

**R1(config)# ip route 192.168.1.96 255.255.255.224 192.168.1.94**

Другой равноценный вариант этой команды

**R1(config)# ip route 192.168.1.96 255.255.255.224 Serial0/0/0**

Пояснение: указываем адрес удаленной сети, его маску и либо адрес соседа (**192.168.1.94**), либо выходной интерфейс (**Serial0/0/0**), обращенный в сторону данного соседа. В обеих случаях в таблице маршрутизации появится запись об этой сети

**S 192.168.1.96 [1/0] via 192.168.1.94**

для первого варианта команды **ip route** или в виде

**S 192.168.1.96 is directly connected, Serial0/0/0**

для второго варианта.

Второй маршрутизатор (R2) рассматривать не будем, там все будет аналогично. Разница будет лишь в том, что он не будет знать о первой сети (**192.168.1.32/27**) и на нем необходимо будет прописать маршрут в эту сеть.

Рассмотрим, как маршрутизаторы позволят сообщению переместиться от PC1 до PC2. Для проверки связности сети пустим пинг с PC1 (рис. 45).

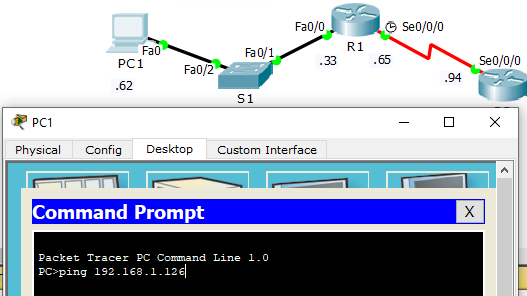


Рис. 45. Командная строка PC1.

В данном примере в качестве адреса пинга указан адрес PC2 **192.168.1.126**. Сообщение, сформированное программой ping, через коммутатор S1 будет доставлено на интерфейс **FastEthernet0/0** маршрутизатора. Из его заголовка извлекается hIPd = **192.168.1.126** и по нему проверяется маршруты из table routing:

**192.168.1.0/27 is subnetted, 3 subnets**

**C 192.168.1.32 is directly connected, FastEthernet0/0**

**C 192.168.1.64 is directly connected, Serial0/0/0**

**S 192.168.1.96 is directly connected, Serial0/0/0**

По маске /27 и по hIPd вычисляется nIPd (адрес сети назначения). Побитовое умножение дает nIPd равным **192.168.1.96.** Этому адресу соответствует третий маршрут

**S 192.168.1.96 is directly connected, Serial0/0/0**

который привязан к интерфейсу **Serial0/0/0.** Следовательно пакет будет перемещен с интерфейса **FastEthernet0/0** на интерфейс **Serial0/0/0** и по кабелю будет доставлен на соседний маршрутизатор (R2). R2 будет действовать по аналогичной схеме и передаст пакет с пингом компьютеру PC2. PC2 обязан будет ответить обратным echo откликом, который вернется на PC1 (рис. 46).

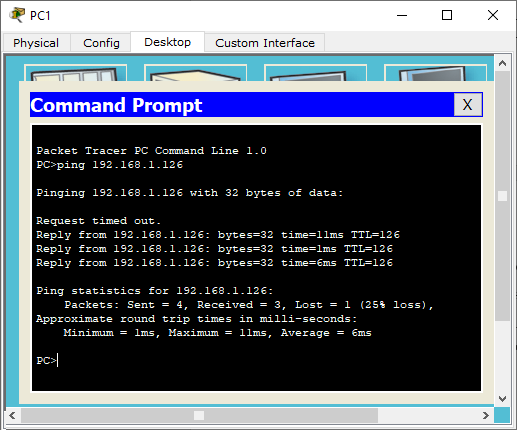


Рис. 46. PC1 и PC2 способны друг другу отправлять сообщения. Коннективность сети обеспечена.

Что произойдет с пакетом на R2, если с PC1 будет отправлен пинг на несуществующий адрес, например, на **192.168.1.129**? R1 определит nIPd, как равный **192.168.1.128** и пакет уничтожит на самом порту прибытия (**FastEthernet0/0**), так для данной сети в таблице нет маршрута. Таким образом, маршрутизатор отфильтровывает все пакеты, для которых маршруты у него отсутствуют.

### Маршрут по умолчанию

В рассмотренных выше таблицах маршрутизации была одна примечательная строка в виде

**Gateway of last resort is not set**

которую можно перевести как «Шлюз последней инстанции не установлен». В рассмотренном выше примере сеть полностью изолирована от внешнего мира. В реальных сетях, как правило у хостов должен быть выход в интернет. А интернет характерен большим количеством адресов[[25]](#footnote-26). Адреса, на которые внутренний хост будет отправлять свой запрос, могут быть любые. Следовательно, на маршрутизаторах должен присутствовать маршрут **Gateway of last resort**, направленный в сторону маршрутизатора, за которым находится интернет с его огромным разнообразием адресов. Такой маршрут имеет адрес **0.0.0.0/0**, условно такой адрес можно назвать адрес «все сети».

Изменим топологию сети, добавив в него еще один маршрутизатор (R3) и создав на нем loopback интерфейс с адресом 8.8.8.8/32 символизирующим некий сервер, расположенный в интернете.

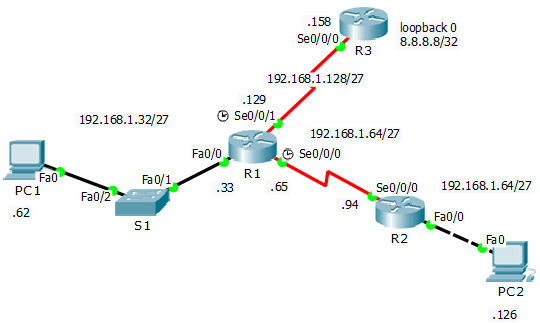


Рис. 47. Обеспечение выхода в интернет.

Линк, соединяющий R1 и R3, получил адрес 192.168.1.128/27. На R1 прописываем дополнительные команды

**R1(config)#int s 0/0/1**

**R1(config-if)#ip add 192.168.1.129 255.255.255.224**

**R1(config-if)#clock rate 64000**

**R1(config-if)#no sh**

**R1(config-if)#exit**

**R1(config)#ip route 0.0.0.0 0.0.0.0 s0/0/1**

На R3 команды

**R3(config)#interface Serial0/0/0**

**R3(config-if)#ip address 192.168.1.158 255.255.255.224**

**R3(config-if)#no sh**

**%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up**

**%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up**

**R3(config-if)#int lo0**

**R3(config-if)#**

**%LINK-5-CHANGED: Interface Loopback0, changed state to up**

**%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up**

**R3(config-if)#ip add 8.8.8.8 255.255.255.255**

**R3(config-if)#exit**

**R3(config)#ip route 192.168.1.32 255.255.255.224 s0/0/0**

**R3(config)#ip route 192.168.1.64 255.255.255.224 s0/0/0**

**R3(config)#ip route 192.168.1.96 255.255.255.224 s0/0/0**

На R1 таблица маршрутов примет вид

**R1#sh ip route**

**<часть текста вывода пропущена>**

**Gateway of last resort is 0.0.0.0 to network 0.0.0.0**

**192.168.1.0/27 is subnetted, 4 subnets**

**C 192.168.1.32 is directly connected, FastEthernet0/0**

**C 192.168.1.64 is directly connected, Serial0/0/0**

**S 192.168.1.96 is directly connected, Serial0/0/0**

**C 192.168.1.128 is directly connected, Serial0/0/1**

**S\* 0.0.0.0/0 is directly connected, Serial0/0/1**

Запись

**Gateway of last resort is 0.0.0.0 to network 0.0.0.0**

сообщает, что маршрут по умолчанию имеется. На R3 таблица имеет вид

**R3#sh ip route**

**<часть текста вывода пропущена>**

**Gateway of last resort is not set**

**8.0.0.0/32 is subnetted, 1 subnets**

**C 8.8.8.8 is directly connected, Loopback0**

**192.168.1.0/27 is subnetted, 4 subnets**

**S 192.168.1.32 is directly connected, Serial0/0/0**

**S 192.168.1.64 is directly connected, Serial0/0/0**

**S 192.168.1.96 is directly connected, Serial0/0/0**

**C 192.168.1.128 is directly connected, Serial0/0/0**

Пинг, отправленный с PC1, подтверждает доступность интернета

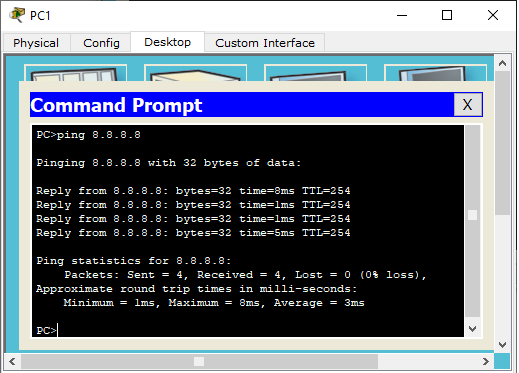


Рис. 48. Подтверждение доступности интернета.

Как в данном случае сработал маршрут по умолчанию на R1?

**S\* 0.0.0.0/0 is directly connected, Serial0/0/1**

Когда сообщение от PC1 попало на интерфейс **FastEthernet0/0**, то из его заголовка был извлечен nIPd = **8.8.8.8**. Маршруты с маской /27 не подходят для сообщения. А при проверке маршрута по умолчанию его маска /0 умножается на **8.8.8.8**, что даст nIPd = **0.0.0.0**, что дает совпадение с самим маршрутом. Пакет был перенаправлен на интерфейс **Serial0/0/1** и, далее, на маршрутизатор R3. Вывод: маршрут по умолчанию подходит для любого пакета с любым адресом назначения.

Как будет проходить теперь пакет от PC1 до PC2, ведь для него подходит два маршрута

**S 192.168.1.96 is directly connected, Serial0/0/0**

и

**S\* 0.0.0.0/0 is directly connected, Serial0/0/1**

Здесь срабатывает правило «наидлиннейшей маски», которая позволит выбрать первый маршрут, т.е. пакет уйдет в сторону маршрутизатора R2, а не R3.

Ссылка на pkt файл рассмотренного в данной теме примера: http://altaev-aa.narod.ru/net/lan\_.pkt .

