ФЕДЕРАЛЬНОЕ АГЕНТСТВО ВОЗДУШНОГО ТРАНСПОРТА

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ГРАЖДАНСКОЙ АВИАЦИИ» (МГТУ ГА)

Кафедра вычислительных машин, комплексов, систем и сетей

Н.И. Романчева

КОМПЬЮТЕРНЫЕ СЕТИ И ИНТЕРНЕТ-ТЕХНОЛОГИИ

Учебно-методическое пособие
по выполнению практических занятий

для студентов II курса
направления 25.03.02
очной формы обучения

Москва
ИД Академии Жуковского
2018

УДК 004.7:004.738.5(07)
ББК 6Ф7.3
Р69

Рецензент:
Терентьев А.И. – канд. техн. наук, доц. каф. ВМКСС

Романчева Н.И.

Р69 Компьютерные сети и интернет-технологии [Текст] : учебно-методическое пособие по выполнению практических занятий / Н.И. Романчева. – М. : ИД Академии Жуковского, 2018. – 36 с.

Данное учебно-методическое пособие издается в соответствии с рабочей программой учебной дисциплины «Компьютерные сети и интернет-технологии» по учебному плану для студентов II курса направления 25.03.02 очной формы обучения.

Рассмотрено и одобрено на заседании кафедры 15.02.2018 г. и методического совета 22.02.2018 г.

УДК 004.7:004.738.5(07)
ББК 6Ф7.3

В авторской редакции

Подписано в печать 26.04.2018 г.
Формат 60x84/16 Печ. л. 2,25 Усл. печ. л. 2,09
Заказ № 289/0403-УМП16 Тираж 40 экз.

Московский государственный технический университет ГА
125993, Москва, Кронштадтский бульвар, д. 20

Издательский дом Академии имени Н. Е. Жуковского
125167, Москва, 8-го Марта 4-я ул., дом 6А
Тел.: (495) 973-45-68
E-mail: zakaz@itsbook.ru

СОДЕРЖАНИЕ

1. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

1.1. Цель и задачи выполнения практических занятий

В соответствии с учебным планом подготовки обучающихся по направлению подготовки 25.03.02 «Техническая эксплуатация авиационных электросистем и пилотажно-навигационных комплексов» (бакалавриат) и рабочей программой по дисциплине «Компьютерные сети и интернет-технологии» и изложенными в них требованиями к уровню подготовки для работы в организациях ГА, студенты должны обладать практическими навыками и компетенциями в решении задач, связанных с организацией компьютерных сетей, работой сетевых протоколов и системой адресации, применением Интернет - приложений для решения профессиональных задач.

Особенностью данного пособия является его прикладная направленность, что способствует формированию у студентов общепрофессиональных компетенций:

ОПК-1 - Способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий с учетом основных требований информационной безопасности;

ОПК-7 - Способность использовать навыки работы с компьютером, владеть методами информационных технологий, соблюдать основные требования информационной безопасности.

Целью данного пособия является закрепления обучающимися теоретического курса дисциплины и приобретение навыков анализа сетевых протоколов, классификации и методически обоснования схемы построения компьютерных сетей, проведения диагностики сетевых соединений, - навыками организации маршрутизации в сетях.

1.2. Основные вопросы, подлежащие изучению

1. Уровни межсетевого взаимодействия;
2. Логика работы сетевых протоколов;
3. Адресация в IP-сетях;
4. Алгоритмы маршрутизации;
5. Средства мониторинга сети;
6. Изучение технологий, используемых в сети Интернет;

2. ПЕРЕЧЕНЬ ТЕМ ПРАКТИЧЕСКИХ ЗАНЯТИЙ (22 часа)

Практическое занятие 1. Инкапсуляция данных. Инкапсуляция на сетевом уровне (2 часа).

Практическое занятие 2. Сравнительный анализ моделей OSI и TCP/IP (2 часа).

Практическое занятие 3. Протоколы канального уровня: HDLC, LLC, PPP, семейство протоколов LAP (2часа).

Практическое занятие 4. Выбор конфигурации Fast Ethernet (2 часа)

Практическое занятие 5. Особенности протоколов IEEE 802.3, 802.11, 802.16. Реализация мостов (2 часа).

Практическое занятие 6,7. Алгоритмы вычисления наикратчайшего пути в сетях (4 часа).
Практическое занятие 8. Средства мониторинга сети. Утилиты тестирования сети (2 часа).
Практическое занятие 9,10 Адресация в IP-сетях. Преобразование адресов (4 часа).
Практическое занятие 11. Работа с удаленным компьютером на FTP-серверах, использование сервиса telnet для доступа к удаленному компьютеру (2 часа).

3. СОДЕРЖАНИЕ ЗАНЯТИЙ

В 4-м семестре учебным планом предусмотрено одиннадцать практических занятия продолжительностью 2 академических часа каждое.

3.1. Занятие 1. Инкапсуляция данных. Инкапсуляция на сетевом уровне (2 часа)

3.1.1. Цель занятия

- ознакомление с процессом инкапсуляции данных;
- изучение инкапсуляции данных в сетях X.25;
- изучение спецификации RFC 1613 «cisco Systems X.25 over TCP (XOT)».

3.1.2. Методические указания по теме

Метод инкапсуляции часто применяется, когда двум сетям, использующим один и тот же сетевой протокол, нужно связаться через транзитную сеть, которая работает с другими сетевыми протоколами. Примером могут служить две сети X.25, которым нужно связаться между собой через сеть TCP/IP.

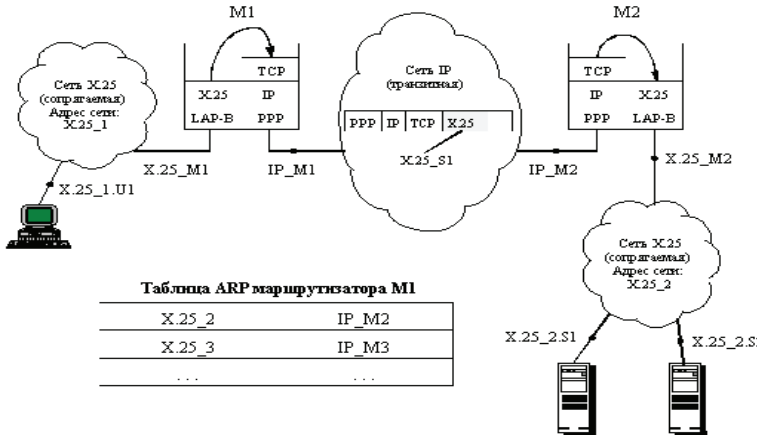


Рисунок 3.1- Соединение сетей X.25 через транзитную сеть TCP/IP методом инкапсуляции

В таких случаях сетевой протокол транзитной сети считается протоколом более низкого уровня, чем сетевой протокол объединяемых сетей (рис. 3.1).

Поэтому пакеты сетевого протокола сопрягаемых сетей X.25 инкапсулируются в пакеты TCP транспортного уровня транзитной сети TCP/IP пограничным маршрутизатором M1, и переносятся в пакетах TCP по транзитной сети до следующего пограничного маршрутизатора M2. Для переноса по сети TCP/IP пакеты TCP в соответствии с технологией этой сети помещаются в пакеты IP, которые инкапсулируются в кадры протокола канального уровня, например, PPP. Пограничный маршрутизатор M2 извлекает пакет X.25 из пакета TCP и отправляет его по сети X.25 в соответствии с правилами этого протокола - то есть предварительно установив виртуальное соединение с узлом назначения по адресу X.25_S1, а затем отправив по этому виртуальному соединению прибывший пакет.