### Динамическая маршрутизация

В данном варианте внесения маршрутов в table routing на маршрутизаторах запускаются программы, позволяющие им обменяться между собой маршрутной информацией. В указанных программах реализованы протоколы маршрутизации, позволяющие маршрутизаторам узнать обо все сетях, присутствующих в топологии, и построить маршруты к ним. Достоинство такого подхода заключается в том, что любое изменение топологии сети будет своевременно зафиксировано и при необходимости маршруты к удаленным сетям будут перестроены. Подробно эта тема в пособии не задета, лишь отметим, что в курсовой работе используются два протокола: RIP (Routing Information Protocol – Протокол маршрутной информации) и OSPF (Open Shortest Path First – Протокол выбора кратчайшего пути на основе анализа состояния линка). Принципы работы у указанных протоколов разные, но есть одинаковая для них команда network, позволяющая указать пулы адресов. Если у интерфейса маршрутизатора IP адрес попадет в указанный пул, то он будет включен в процесс протокола. Адрес сети с маской, в которой находится интерфейс, его пропускная способность (bandwidth) будут переданы другим маршрутизаторам, через него начнут приходить и уходить служебные сообщения протокола.

# Задание

1. Скачайте заготовку сети, используя ссылку http://altaev-aa.narod.ru/net/blank.pka. В скачанном pka файле постройте модель своей сети (рис. 49, табл. 16).
2. Выполните базовую настройку маршрутизаторов (системные имена должны совпадать надписями на схеме, enable пароль принять class, консольный и telnet пароли - cisco).
3. Добавьте в приватную и публичную сети по одному серверу. На сервере приватной сети настройте сайт организации, на сервере публичной сети свой авторский сайт.
4. Проведите IP планирование сети (табл. 2), применяя технологии VLSM и CIDR. Броадкастовые подсети должны иметь максимальные размеры (использовать короткую длину маски), двухточечные линки, напротив, должны иметь маску /30. С помощью кнопки Place Note на схеме укажите адреса подсетей и их маски. Настройте интерфейсы маршрутизаторов. Serial линки должны иметь разную пропускную способность. На схеме укажите, у какого линка какая принята пропускная способность. Коммутаторы должны получить IP адреса и адрес основного шлюза.
5. Для каждой броадкастовой подсети создайте DHCP серверы. Исключите из раздачи адреса коммутаторов и серверов. Добавив в топологию рабочие станции (PC), проверьте доступность DHCP-сервера. Убедитесь, что DHCP сервер не выдает сообщения о конфликте адресов.
6. Настройте статическую и динамическую маршрутизации. Для внутренней сети используйте протокол RIP, для внешней OSPF. Внутренние узлы должны «видеть» внешнюю сеть, для чего на пограничном маршрутизаторе R2 настройте маршрут по умолчанию и распространите его по RIP сети. RIP сеть свои маршруты во внешнюю сеть не должна анонсировать. Командой show ip route проверьте содержимое таблиц маршрутизации. OSPF должен знать пропускную способность serial линков. Примените для этого команду bandwidth на serial интерфейсах. Пингами проверьте используются ли среди serial более быстрые линки.
7. В public сети создайте DNS сервер и внесите в его базу данных записи о сайте организации и авторском сайте. Сайт организации должен иметь public адрес пограничного маршрутизатора.
8. На маршрутизаторе R2 реализуйте NAT. Обеспечьте видимость из public сети сайта организации.

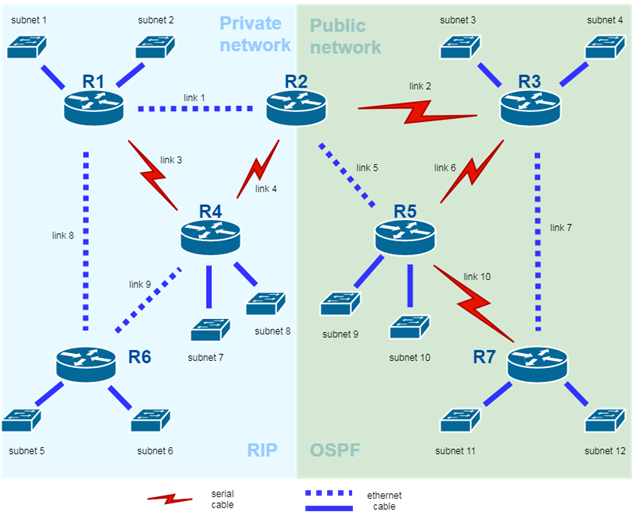


Рис. . Схема сети для выбора вариантов

1. Оформите отчет согласно следующим требованиям:
   1. Привести данные своего варианта;
   2. Скрин топологии сети;
   3. Расчеты адреса суперсетей;
   4. Конфигурационные файлы сетевых устройств;
   5. Таблицы маршрутизации;
   6. Скрины результатов тестирования пунктов 5-8.

Таблица . Топология сети

| № вар. | Двухточечные подсети (links) | | | | | | | | | | Броадкастовые подсети (subnets) | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| 1 | X | X | X | X |  | X | X |  | X |  |  |  |  | X | X |  | X | X |  |  | X |  |
| 2 | X | X | X |  | X | X |  | X |  | X |  |  |  |  | X | X |  |  |  |  | X | X |
| 3 |  |  |  | X | X | X |  | X | X | X | X |  | X |  | X |  |  |  |  | X |  | X |
| 4 | X | X |  |  |  | X | X | X | X |  |  | X |  | X | X |  |  |  |  | X | X |  |
| 5 | X |  | X |  | X | X |  | X |  | X |  |  |  |  | X | X |  |  | X |  | X | X |
| 6 | X |  |  |  | X |  | X | X | X | X |  |  |  | X | X |  | X |  | X |  | X |  |
| 7 | X | X | X |  |  | X | X |  | X |  |  |  | X |  | X |  | X | X | X |  |  | X |
| 8 |  |  | X | X | X | X | X |  | X |  | X | X |  |  | X |  |  | X |  |  | X | X |
| 9 | X | X | X |  |  | X |  | X |  | X | X |  |  | X | X |  |  |  | X |  | X |  |
| 10 | X |  | X | X | X | X | X |  | X |  |  |  |  | X | X |  | X |  |  | X | X |  |
| 11 | X |  | X |  | X | X |  |  | X | X | X | X |  | X | X |  |  |  |  | X | X |  |
| 12 |  |  |  | X | X | X | X | X | X |  |  |  | X | X | X |  | X |  | X | X |  |  |
| 13 |  |  | X | X | X |  | X | X |  | X |  |  |  | X | X |  | X |  | X | X | X |  |
| 14 | X |  |  |  | X | X |  | X | X | X | X |  | X |  | X |  |  |  |  |  |  | X |
| 15 |  | X |  | X |  | X | X | X | X |  |  |  | X | X | X |  | X |  |  |  |  |  |
| 16 |  | X |  | X |  | X | X | X | X |  |  |  |  | X | X |  | X | X | X |  | X |  |
| 17 |  | X |  | X | X |  | X | X | X |  | X |  |  | X | X |  | X | X |  |  | X |  |
| 18 | X | X | X |  | X |  | X |  | X |  |  | X | X | X | X |  |  |  | X |  |  |  |
| 19 |  | X |  | X |  |  | X | X | X | X |  |  |  | X | X |  | X |  |  |  | X |  |
| 20 | X | X | X | X |  |  | X | X |  | X |  |  |  | X | X |  | X |  |  |  | X |  |
| 21 | X |  | X |  | X | X |  |  | X | X |  |  |  | X | X |  | X | X |  | X | X |  |
| 22 | X | X | X | X |  |  | X |  | X | X |  |  |  |  | X | X |  |  |  |  | X | X |
| 23 | X |  | X |  | X |  | X |  | X | X |  |  |  | X | X |  | X |  |  | X | X |  |
| 24 | X | X | X |  |  |  | X |  | X | X | X | X |  | X | X |  |  |  |  | X | X |  |
| 25 |  | X | X | X |  |  | X | X |  | X |  |  |  | X | X |  | X |  |  | X | X |  |
| 26 |  | X | X | X |  |  | X | X |  | X |  | X |  | X | X |  |  |  | X |  | X |  |
| 27 |  |  | X | X | X | X | X |  | X |  | X |  | X | X | X |  | X |  | X |  |  |  |
| 28 | X | X | X |  |  |  | X | X |  | X |  |  | X | X | X |  | X |  |  |  |  |  |
| 29 | X |  | X | X | X | X |  |  | X | X | X |  | X |  | X |  |  |  | X | X |  | X |
| 30 | X |  |  | X | X |  | X | X |  | X |  |  |  |  | X |  | X |  |  |  | X | X |
| 31 |  |  |  | X | X |  | X | X | X | X |  |  |  | X | X |  | X |  |  |  | X |  |
| 32 |  | X |  | X | X |  | X | X | X |  |  | X | X |  | X |  |  |  | X |  |  | X |
| 33 | X | X | X | X |  | X | X |  | X |  |  |  |  |  | X |  | X | X | X | X | X | X |
| 34 |  |  | X | X | X | X |  | X |  | X |  |  |  | X | X |  | X |  |  |  | X |  |
| 35 | X | X |  |  |  |  | X | X | X | X |  |  |  | X | X |  | X |  | X | X | X |  |
| 36 | X |  | X |  | X | X |  | X |  | X |  |  |  | X | X |  | X |  |  |  | X |  |
| 37 |  |  | X | X | X |  | X |  | X | X |  |  | X | X | X |  | X |  |  |  |  |  |
| 38 | X | X |  |  |  | X |  | X | X | X |  |  |  | X | X |  | X |  |  | X | X |  |
| 39 |  | X | X | X |  | X | X |  | X |  | X |  |  | X | X |  |  |  | X | X | X |  |
| 40 | X | X | X | X |  |  | X | X |  | X |  | X |  |  | X |  |  |  | X |  | X | X |
| 41 | X | X | X |  | X | X | X |  | X |  |  |  |  | X | X |  | X |  | X |  | X |  |
| 42 | X | X |  | X | X | X |  |  | X | X |  |  | X |  | X |  | X |  | X |  |  | X |
| 43 | X |  | X |  | X | X |  | X |  | X |  |  | X | X | X | X | X | X |  |  |  |  |
| 44 | X | X |  |  |  | X | X | X | X |  |  |  | X |  | X |  | X |  | X | X |  | X |
| 45 |  | X | X | X | X | X | X | X |  |  | X |  |  | X | X |  |  |  |  |  | X |  |
| 46 | X | X |  | X | X | X | X | X |  |  |  |  | X | X | X |  | X |  | X |  |  |  |
| 47 |  |  | X | X | X | X | X | X |  |  | X |  | X | X | X |  |  |  | X |  |  |  |
| 48 | X | X | X | X | X |  |  |  | X | X | X | X | X |  | X |  |  | X |  |  |  | X |
| 49 | X |  | X |  | X | X |  |  | X | X |  | X |  |  | X |  | X |  | X |  | X | X |
| 50 |  | X | X | X |  |  | X |  | X | X | X |  |  | X | X |  |  |  | X |  | X |  |
| 51 | X |  |  |  | X | X | X | X | X |  |  | X |  |  | X |  |  | X |  |  | X | X |
| 52 | X | X | X |  |  | X |  |  | X | X | X | X | X | X | X |  |  |  |  |  |  |  |
| 53 |  | X | X | X | X | X |  |  | X | X |  |  |  | X | X |  | X |  | X |  | X |  |
| 54 | X | X |  | X |  | X |  |  | X | X |  |  |  |  | X | X |  |  | X | X | X | X |
| 55 | X | X | X |  |  | X |  | X |  | X |  |  |  | X | X |  | X |  |  |  | X |  |
| 56 | X |  |  |  | X | X |  | X | X | X |  |  |  | X | X |  | X |  | X |  | X |  |
| 57 |  |  | X | X | X | X |  | X |  | X |  |  |  |  | X | X |  |  |  |  | X | X |
| 58 |  | X | X | X |  | X | X |  | X |  |  | X |  |  | X |  |  |  | X | X | X | X |
| 59 | X | X | X |  | X |  |  |  | X | X | X |  |  | X | X |  |  |  | X | X | X |  |
| 60 | X | X |  |  | X | X |  | X | X | X |  |  | X | X | X |  | X |  |  |  |  |  |
| 61 | X | X | X | X |  | X |  |  | X | X | X |  | X |  | X |  |  |  |  |  |  | X |
| 62 | X | X | X |  |  |  | X |  | X | X |  |  |  | X | X |  | X |  | X |  | X |  |
| 63 |  | X |  | X | X | X |  | X | X | X |  | X | X | X | X |  |  |  | X |  |  |  |
| 64 | X | X |  | X |  | X | X | X |  |  |  | X |  |  | X |  |  |  | X |  | X | X |
| 65 |  |  |  | X | X | X |  | X | X | X | X | X |  | X | X |  |  |  |  |  | X |  |
| 66 | X |  | X |  | X |  | X |  | X | X |  |  | X |  | X |  | X |  | X | X |  | X |
| 67 |  | X | X | X |  |  | X | X |  | X |  |  |  | X | X |  | X |  | X |  | X |  |
| 68 | X | X | X |  |  | X |  | X |  | X |  |  | X | X | X |  | X |  | X |  |  |  |
| 69 |  |  | X | X | X | X |  |  | X | X |  |  | X |  | X |  | X | X |  |  |  | X |
| 70 | X |  |  |  | X |  | X | X | X | X | X |  | X |  | X |  |  |  |  |  |  | X |

Примечание: X означает наличие двухточечного линка или броадкастовой подсети

Таблица . Пулы выделенных адресов

| № вар. | IP адреса приватной сети | | IP адреса публичной сети | |
| --- | --- | --- | --- | --- |
| min | max | min | max |
| 1 | 172.20.0.0 | 172.24.0.0 | 173.16.168.0 | 173.16.172.0 |
| 2 | 10.18.0.0 | 10.22.0.0 | 220.208.29.0 | 220.208.33.0 |
| 3 | 10.221.0.0 | 10.225.0.0 | 178.144.179.0 | 178.144.183.0 |
| 4 | 192.168.241.0 | 192.168.245.0 | 172.49.59.0 | 172.49.63.0 |
| 5 | 10.232.0.0 | 10.236.0.0 | 172.5.196.0 | 172.5.200.0 |
| 6 | 192.168.10.0 | 192.168.14.0 | 215.179.63.0 | 215.179.67.0 |
| 7 | 172.19.0.0 | 172.23.0.0 | 172.54.105.0 | 172.54.109.0 |
| 8 | 10.126.0.0 | 10.130.0.0 | 153.234.180.0 | 153.234.184.0 |
| 9 | 10.149.0.0 | 10.153.0.0 | 142.250.193.0 | 142.250.197.0 |
| 10 | 192.168.199.0 | 192.168.203.0 | 211.17.194.0 | 211.17.198.0 |
| 11 | 10.43.0.0 | 10.47.0.0 | 206.197.91.0 | 206.197.95.0 |
| 12 | 192.168.127.0 | 192.168.131.0 | 160.210.92.0 | 160.210.96.0 |
| 13 | 172.19.0.0 | 172.23.0.0 | 207.200.203.0 | 207.200.207.0 |
| 14 | 192.168.100.0 | 192.168.104.0 | 196.171.168.0 | 196.171.172.0 |
| 15 | 10.195.0.0 | 10.199.0.0 | 120.195.0.0 | 120.199.0.0 |
| 16 | 172.26.0.0 | 172.30.0.0 | 195.17.71.0 | 195.17.75.0 |
| 17 | 10.199.0.0 | 10.203.0.0 | 130.228.47.0 | 130.228.51.0 |
| 18 | 172.19.0.0 | 172.23.0.0 | 221.168.102.0 | 221.168.106.0 |
| 19 | 172.23.0.0 | 172.27.0.0 | 221.239.235.0 | 221.239.239.0 |
| 20 | 192.168.220.0 | 192.168.224.0 | 172.210.64.0 | 172.210.68.0 |
| 21 | 192.168.180.0 | 192.168.184.0 | 223.189.136.0 | 223.189.140.0 |
| 22 | 10.172.0.0 | 10.176.0.0 | 98.44.0.0 | 98.48.0.0 |
| 23 | 10.228.0.0 | 10.232.0.0 | 178.97.52.0 | 178.97.56.0 |
| 24 | 172.18.0.0 | 172.22.0.0 | 102.125.0.0 | 102.129.0.0 |
| 25 | 10.171.0.0 | 10.175.0.0 | 58.125.0.0 | 58.129.0.0 |
| 26 | 172.20.0.0 | 172.24.0.0 | 73.118.0.0 | 73.122.0.0 |
| 27 | 172.23.0.0 | 172.27.0.0 | 67.51.0.0 | 67.55.0.0 |
| 28 | 172.19.0.0 | 172.23.0.0 | 172.152.137.0 | 172.152.141.0 |
| 29 | 192.168.233.0 | 192.168.237.0 | 11.121.0.0 | 11.125.0.0 |
| 30 | 172.17.0.0 | 172.21.0.0 | 172.1.96.0 | 172.1.100.0 |
| 31 | 192.168.111.0 | 192.168.115.0 | 209.1.11.0 | 209.1.15.0 |
| 32 | 10.72.0.0 | 10.76.0.0 | 202.105.52.0 | 202.105.56.0 |
| 33 | 192.168.36.0 | 192.168.40.0 | 118.59.0.0 | 118.63.0.0 |
| 34 | 172.27.0.0 | 172.31.0.0 | 189.107.244.0 | 189.107.248.0 |
| 35 | 192.168.137.0 | 192.168.141.0 | 217.188.245.0 | 217.188.249.0 |
| 36 | 10.237.0.0 | 10.241.0.0 | 51.49.0.0 | 51.53.0.0 |
| 37 | 10.57.0.0 | 10.61.0.0 | 93.201.0.0 | 93.205.0.0 |
| 38 | 10.114.0.0 | 10.118.0.0 | 204.240.251.0 | 204.240.255.0 |
| 39 | 192.168.132.0 | 192.168.136.0 | 198.38.82.0 | 198.38.86.0 |
| 40 | 10.132.0.0 | 10.136.0.0 | 172.235.231.0 | 172.235.235.0 |
| 41 | 172.23.0.0 | 172.27.0.0 | 60.216.0.0 | 60.220.0.0 |
| 42 | 172.16.0.0 | 172.20.0.0 | 200.28.29.0 | 200.28.33.0 |
| 43 | 10.161.0.0 | 10.165.0.0 | 203.159.74.0 | 203.159.78.0 |
| 44 | 172.27.0.0 | 172.31.0.0 | 145.250.243.0 | 145.250.247.0 |
| 45 | 192.168.170.0 | 192.168.174.0 | 172.121.114.0 | 172.121.118.0 |
| 46 | 192.168.111.0 | 192.168.115.0 | 129.254.68.0 | 129.254.72.0 |
| 47 | 10.216.0.0 | 10.220.0.0 | 187.216.161.0 | 187.216.165.0 |
| 48 | 172.23.0.0 | 172.27.0.0 | 206.218.180.0 | 206.218.184.0 |
| 49 | 10.177.0.0 | 10.181.0.0 | 130.62.165.0 | 130.62.169.0 |
| 50 | 10.75.0.0 | 10.79.0.0 | 197.197.55.0 | 197.197.59.0 |
| 51 | 172.24.0.0 | 172.28.0.0 | 60.93.0.0 | 60.97.0.0 |
| 52 | 10.0.0.0 | 10.4.0.0 | 173.236.37.0 | 173.236.41.0 |
| 53 | 10.236.0.0 | 10.240.0.0 | 180.119.18.0 | 180.119.22.0 |
| 54 | 172.23.0.0 | 172.27.0.0 | 76.136.0.0 | 76.140.0.0 |
| 55 | 192.168.5.0 | 192.168.9.0 | 199.68.34.0 | 199.68.38.0 |
| 56 | 10.161.0.0 | 10.165.0.0 | 69.53.0.0 | 69.57.0.0 |
| 57 | 192.168.52.0 | 192.168.56.0 | 192.64.243.0 | 192.64.247.0 |
| 58 | 10.218.0.0 | 10.222.0.0 | 82.196.0.0 | 82.200.0.0 |
| 59 | 192.168.178.0 | 192.168.182.0 | 219.232.65.0 | 219.232.69.0 |
| 60 | 10.55.0.0 | 10.59.0.0 | 88.187.0.0 | 88.191.0.0 |
| 61 | 192.168.100.0 | 192.168.104.0 | 187.64.238.0 | 187.64.242.0 |
| 62 | 10.159.0.0 | 10.163.0.0 | 142.168.232.0 | 142.168.236.0 |
| 63 | 172.24.0.0 | 172.28.0.0 | 204.212.140.0 | 204.212.144.0 |
| 64 | 172.21.0.0 | 172.25.0.0 | 172.8.112.0 | 172.8.116.0 |
| 65 | 172.20.0.0 | 172.24.0.0 | 91.142.0.0 | 91.146.0.0 |
| 66 | 172.23.0.0 | 172.27.0.0 | 120.77.0.0 | 120.81.0.0 |
| 67 | 172.17.0.0 | 172.21.0.0 | 217.22.34.0 | 217.22.38.0 |
| 68 | 192.168.67.0 | 192.168.71.0 | 44.132.0.0 | 44.136.0.0 |
| 69 | 192.168.29.0 | 192.168.33.0 | 136.102.217.0 | 136.102.221.0 |
| 70 | 10.242.0.0 | 10.246.0.0 | 4.97.0.0 | 4.101.0.0 |

# Пример выполнения задания

## **Построение модели сети**

В качестве примера выполнения курсовой работы выберем вариант 53 (рис. 50):

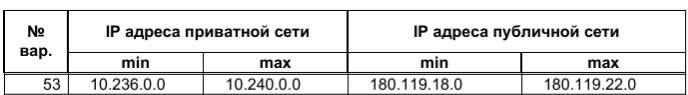
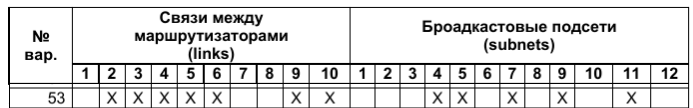


Рис. . Вариант 53.