Реализация такого подхода требует от протокола сетевого уровня объединяемых сетей разработанных процедур нахождения адреса пограничного маршрутизатора M2 в транзитной сети TCP/IP по адресу сети назначения протокола X.25. Обычно такие протоколы называют протоколами разрешения адресов - Address Resolution Protocol, ARP. Если бы такой протокол был определен для сети X.25, то он бы должен оперировать с ARP-таблицей, подобно той, которая приведена на рис. 3.2. Эта таблица содержит для каждого адреса сети назначения X.25 соответствующий IP-адрес пограничного маршрутизатора, через который эту сеть можно достичь.

К сожалению, многие сетевые протоколы разрабатывались или в расчете на работу только в сетях своего стека протоколов (например, протокол X.25), или в расчете на работу только через локальные сети (например, протокол IPX).

Наибольшую гибкость при инкапсуляции своих пакетов в пакеты других сетевых протоколов демонстрирует протокол IP. Для него разработано семейство протоколов ARP, каждый из которых предназначен для выполнения процедуры инкапсуляции пакетов IP в определенный протокол - Ethernet, X.25, frame relay, ATM и т.п.

Протокол X.25 рассчитан на инкапсуляцию только в кадры протокола LAP-B, который в свою очередь передает эти кадры между соседними коммутаторами, соединенными выделенными каналами "точка-точка". Процедуры и таблицы нахождения локальных адресов соседних коммутаторов при передачах пакетов X.25 не предусмотрены, так как в них нет надобности, если трафик не выходит за пределы сети X.25. Поэтому для передачи пакетов X.25 через сети с другим стеком протоколов процедуры разрешения адресов и форматы инкапсуляции необходимо определять отельным стандартом.

Похожая ситуация возникает при необходимости передачи пакетов протокола IPX через сети TCP/IP. Процедуры передачи пакетов через составные локальные сети в протоколе IPX предусмотрены. Однако, из-за того, что в протоколе IPX на сетевом уровне для задания адреса узла просто дублируется аппаратный адрес сетевого адаптера, процедуры отображения сетевого адреса узла на локальные адреса транзитных сетей здесь отсутствуют.

Поэтому, для протокола IPX, как и для протокола X.25, необходимо разработать дополнительный протокол для того, чтобы этот протокол мог использовать сети TCP/IP как транзитные.

Стандартный способ для передачи трафика сетей X.25 через сети TCP/IP предусмотрен в спецификации RFC 1613, имеющей название "*cisco Systems X.25 over TCP (XOT)*", разработанный сотрудниками компании cisco Systems и организации JANET.

Эта спецификация определяет способ инкапсуляции пакетов X.25 в сообщения TCP для переноса их по магистральной сети TCP/IP.

Так как протокол X.25 работает на основе установления соединения, то спецификация использует для инкапсуляции протокол TCP, который также работает с установлением соединения. Для каждого виртуального соединения X.25 маршрутизатор, который поддерживает стандарт XOT, устанавливает отдельное TCP-соединение с другим пограничным маршрутизатором. Новое TCP-соединение устанавливается при поступлении из сети X.25 служебного кадра Call Request, запрашивающего новое соединение и несущего X.25-адрес узла назначения. Спецификация XOT не предусматривает какого-либо способа определения IP-адреса маршрутизатора-партнера по сети IP, поэтому наиболее целесообразно использовать ее для случая, когда такой партнер один, или же маршрутизатор должен иметь таблицу соответствия адресов X.25 и IP-адресов маршрутизаторов-партнеров, сформированную вручную.

После установления TCP-соединения все пакеты X.25, принадлежащие данному виртуальному соединению сети X.25, передаются в сообщениях TCP, принадлежащих этому TCP-соединению. Так как протокол TCP ориентирован на передачу неструктурированного потока байт, то стандарт XOT имеет небольшой заголовок, состоящий из 4-х байт, для выделения пакетов X.25 в потоке байт сообщений TCP.

Для туннелирования пакетов IPX через магистрали TCP/IP компания Novell предложила спецификацию "*Tunneling IPX Traffic through IP Networks*", которая была утверждена в качестве стандарта RFC 1234.