Скачаем заготовку blank.pka, в User Profile … укажем свою фамилию и номер группы. На поле модели введем также фамилию и номер группы, пулы приватных и публичных адресов. Добавим физические сервера (DNS, web-сервера в приватной и публичной сети), коммутаторы S4, S5, S7, S9 и S11, перекинем между устройствами кабели. Кроссоверные кабели и кабели прямого обжима в CPT обозначаются разным типом линии.

Поясняющее видео по первоначальной сборке сети см. в таблице 18 (первая ссылка). В тупиковых (широковещательных) сетях (subnet) к коммутаторам добавляем по одному или два персональных компьютера (PC) или ноутбука (laptop).

## **Планирование IP адресации сети**

### Приватная сеть

Пул приватных адресов 10.236.0.0 - 10.240.0.0. Всего насчитывается пять подсетей с маской /16.

|  |  |
| --- | --- |
| 10.236.0.0 | 236 → 11101100 |
| 10.237.0.0 | 237 → 11101101 |
| 10.238.0.0 | 238 → 11101110 |
| 10.239.0.0 | 239 → 11101111 |
| 10.240.0.0 | 240 → 11110000 |

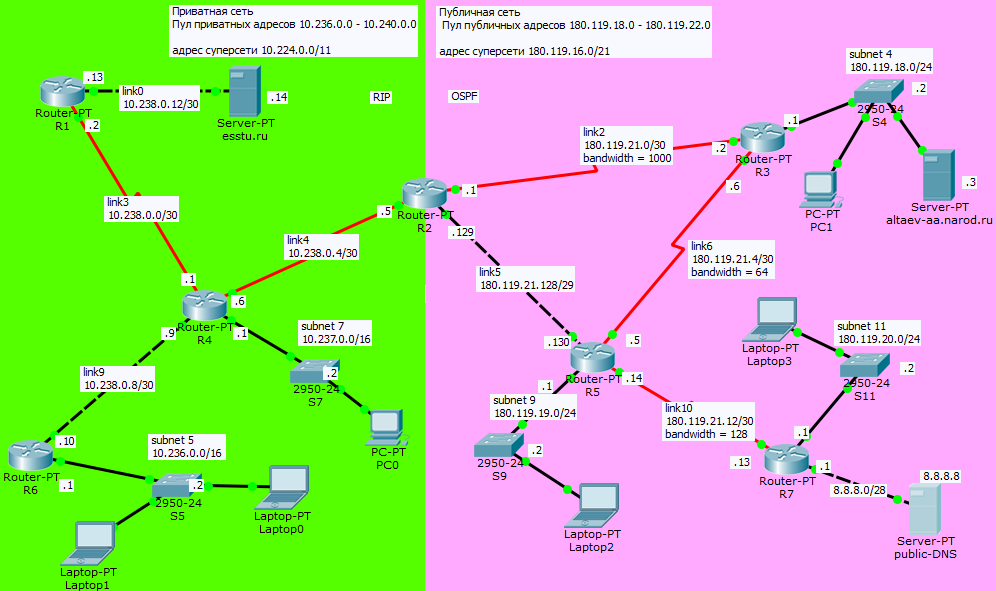


Рис. 51. Поле модели сети 53-го варианта

Сравнивая старшие биты, выделяем совпадающие биты. Таких битов в нашем случае 3 и их оставляем в качестве сетевых, остальные пять младших битов превращаем в хостовые и обнуляем: 11100000 → 224.

Следовательно, адрес суперсети 10.224.0.0/11[[26]](#footnote-27).

Адрес 10.236.0.0/16 назначаем тупиковой[[27]](#footnote-28) сети на коммутаторе S5.

Адрес 10.237.0.0/16 назначаем тупиковой сети на коммутаторе S7.

Сеть 10.238.0.0 разбиваем на подсети с маской /30. Это позволит назначить адреса двухточечным линкам. Работаем в четвертом октете:

|  |  |  |
| --- | --- | --- |
| 00000000 | → | 10.238.0.0/30 для линка между R1 и R4. |
| 00000100 | → | 10.238.0.4/30 для линка между R2 и R4. |
| 00001000 | → | 10.238.0.8/30 для линка между R4 и R6. |
| 00001100 | → | 10.238.0.12/30 для линка между R1 и сервером esstu.ru. |

Подсети 10.239.0.0 и 10.240.0.0 оставляем в резерве.

### Публичная сеть

Пул публичных адресов 180.119.18.0 - 180.119.22.0. Находим битовое представление третьего октета для всех пяти подсетей.

18 → 00010010

19 → 00010011

20 → 00010100

21 → 00010101

22 → 00010110

Сравнивая старшие биты, выделяем совпадающие биты. Таких битов в нашем случае 5, их оставляем в качестве сетевых, остальные три младших бита превращаем в хостовые и обнуляем: 16 → 00010000.

Таким образом, адрес суперсети - 180.119.16.0/21[[28]](#footnote-29)

Назначаем тупиковым сетям их адреса:

S4 - 180.119.18.0/24

S9 - 180.119.19.0/24

S11 - 180.119.20.0/24

Сеть 180.119.21.0/24 предназначается для получения подсетей более мелкого масштаба, сеть 180.119.22.0/24 помещается в резерв.

Сеть 180.119.21.0/24 поделим на две подсети равного размера:

00000000 → 180.119.21.0/25

10000000 → 180.119.21.128/25

Из второй подсети выделим подсеть еще меньшего масштаба

10000000 → 180.119.21.128/29

Этот адрес используем для линка между R2 и R5. Такой увеличенный пул адресов необходим для функционирования NAT на маршрутизаторе R2.

Подсеть 180.119.21.0/25 разбиваем на более мелкие подсети с маской /30 и назначаем всем оставшимся двухточечным линкам публичной сети

00000000 → 180.119.21.0/30 для линка между R2 и R3

00000100 → 180.119.21.4/30 для линка между R3 и R5

00001000 → 180.119.21.8/30 оставили в резерве

00001100 → 180.119.21.12/30 для линка между R5 и R7

## **Базовая настройка маршрутизаторов и коммутаторов и реализация IP адресации**

На поле модели помещаем все поясняющие надписи с указанием адресов подсетей с их масками (рис. 51).

IP адресацию устройств проводим, исходя из следующих указаний: в широковещательной сети первый хостовый адрес должен получить маршрутизатор, второй -коммутатор[[29]](#footnote-30), серверу следует определить третий адрес. Персональным компьютерам PC и ноутбукам laptop IP адреса не назначаются, так как они их получат по протоколу DHCP.

Назначаем интерфейсам всех промежуточных и конечных устройств их IP адреса. На serial линках на их DCE интерфейсах указываем пропускную способность.

Пример настройки интерфейса **FastEthernet1/0** на R5

**R5(config)#interface FastEthernet1/0**

**R5(config-if)#ip address 180.119.19.1 255.255.255.0**

**R5(config-if)#no shutdown**

**R5(config-if)#exit**

Пример настройки интерфейса **Serial2/0** на R5

**R5(config)#interface Serial2/0**

**R5(config-if)#clock rate 64000**

**This command applies only to DCE interfaces**

**R5(config-if)#bandwidth 64**

**R5(config-if)#ip address 180.119.21.5 255.255.255.252**

**R5(config-if)#no shutdown**

**R5(config-if)#exit**

Команда **clock rate** не была выполнена, так как интерфейс относится к типу DTE. Следовательно, другой стороне serial линка (на маршрутизаторе R3) данную команду следует обязательно реализовать. На разных serial соединениях следует использовать разные значения **bandwidth.** Это позволит убедиться, что протокол маршрутизации OSPF для выбора оптимальных маршрутов учитывает значение пропускной способности линка.

Весь порядок действий по первоначальной настройке сети приведен на серии видео (табл. 18).

Таблица .

| Описание | Ссылка |
| --- | --- |
| Начало сборки сети | <https://yadi.sk/i/4K2SCUlp5SmGCg> |
| Кабельная инфраструктура сети | <https://yadi.sk/i/FPeq1xCbTXouGg> |
| Размещение коммутаторов | <https://yadi.sk/i/pHS35uq_QZfu3Q> |
| Размещение серверов | <https://yadi.sk/i/tk-iYKszAbt8Hg> |
| Планирование IP адресации для внутренней сети | <https://yadi.sk/i/PtDckcv6ck2zhg> |
| IP конфигурирование сетевых устройств в приватной сети | <https://yadi.sk/i/zY9uyOyesL9u5w> |
| Вычисление адресов суперсетей и разбиение на подсети | <https://yadi.sk/i/3_BYiHjjez9-BQ> |
| IP конфигурирование сетевых устройств в публичной сети | <https://yadi.sk/i/liMBbFR0GArWzw> |
| Настройка serial линков в публичной сети | <https://yadi.sk/i/APSfAuMAvYQqhA> |

Завершается данный этап проверкой связи между соседними устройствами. Пинги между всеми соседями должны быть успешными.

## **DHCP**

Механизм функционирования DHCP был рассмотрен в теме «Служебные адреса». DHCP сервера следует развернуть на тех маршрутизаторах, интерфейсы которых обращены в сторону тупиковых (широковещательных) сетей. Отметим, что при более сложной настройке DHCP сервера размещаются на одном маршрутизаторе, а на других маршрутизаторах используют ip helper address, позволяющий переадресовывать DHCP запросы на маршрутизатор с сервером.

Пример настройки DHCP сервера для тупиковой сети S5 на маршрутизаторе R6 (рис. 52). Корректность работы DHCP проверяется на двух ноутбуках.

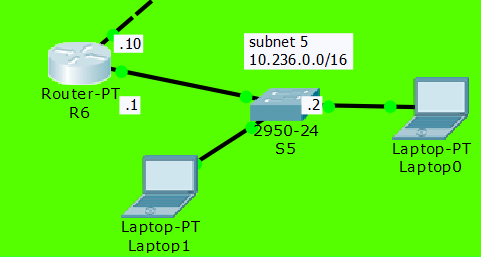


Рис. 52. Тупиковая сеть 10.236.0.0/16.