Спецификация использует инкапсуляцию пакетов IPX в пакеты протокола UDP. Это объясняется тем, что протокол IPX является дейтаграммным протоколом, поэтому удобнее для его транспортировки использовать также дейтаграммный протокол UDP.

Транзитная IP-сеть вместе с объединяемыми IPX сетями рассматривается в этой спецификации как одна IPX-сеть, поэтому узлы по разные стороны IP-сети должны иметь один и тот же сетевой IPX-адрес.

Спецификация RFC 1234 предлагает автоматизировать процесс поиска маршрутизатора-партнера в сети IP за счет прямого преобразования MAC-адреса узла на фиктивный IP-адрес этого же узла. Адрес узла в приходящем для транзитной передачи IPX-пакете состоит из 6 байт. Для получения IP-адреса назначения пакета, который будет переносить данный IPX-пакет, старшие два

байта адреса узла отбрасываются, а оставшиеся 4 рассматриваются как IP-адрес.

Пограничный IP-маршрутизатор должен иметь в своей таблице маршрутизации все IP-адреса узлов сети-партнера IPX, и для этих адресов должен быть указан в качестве IP-адреса следующего маршрутизатора IP-адрес пограничного маршрутизатора с другой стороны транзитной сети. IP-адреса узлов должны иметь маску 255.255.255.255, чтобы номер узла рассматривался как номер сети. В этом случае эти произвольные IP-адреса не будут конфликтовать с легальными номерами сетей в Internet.

Широковещательные пакеты сервиса SAP предлагается распространять на индивидуальной основе, то есть путем дублирования пакета всем узлам сети IPX через сеть IP.

3.1.3. Задания для самостоятельного решения

Задание 1.

Преобразовать согласно спецификации RFC 1234 (“Tunelling IPX Traffic through IP Networks”) MAC-адрес узла в фиктивный IP-адрес того же узла. (Маска подсети – 255.255.255.255)

00-1C-24-1A-FF-67

Задание 2.

Преобразовать согласно спецификации RFC 1234 (“Tunelling IPX Traffic through IP Networks”) MAC-адрес узла в фиктивный IP-адрес того же узла. (Маска подсети – 255.255.255.255)

00-F1-17-65-AB-11

Задание 3

Преобразовать согласно спецификации RFC 1234 (“Tunelling IPX Traffic through IP Networks”) MAC-адрес узла в фиктивный IP-адрес того же узла. (Маска подсети – 255.255.255.255)

00-DD-1A-62-1B-03

Задание 4

Преобразовать согласно спецификации RFC 1234 (“Tunelling IPX Traffic through IP Networks”) MAC-адрес узла в фиктивный IP-адрес того же узла. (Маска подсети – 255.255.255.255)

00-17-A6-49-02-1C

Задание 5

Преобразовать согласно спецификации RFC 1234 (“Tunelling IPX Traffic through IP Networks”) MAC-адрес узла в фиктивный IP-адрес того же узла. (Маска подсети – 255.255.255.255)

00-39-1C-7A-04-15

Контрольные вопросы

1) Что понимается под инкапсуляцией?
2) Какие протоколы называют протоколами разрешения адресов?
3) Что предлагается в спецификация RFC 1234?

4) Как предлагается распространять широковещательные пакеты сервиса SAP?

3.2. Занятие 2. Сравнительный анализ моделей OSI и TCP/IP (2 часа)

3.2.1. Цель занятий

- изучение уровней эталонной модели взаимодействия сетей;
- получение навыков классификации функций, реализуемых в моделях.

3.2.2. Методические указания по теме

Для выполнения заданий по данной теме необходимо заранее изучить разделы 3.2-3.5[2]. Ниже кратко приведен сравнительный анализ моделей.

Семейство протоколов TCP/IP может быть описано с точки зрения эталонной модели OSI. В модели взаимодействия открытых систем (OSI) уровень доступа к сети и уровень приложений в модели TCP/IP дополнительно подразделяются для описания отдельных функций, которые реализуются на этих уровнях. На уровне доступа к сети семейство протоколов TCP/IP не определяет список протоколов, используемых для передачи по физической среде, описывая только передачу с межсетевого уровня физическим сетевым протоколам. Уровни 1 и 2 модели OSI описывают необходимые процедуры для доступа к среде передачи и физическим средствам отправки данных по сети. Полное совпадение двух сетевых моделей происходит на уровнях 3 и 4 модели OSI. Уровень 3 используется для описания ряда процессов, которые возникают во всех сетях передачи данных при адресации и маршрутизации сообщений в объединённой сети. Протокол IP реализует функциональность, описанную на третьем уровне модели OSI. Четвертый уровень OSI описывает общие сервисы и функции, которые предоставляют упорядоченную и надёжную доставку данных между узлами источника и назначения. На этом уровне основные функции обеспечивают протоколы TCP и UDP семейства TCP/IP. Уровни 5, 6 и 7 модели OSI используются в качестве ссылки для разработчиков и поставщиков прикладного программного обеспечения в производстве сетевой продукции.

Таким образом, в отличие от эталонной модели OSI, модель TCP/IP в большей степени ориентируется на обеспечение сетевых взаимодействий, чем на жесткое разделение функциональных уровней. Для этой цели признается важность иерархической структуры функций, но предоставляет проектировщикам протоколов достаточную гибкость в реализации.

3.2.3. Задания для самостоятельного решения

Задание 1.

Постройте структуру данных на каждом уровне Protocol Data Unit.

Задание 2.

В приведенном ниже формате заголовка сегмента TCP, укажите поле, в котором индицируется конец срочных данных:

Задание 3.

В заголовке сегмента TCP в поле Код необходимо указать:

- сегмент установки соединения;
- сегмент завершения сеанса;
- сегмент подтверждения принятых данных;
- срочного сообщения;
- протолкнуть данные;
- оборвать соединение.

Задание 4.

В приведенном ниже формате заголовка UDP, укажите поле, в котором указывается число байтов в заголовке и в поле данных; контрольная сумма заголовка и поля данных:

Контрольные вопросы

1) Поясните отличия стека протоколов TCP/IP от модели OSI.
2) Что стандартизирует модель OSI?
3) На каком из уровней модели OSI будет инкапсулирован логический адрес?
4) Какие три протокола уровня приложений входят в стек протоколов TCP/IP ?
5) Какой уровень описывает процессы, возникающие во всех сетях передачи данных при адресации и маршрутизации сообщений в объединённой сети?

3.3. Занятие 3. Протоколы канального уровня: HDLC, LLC, PPP, семейство протоколов LAP (2 часа)

3.3.1. Цель занятий

- ознакомление с протоколами канального уровня;
- получение навыков манчестерского и разностного манчестерского кодирования.

3.3.2. Методические указания по теме

Для создания надежного механизма передачи данных между двумя

станциями необходимо определить протокол, который позволит принимать и передавать различные данные по каналам связи. Протоколы представляют собой просто набор условий (правил), которые регламентируют формат и процедуры обмена информацией между двумя или несколькими независимыми устройствами или процессами. Протокол имеет три важнейших элемента: синтаксис, семантику и синхронизацию. Синтаксис протокола определяет поля; например, может быть 16-байтовое поле для адресов, 32-байтовое поле для контрольных сумм и 512 байт на пакет. Семантика протокола придает этим полям значение: например, если адресное поле состоит из всех адресов, это «широковещательный» пакет. Синхронизация - количество битов в секунду - это скорость передачи данных. Она важна не только на самых низких уровнях протокола, но и на высших. Протокол высокоуровнего управления каналом передачи данных HDLC (High-Level Data Link Control) является протоколом канального уровня (бит-ориентированный) модели ISO и является базовым для построения других протоколов канального уровня (SDLC, LAP, LAPB, LAPD, LAPX и LLC). Основные принципы работы протокола HDLC: режим логического соединения, контроль искаженных и потерянных кадров с помощью метода скользящего окна, управление потоком кадров с помощью команд RNR (приемник не готов) и RR (приемник готов). Более подробная информация изложена в [1].