**R6(config)#ip dhcp excluded-address 10.236.0.1 10.236.0.10**

**R6(config)#ip dhcp pool S5**

**R6(dhcp-config)#network 10.236.0.0 255.255.0.0**

**R6(dhcp-config)#default-router 10.236.0.1**

**R6(dhcp-config)#dns-server 8.8.8.8**

**R6(dhcp-config)#exit**

Команда

**R6(config)#ip dhcp excluded-address 10.236.0.1 10.236.0.10**

служит для исключения из раздачи первых десяти адресов. В коммутируемой сети большого размера может быть несколько коммутаторов, которые должны иметь свои IP адреса. Также в сети могут быть и сервера, адреса которым не следует раздавать через DHCP протокол.

Пробное выполнение DHCP клиента на сетевой карте ноутбука Laptop1 показывает, что DHCP сервер настроен верно (рис.53).

DHCP сервера должны быть настроены для всех широковещательных сетей. Следует ко всем коммутаторам тупиковых сетей присоединить по одному персональному компьютеру (PC) или ноутбуку (laptop) и проверить получат ли они IP информацию от DHCP сервера своей сети.

На видео (https://yadi.sk/i/FBGF-7va1e8cEg) отображен порядок действий, выполняемых при настройке DHCP.

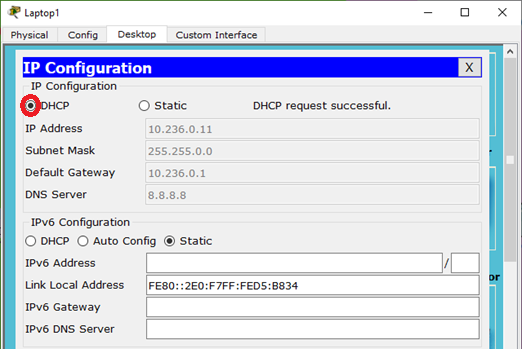


Рис. . Проверка DHCP на Laptop1.

## **Реализация динамической маршрутизации**

### RIP

Рассмотрим настройку протокола RIP на маршрутизаторе R4 (рис. 54).

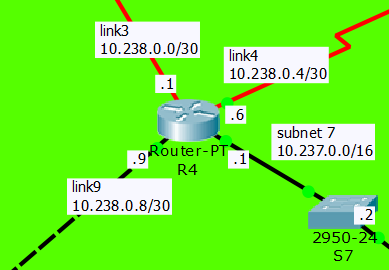


Рис. . Маршрутизатор R4.

Все присоединенные (directly connected) к маршрутизатору сети должны быть включены в RIP процесс с учетом, что подсети 10.238.0.0/30, 10.238.0.4/30, 10.238.0.8/30, 10.237.0.0/16 являются частью классовой сети 10.0.0.0/8.

**R4(config)#router rip**

**R4(config-router)#version 2**

**R4(config-router)#passive-interface FastEthernet1/0**

**R4(config-router)#network 10.0.0.0**

**R4(config-router)#exit**

2-я версия RIP позволяет включить в маршрутные информации помимо адреса сети, так ее маску. Через порт **FastEthernet1/0** запрещаем RIP трафик, так как данный интерфейс обращен в сторону тупиковой сети, где нет других RIP маршрутизаторов.

Рассмотрим настройку RIP на пограничном маршрутизаторе R2 (рис. 55), на котором два интерфейса не должны быть включены в RIP процесс, ибо они находятся в зоне OSPF сети.

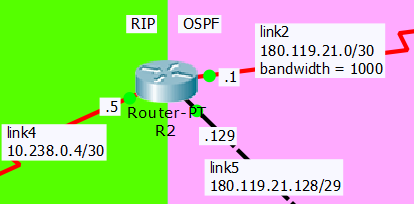


Рис. . Маршрутизатор R2.

Команды настройки RIP:

**R2(config)#router rip**

**R2(config-router)#version 2**

**R2(config-router)#redistribute static**

**R2(config-router)#network 10.0.0.0**

**R2(config-router)#default-information originate**

**R2(config-router)#exit**

На пограничном маршрутизаторе присутствует маршрут по умолчанию, направленный в OSPF сеть через **FastEthernet0/0** интерфейс

**R2(config)#ip route 0.0.0.0 0.0.0.0 FastEthernet0/0**

Данный маршрут следует распространить по RIP сети, чтобы остальные маршрутизаторы узнали свой дефолтный маршрут. За это отвечают две команды:

**default-information originate**

**redistribute static**

Первая команда сообщает, что данный маршрутизатор является источником дефолтного маршрута, вторая предписывает протоколу распространить данный маршрут на соседние RIP маршрутизаторы.

RIP также запускается на маршрутизаторах R1 и R6. По завершении настройки протокола на всех маршрутизаторах следует проверить таблицы маршрутизации. Маршрутизаторы должны иметь маршруты ко всем сетям приватной зоны и иметь маршрут по умолчанию.

Просмотр таблицы маршрутизации на R4:

**R4# sh ip route**

**<часть текста вывода пропущена>**

**Gateway of last resort is 10.238.0.5 to network 0.0.0.0**

**10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks**

**R 10.236.0.0/16 [120/1] via 10.238.0.10, 00:00:02, FastEthernet0/0**

**C 10.237.0.0/16 is directly connected, FastEthernet1/0**

**C 10.238.0.0/30 is directly connected, Serial2/0**

**C 10.238.0.4/30 is directly connected, Serial3/0**

**C 10.238.0.8/30 is directly connected, FastEthernet0/0**

**R 10.238.0.12/30 [120/1] via 10.238.0.2, 00:00:00, Serial2/0**

**R\* 0.0.0.0/0 [120/1] via 10.238.0.5, 00:00:03, Serial3/0**

Адреса двух удаленных сетей приватной зоны получены через RIP. Протокол в таблицу также добавил маршрут по умолчанию. Он направлен через маршрутизатор R2 (адрес 10.238.0.5).

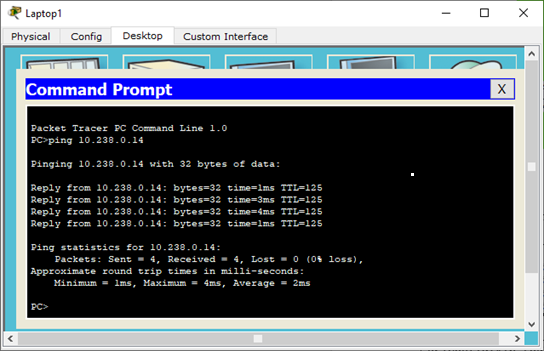


Рис. . Проверка доступности сервера.

Если RIP был настроен без ошибок, то все узлы приватной области должны иметь доступ к друг другу. Например, с Laptop1 должен быть доступен сервер esstu.ru (рис. 56).

На видео (https://yadi.sk/i/8gz8JXEjKIwQHQ) отображен порядок действий, выполняемых при настройке RIP.

### OSPF

В public области сети настраивается OSPF, являющийся на данный момент самым распространённым протоколом для LAN и MAN сетей. OSPF позволяет разбить сеть на отдельные области (area), что позволяет добиться агрегирования маршрутов. Область с идентификатором 0 является магистральной, и она может быть единственной если OSPF настраивается в сети небольшого размера. В курсовой работе используется лишь область 0 (area 0). Настройки OSPF на маршрутизаторе R5 (рис. 57) выглядят следующим образом:

**R5(config)#router ospf 1**

**R5(config-router)#router-id 5.5.5.5**

**R5(config-router)#passive-interface FastEthernet1/0**

**R5(config-router)#network 180.119.19.0 0.0.0.255 area 0**

**R5(config-router)#network 180.119.21.12 0.0.0.3 area 0**

**R5(config-router)#network 180.119.21.128 0.0.0.7 area 0**

**R5(config-router)#network 180.119.21.4 0.0.0.3 area 0**

Первая команда запускает OSPFпроцесс с идентификатором 1. Идентификатор процесса имеет локальное значение (другим OSPF маршрутизаторам не передается) и может быть произвольным целым числом.

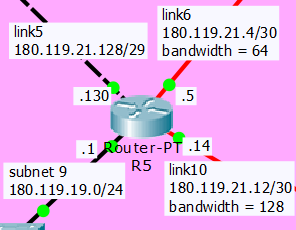


Рис. 57. Маршрутизатор R5.

OSPF маршрутизаторы должны различаться межу собой идентификаторами. В данном варианте настройки использована команда **router-id**. Идентификатор похож на IP адрес, но им фактически не является. Отметим, что помимо команды **router-id** предусмотрены и другие способы назначения идентификатора (в данной работе они не рассматриваются).

В команде network помимо адреса сети указывается шаблонная маска, являющаяся инверсией основной маски подсети.

OSPF должен знать пропускную способность (bandwidth) serial интерфейсов. Например, для интерфейса **Serial2/0** должна быть выполнена команда **bandwidth**:

**R5(config)#interface Serial2/0**

**R5(config-if)#bandwidth 64**

По завершении настроек OSPF на всех маршрутизаторах public сети, следует проверить состояние таблиц маршрутизации. На маршрутизаторе на маршрутизаторе R5 (рис. 57) таблица имеет вид:

**R5#sh ip route**

**<часть текста вывода пропущена>**

**Gateway of last resort is not set**

**8.0.0.0/28 is subnetted, 1 subnets**

**O 8.8.8.0 [110/65] via 180.119.21.13, 00:35:54, Serial3/0**

**180.119.0.0/16 is variably subnetted, 7 subnets, 3 masks**

**O 180.119.18.0/24 [110/102] via 180.119.21.129, 00:35:24, FastEthernet0/0**

**C 180.119.19.0/24 is directly connected, FastEthernet1/0**

**O 180.119.20.0/24 [110/65] via 180.119.21.13, 00:35:54, Serial3/0**

**O 180.119.21.0/30 [110/101] via 180.119.21.129, 00:35:24, FastEthernet0/0**

**C 180.119.21.4/30 is directly connected, Serial2/0**

**C 180.119.21.12/30 is directly connected, Serial3/0**

**C 180.119.21.128/29 is directly connected, FastEthernet0/0**

Четыре маршрута получены через протокол OSPF и четыре сети напрямую присоединены к маршрутизатору.

Проверка таблиц маршрутизации на пограничном маршрутизаторе:

**R2#sh 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 0.0.0.0 to network 0.0.0.0**

**8.0.0.0/28 is subnetted, 1 subnets**

**O 8.8.8.0 [110/66] via 180.119.21.130, 00:57:50, FastEthernet0/0**

**10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks**

**R 10.236.0.0/16 [120/2] via 10.238.0.6, 00:00:07, Serial3/0**

**R 10.237.0.0/16 [120/1] via 10.238.0.6, 00:00:07, Serial3/0**

**R 10.238.0.0/30 [120/1] via 10.238.0.6, 00:00:07, Serial3/0**

**C 10.238.0.4/30 is directly connected, Serial3/0**

**R 10.238.0.8/30 [120/1] via 10.238.0.6, 00:00:07, Serial3/0**

**R 10.238.0.12/30 [120/2] via 10.238.0.6, 00:00:07, Serial3/0**

**180.119.0.0/16 is variably subnetted, 7 subnets, 3 masks**

**O 180.119.18.0/24 [110/101] via 180.119.21.2, 00:58:25, Serial2/0**

**O 180.119.19.0/24 [110/2] via 180.119.21.130, 00:57:50, FastEthernet0/0**

**O 180.119.20.0/24 [110/66] via 180.119.21.130, 00:57:50, FastEthernet0/0**

**C 180.119.21.0/30 is directly connected, Serial2/0**

**O 180.119.21.4/30 [110/1563] via 180.119.21.130, 00:57:50, FastEthernet0/0**

**O 180.119.21.12/30 [110/65] via 180.119.21.130, 00:57:50, FastEthernet0/0**

**C 180.119.21.128/29 is directly connected, FastEthernet0/0**

**S\* 0.0.0.0/0 is directly connected, FastEthernet0/0**

Маршрутизатор «видит» все сети приватной и публичной областей. Маршрут по умолчанию направлен через интерфейс **FastEthernet0/0**. Выбор данного интерфейса обусловлен его более высокой пропускной способностью (100 Mbps) по сравнению с serial интерфейсом с принятым на нем bandwidth = 1000 kbps.

Следует проверить наличие коннективности между всеми хостами публичной области, используя команду **ping**. На рисунке 58 приведен пример одной из таких проверок. Все хосты должны быть доступны друг другу.

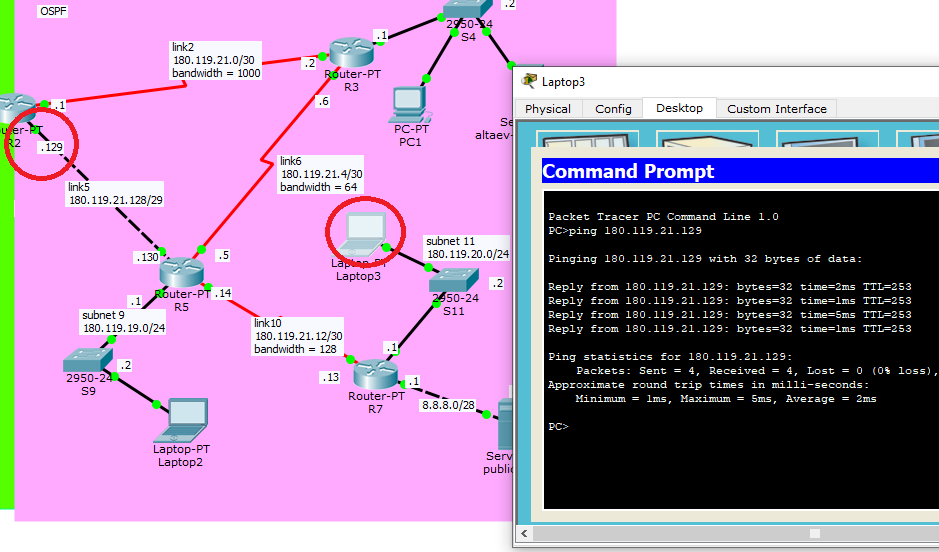


Рис. 58. Проверка доступности маршрутизатора R2 со стороны ноутбука laptop3.

На видео (https://yadi.sk/i/QFazCU6\_okpfWw) отображен порядок действий, выполняемых при настройке OSPF.

## **Настройка NAT на пограничном маршрутизаторе**

Хотя маршрутизатор R2 знает обо всех сетях данной топологии и маршрутизаторы внутренней сети имеют маршрут по умолчанию, хосты приватной и публичных областей не могут отправлять друг другу сообщения, так как внешние хосты не «видят» приватные адреса внутренней сети. Исправляется такая ситуация использованием NAT (Network Address Translation - Трансляция сетевых адресов) на пограничном маршрутизаторе. NAT позволяет хостам внутренней сети получить доступ к ресурсам Всемирной сети, в основном, к web серверам. Общий принцип функционирования NAT рассматривался в разделе «Публичные и приватные адреса, технология NAT».

Маршрутизатору следует указать какие порты него находятся в приватной, а какие в публичной зонах (рис.15 и 59):

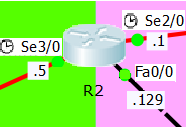


Рис. . Интерфейсы маршрутизатора R2.

**R2(config)#interface Serial3/0**

**R2(config-if)#ip nat inside**

**R2(config-if)#interface Serial2/0**

**R2(config-if)#ip nat outside**

**R2(config-if)#interface FastEthernet0/0**

**R2(config-if)#ip nat outside**

**R2(config-if)#exit**

С помощью списка контроля доступа ACL (Access List Control, ACL) указываем из какого пула приватных адресов будет разрешена трансляция. Списки ACL позволяют эффективно контролировать сеть, отфильтровывая нежелательный трафик. В данной работе не обсуждаются принципы функционирования списков, подробные сведения об ACL с примерами их применения можно почерпнуть из web источников [22, 23].

ACL, примененный на R2, имеет вид:

**R2(config)#access-list 1 permit 10.224.0.0 0.31.255.255**

Список относится к стандартному ACL, его номер равен 1, в качестве пула был указан адрес суперсети[[30]](#footnote-31) и его шаблонная маска.

Создается пул публичных адресов

**R2(config)#ip nat pool public\_pool 180.119.21.131 180.119.21.134 netmask 255.255.255.248**

Диапазон адресов **180.119.21.131 180.119.21.134** относится к интерфейсу **FastEthernet0/0**. Именно через него проложен маршрут по умолчанию.

Оба пула (приватный и публичный) связываем между собой командой

**R2(config)#ip nat inside source list 1 pool public\_pool overload**

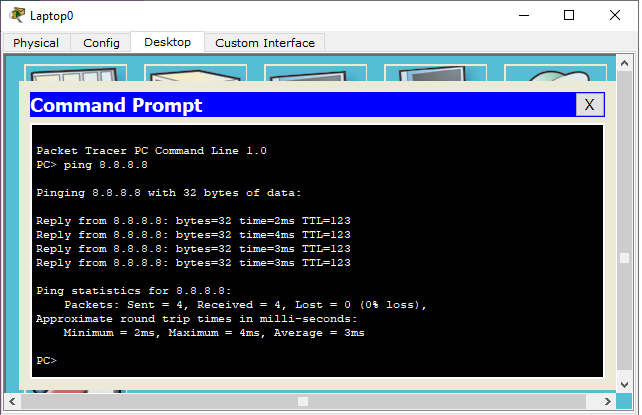


Рис. . Проверка доступности внешнего хоста.

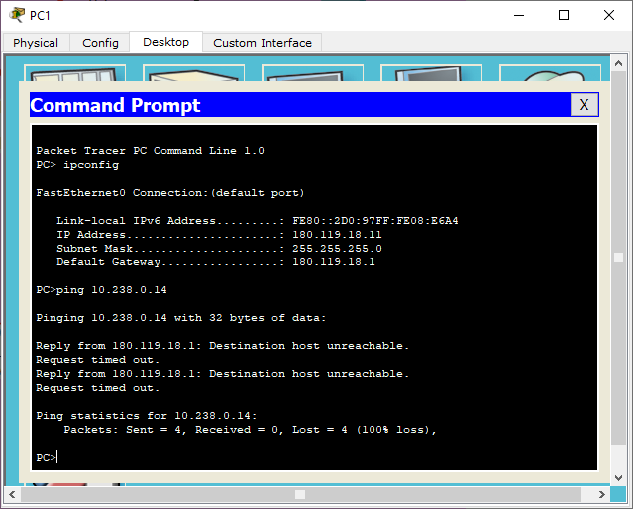


Рис. . Хост из сети 180.119.18.0/24 не имеет доступ серверу корпоративной сети.

Проверяем работу NAT, отправив пинг с хоста внутренней сети на любой внешний хост. Пример успешного пинга приведен на рисунке 60.

При этом внешние хосты не имеют доступ ко внутренним (рис. 61).

## **Настройка DNS серверов**

### Обеспечение доступности публичного сайта

Общие принципы работы DNS были рассмотрены в разделе «DNS». При настройке DHCP серверов адрес DNS сервера был указан как 8.8.8.8. Хосты, соответственно, получают его адрес (рис.53). В топологии сети такой сервер имеется. Перейдя на его вкладку Services активируем DNS службу и внесем запись о web сайте altaev-aa.narod.ru (рис. 62).

На сервере altaev-aa.narod.ru активируем HTTP службу и вносим изменения в файл index.html (рис. 63 и рис. 64). Перед проверкой доступности сайтов предварительно следует обеспечить хосты информацией о MAC адресах соседей, используя кнопку Add Simple PDU (P) (рис. 65). Результаты пингов удаляются кнопкой Delete. Проверяем доступность сайта с внешнего (рис. 66) и внутреннего (рис. 67) хостов.

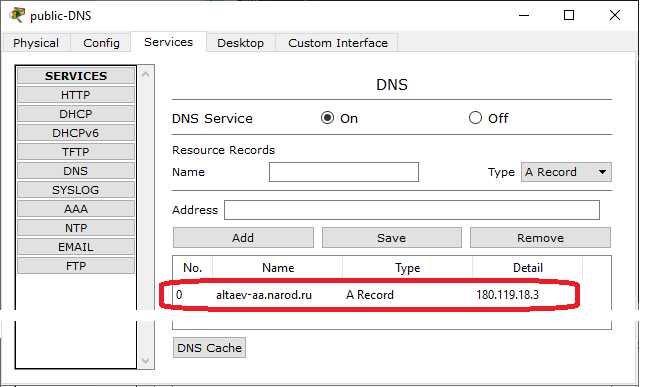


Рис. 62. Заполнение базы данных в DNS.

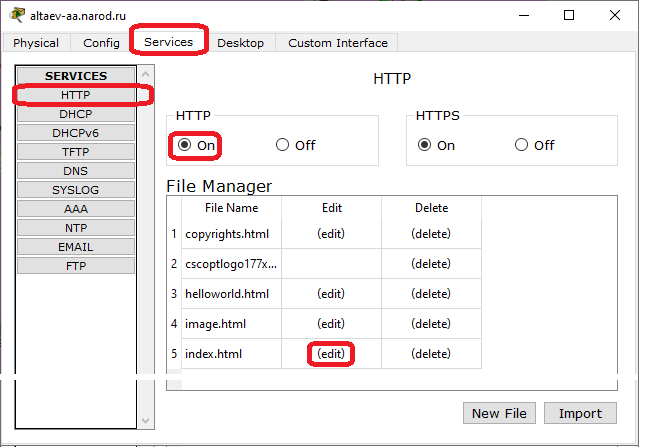


Рис. 63. Активация HTTP службы и редактирование сайта.

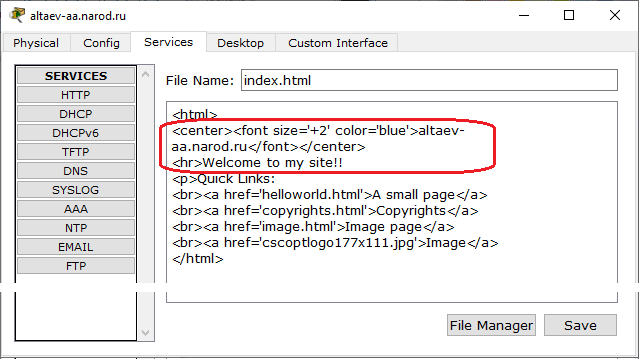


Рис. 64. Редактирование сайта.

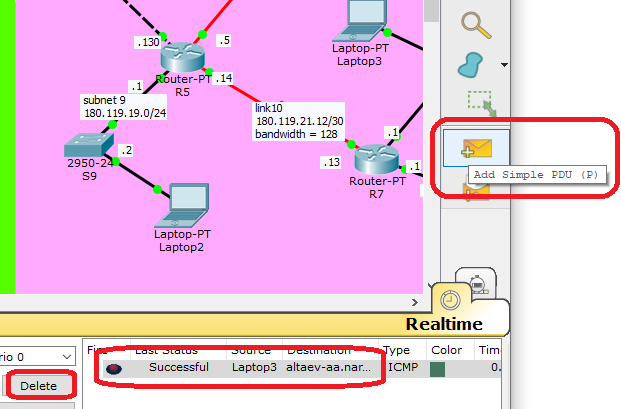


Рис. . Кнопка генерации пингов.

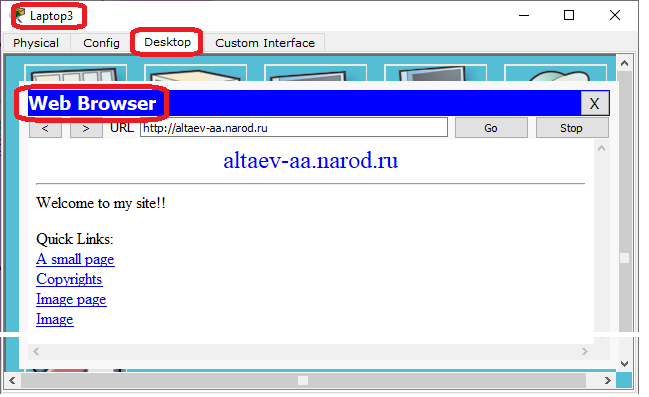


Рис. 66. Проверка доступности сайта с внешнего хоста.

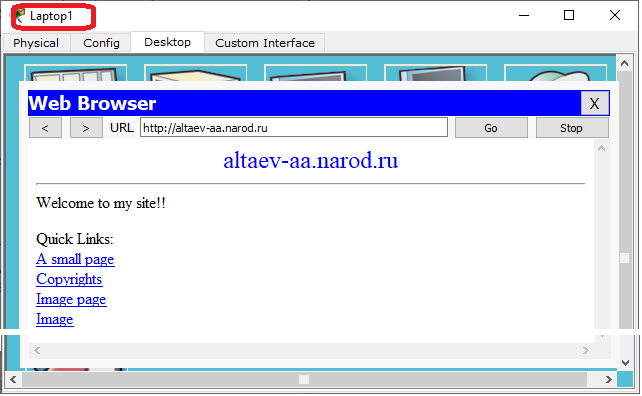


Рис. 67. Внешний сайт доступен с внутреннего хоста.

### Обеспечение доступности корпоративного сайта извне

Корпоративный сайт esstu.ru может быть доступен лишь через публичный адрес пограничного маршрутизатора. Вносим соответствующую DNS запись в БД сервера (рис. 68).

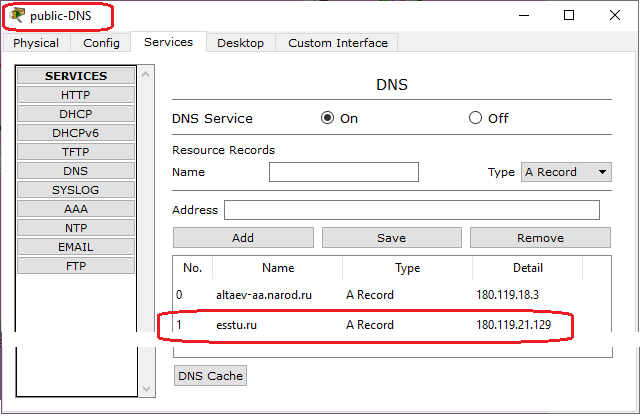


Рис. . Обеспечение доступа к корпоративному сайту с public сети.

Вносим на пограничном маршрутизаторе изменения в настройки NAT, позволяющие ему распознавать TCP запросы с public сети к внутреннему сайту. Распознавание должно выполняться по порту назначения, который равен 80 (данный порт закреплен за протоколом HTTP). Команда, разрешающая прохождение через R2 HTTP трафика, имеет вид

**R2(config)#ip nat inside source static tcp 10.238.0.14 80 180.119.21.129 80**

Данная команда связывает приватный адрес (**10.238.0.14**) и публичный адрес маршрутизатора (**180.119.21.129**) между собой. Связка из двух адресов (IP и порт) образует сокет (**180.119.21.129:80**), позволяющий распознавать HTTP трафик и, следовательно, пропускать через маршрутизатор с заменой публичного адреса на приватный.

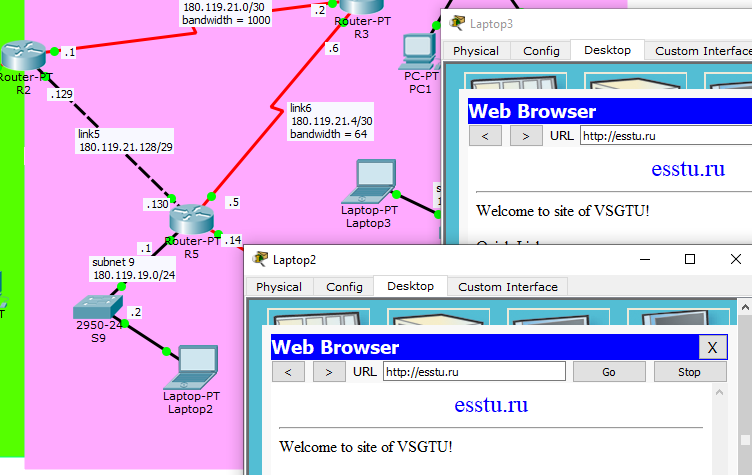


Рис. . Корпоративный сайт открыт для просмотра из сети Интернет.

Проверка на public хостах показывает, что сайт esstu.ru открыт для просмотра со стороны внешних хостов (рис. 69).

### Обеспечение доступности корпоративного сайта внутренним хостам по URL адресу

Хосты корпоративной сети имеют доступ к сайту организации по его IP адресу (рис. 70). Однако попытка обращения к сайту по его URL адресу приводит к отказу (рис. 71).

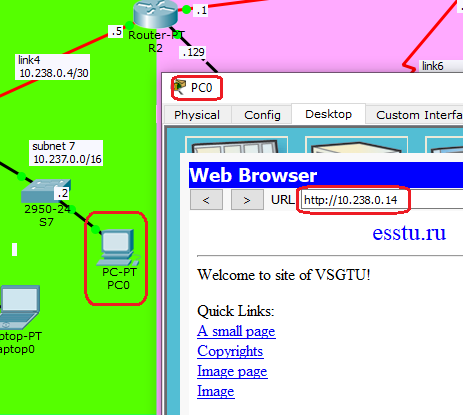


Рис. 70. Проверка доступа к сайту организации по его IP адресу.

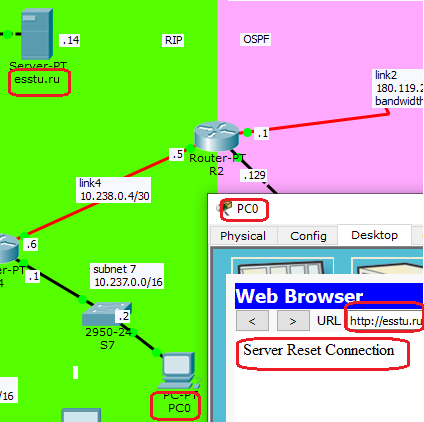


Рис. 71. Отсутствие доступа к сайту по его URL адресу.

Связано данное обстоятельство с тем, что у хостов приватной области в качестве адреса DNS сервера указан 8.8.8.8 (рис.53). Но на самом DNS сервере запись на ресурс esstu.ru указывает на адрес 180.119.21.129 (рис. 68). Это адрес внешнего интерфейса пограничного маршрутизатора. Хостам приватной области следует получать адрес самого сервера esstu.ru. Решается данный вопрос применением локального DNS сервера. Развернуть его можно на том же сервере, на котором работает web сервер (10.238.0.14).

Настройки DNS сервера приведены на рисунке 72.

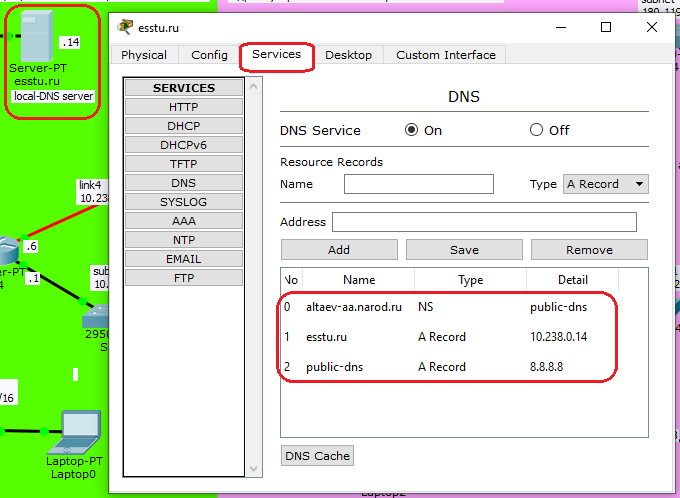


Рис. 72. Записи локального DNS сервера.

Если внутренний хост присылает DNS запрос на сайт altaev-aa.narod.ru, то он пересылается на публичный DNS сервер (8.8.8.8).

Внутренние хосты должны иметь своих настройках адрес DNS сервера в виде 10.238.0.14. Следует изменить записи DHCP серверов всей приватной области (маршрутизаторы R1, R4 и R6). Например, на R6 вводим команды:

**R6(config)#ip dhcp pool S5**

**R6(dhcp-config)#dns-server 10.238.0.14**

Пример проверки доступности корпоративного сайта с внутреннего хоста приведен на рисунке 73.

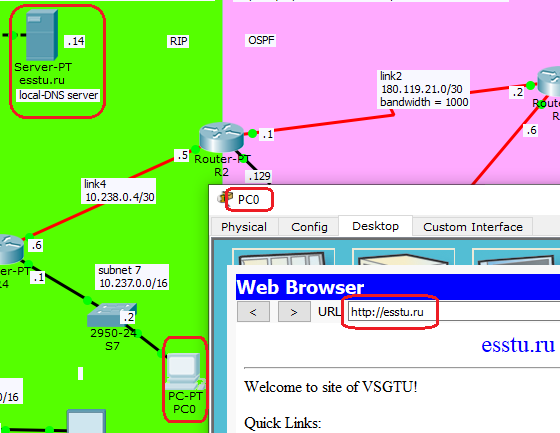


Рис. 73. Приватный сайт доступен с приватных хостов.

# Оформление отчета

Отчет по курсовой работе оформляется 12–м шрифтом, через 1,5 интервала, на одной стороне листа стандартного формата А4 с полями: левое - 3,0 см, правое - 1,0 см, верхнее и нижнее – по 2 см. Пример титульного лист приведен в Приложении 1.

# Литература

1. Компьютерные сети. Принципы, технологии, протоколы : учебное пособие для студентов высших учебных заведений, обучающихся по направлению "Информатика и вычислительная техника" и по специальностям "Вычислительные машины, комплексы, системы и сети", "Автоматизированные машины, комплексы, системы и сети", "Программное обеспечение вычислительной техники и автоматизированных систем" / В. Г. Олифер, Н. А. Олифер. - 4-е изд. - Санкт-Петербург ; Москва ; Нижний Новгород : ПИТЕР, 2015. - 943 с. : ил. ; 24 см.
2. Компьютерные сети. Ч.2. Технологии локальных и глобальных сетей / Е. В. Нужнов. - Таганрог : Изд-во ЮФУ, 2015. - 176 с.
3. Компьютерные сети: учебное пособие для образовательных учреждений, реализующих программы среднего профессионального образования по специальности "Информационные системы (по отраслям)" / Е. О. Новожилов, О. П. Новожилов. - 5-е изд., стер. - Москва : Издательский центр "Академия", 2017. - 223, [1] с. : рис., табл. ; 21 см.
4. Организация сетевого администрирования: учебник по специальности 09.02.02 "Компьютерные сети" / А. И. Баранчиков [и др.]. - Москва : КУРС : ИНФРА-М, 2019. - 383 с. : рис., табл. ; 21 см.

# Web ресурсы

1. Авторский сайт Алтаева А.А. URL: http://www.altaev-aa.narod.ru/net/ (дата обращения: 07.12.21).
2. Сети и сетевые технологии. URL: http://datanets.ru/ (дата обращения: 07.12.21).
3. Справочник портов TCP и UDP. URL: https://pc.ru/docs/network/port-numbers#vstuplenie (дата обращения: 07.12.21).
4. Основы протокола TCP для чайников простыми словами. URL: https://zvondozvon.ru/tehnologii/protokoli/tcp (дата обращения: 11.12.21).
5. Классовая адресация – Википедия. URL: https://ru.wikipedia.org/wiki/Классовая\_адресация (дата обращения: 23.12.21).
6. List of assigned /8 IPv4 address blocks. URL: https://en.wikipedia.org/wiki/List\_of\_assigned\_/8\_IPv4\_address\_blocks (дата обращения: 25.12.21).
7. Regional Internet registry. URL: https://en.wikipedia.org/wiki/Regional\_Internet\_registry (дата обращения: 25.12.21).
8. Classless Inter-Domain Routing. URL: https://en.wikipedia.org/wiki/Classless\_Inter-Domain\_Routing (дата обращения: 25.12.21).
9. Open Systems Interconnection. URL: https://ru.wikipedia.org/wiki/Open\_Systems\_Interconnection (дата обращения: 28.12.21).
10. Cisco - Википедия. URL: https://ru.wikipedia.org/wiki/Cisco (дата обращения: 29.12.21).
11. Сети для самых маленьких. Часть десятая. Базовый MPLS / Хабр. URL: https://habr.com/ru/post/246425/ (дата обращения: 03.01.22).
12. Канальный уровень. URL: http://altaev-aa.narod.ru/net/tests/images/lecture3.pdf (дата обращения: 04.01.22).
13. Цветовые схемы обжима (распиновки) utp кабеля витая пара (патч-корда) в вилке RJ-45. URL: https://ydoma.info/kompjuter/seti/kompjuter-cvetovaja-shema-obzhima-raspinovki-rj-45.html (дата обращения: 05.01.22).
14. Физический уровень Ethernet - Ethernet physical layer - abcdef.wiki. URL: https://ru.abcdef.wiki/wiki/Ethernet\_physical\_layer (дата обращения: 09.01.22).
15. WorldSkills - Википедия. URL: https://ru.wikipedia.org/wiki/WorldSkills (дата обращения: 09.01.22).
16. Как скачать и установить Cisco Packet Tracer. URL: https://wiki.merionet.ru/seti/50/ustanovka-i-zapusk-cisco-packet-tracer/ (дата обращения: 09.01.22).
17. Основы работы с Cisco Packet Tracer. URL: https://pc.ru/articles/osnovy-raboty-s-cisco-packet-tracer#interfejs\_cisco\_packet\_tracer (дата обращения: 09.01.22).
18. Урок 39. URL: Списки контроля доступа ACL (Access List Control). https://easy-network.ru/67-urok-39.html (дата обращения: 16.01.22).
19. ACL: списки контроля доступа в Cisco IOS / Хабр. URL: https://habr.com/ru/post/121806/ (дата обращения: 16.01.22).
20. Иерархия DNS серверов. URL: http://datanets.ru/ierarhiia-dns-serverov.html. (дата обращения: 16.01.22).
21. DNS - Википедия. URL: https://ru.wikipedia.org/wiki/DNS (дата обращения: 16.01.22).
22. hosts - Википедия. URL: https://ru.wikipedia.org/wiki/Hosts (дата обращения: 16.01.22).

# Приложение 1

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ**

**Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования   
«ВОСТОЧНО-СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ» (ФГБОУ ВПО ВСГУТУ)**

**Кафедра "Системы информатики"**

**курсовая работа   
по дисциплине «Компьютерные сети. Курсовая работа»**

**настройка сети**

**Вариант 53**

**Выполнил: студент <номер>группы**

**Фамилия И.О.**

**Проверил: Алтаев А.А.**

**Улан-Удэ**

**2022**

1. Не путать с WLAN (Wireless Local Area Network, локальная беспроводная сеть) [↑](#footnote-ref-2)
2. Denied of service – отказ в обслуживании [↑](#footnote-ref-3)
3. В протоколе IP вместо слова **байт** принято говорить **октет**. [↑](#footnote-ref-4)
4. URL - Uniform Resource Locator, Единый указатель ресурса [↑](#footnote-ref-5)
5. Если в сети применяются технология VLAN, то компьютеры могут быть распределены по разным сетям. В данной работе VLAN’ы не рассматриваются. [↑](#footnote-ref-6)
6. В данном случае рассматривается коммутатор уровня L2. Если настраивается коммутатор L3, то его порт может быть переведен в режим маршрутизации и ему в этом случае будет назначен IP адрес. Коммутаторы L3 в данной работе не рассматриваются. [↑](#footnote-ref-7)
7. Cisco начинала с производства одних лишь маршрутизаторов. Возможно, этим обстоятельством и объясняется слово «Межсетевая» в названии ОС. [↑](#footnote-ref-8)
8. MACd, условно - MAC address destination, адрес получателя и, соответственно, MACs – MAC address source, адрес отправителя. [↑](#footnote-ref-9)
9. HQ (Headquarters, Штаб-квартира) – условное обозначение головного офиса корпорации, Branch – соответственно, филиал организации. [↑](#footnote-ref-10)
10. source – источник. [↑](#footnote-ref-11)
11. destination – назначение. [↑](#footnote-ref-12)
12. FCS - Frame check sequence, контрольная сумма фрейма (32-битовая последовательность, записываемая отправителем в концевик фрейма). [↑](#footnote-ref-13)
13. Cyclic Redundancy Check - Циклический избыточный код. [↑](#footnote-ref-14)
14. Non Return to Zero - без возврата к нулю. [↑](#footnote-ref-15)
15. Read Only Memory [↑](#footnote-ref-16)
16. В курсовой работе данный вариант защиты не предусмотрен. [↑](#footnote-ref-17)
17. В разных версиях CPT интерфейс окон может быть разным. В данном случае используется CPT версии 6.2. [↑](#footnote-ref-18)
18. Команды выполняются в симуляторе CPT. При их выполнении на реальной ОС могут быть отличия. [↑](#footnote-ref-19)
19. О необходимости защиты паролем этой команды сообщалось в разделе «Сетевые устройства Cisco». [↑](#footnote-ref-20)
20. виртуальные линии позволяют получить удаленный доступ к маршрутизатору через протоколы Telnet и SSH. В данной работе используется Telnet, как менее сложный в настройке. [↑](#footnote-ref-21)
21. Пароли хранятся в конфигурационном файле в виде обычного читаемого текста. Команда **enable secret** обеспечивает хранение в файле не самого пароля, а его хеша, что обеспечивает высокую степень его защиты. [↑](#footnote-ref-22)
22. В данном случае коммутатор настраивается аналогично сетевой карте компьютера, благодаря чему он получает возможность отправлять и получать сообщения. [↑](#footnote-ref-23)
23. Интерфейс имеет своим синонимом слово «порт». В данном контексте портом мы называем не адрес транспортного уровня (см. раздел «Порты»), а интерфейс маршрутизатора, через который прибывают и убывают пакеты. [↑](#footnote-ref-24)
24. Адрес назначения. [↑](#footnote-ref-25)
25. публичных [↑](#footnote-ref-26)
26. Данный адрес будет применен в настройке NAT на пограничном маршрутизаторе R2 [↑](#footnote-ref-27)
27. Тупиковой является коммутируемая сеть, имеющая выход лишь на один маршрутизатор. В выполняемой работе широковещательные сети являются тупиковыми. [↑](#footnote-ref-28)
28. Данный адрес в настройках нигде не будет применен. [↑](#footnote-ref-29)
29. Отметим, что в коммутируемых сетях большого размера могут быть десятки коммутаторов. [↑](#footnote-ref-30)
30. В разделе «Приватная сеть» был вычислен адрес суперсети для приватной области в виде 10.224.0.0/11 [↑](#footnote-ref-31)