Способы кодирования данных зависят от носителя, диапазона частот, внешних ограничений (например, качества линии связи), типа линии (аналоговая или цифровая). Модем - устройство, принимающее последовательный поток битов и преобразующее его в выходной модулированный сигнал, а также выполняющее обратное преобразование

Частота дискретизации - количество отсчетов сигнала, которые выполняет модем. Число отсчетов (сэмплов) в секунду измеряется в бодах. Нельзя увеличить скорость передачи простым увеличением частоты дискретизации. На практике большинство модемов делают 2400 отсчетов сигнала в секунду. Стремятся не к повышению этого значения, а к повышению числа битов на отсчет.

В сетях Ethernet не используют модулирование какого-либо аналогового сигнала: сигнал формируется изначально цифровой; 0 и 1 кодируются различным напряжением, но не просто напряжением, а его переходами.

Если при кодировании логического 0 нулем вольт, а логической 1 двумя вольтами образовалась последовательность в несколько нулей, как мы узнаем, сколько их? Возникает проблема синхронизации передачи данных, следовательно, нельзя менять скорость передачи. Для преодоления этих проблем используют манчестерский код: ноль кодируют переходом напряжения от низкого значения к высокому; единицу кодируют обратным переходом: от высокого значения к низкому. Во всех сетях Ethernet используют манчестерское кодирование из-за его простоты. Такая схема гарантирует смену

напряжения в середине периода битов, что позволяет приемнику синхронизироваться с передатчиком. Недостатком манчестерского кодирования является то, что оно требует двойной пропускной способности линии по отношению к прямому двоичному кодированию, так как импульсы имеют половинную ширину. Например, для того чтобы отправлять данные со скоростью 10 Мбит/с, необходимо изменять сигнал 20 миллионов раз в секунду. Манчестерское кодирование показано ниже, на рис.3.2 (б).

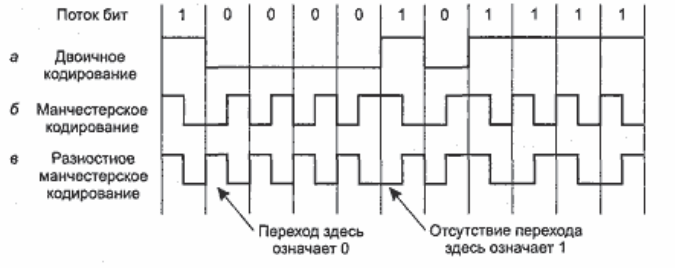


Рисунок 3.2 –Примеры кодирования

Разностное манчестерское кодирование, показанное на рис.3.2 (в), является вариантом основного манчестерского кодирования. В нем бит 0 кодируется изменением состояния в начале интервала, а бит 1 -сохранением предыдущего уровня. В обоих случаях в середине интервала обязательно присутствует переход. Разностная схема требует более сложного оборудования, зато обладает хорошей защищенностью от шума. Разностное манчестерское кодирование в Ethernet не используется, но используется в других ЛВС (например, стандарт 802.5, маркерное кольцо).

3.3.3. Задания для самостоятельного решения

Преобразовать нижеприведенные варианты исходной последовательности двоичного кода в манчестерский код и разностный манчестерский код:

Задание 1

Исходная последовательность	0	0	0	0	1	0	1	0	0	1	1	1	0
Двоичное кодирование													
Манчестерский код													
Разностный манчестерский код													

Задание 2

Исходная последовательность	1	1	0	1	0	1	1	0	0	0	1	1	1
Двоичное кодирование													
Манчестерский код													

Разностный манчестерский код	

Задание 3

Исходная последовательность	0	1	0	1	0	1	1	1	1	0	1	1	1
Двоичное кодирование													
Манчестерский код													
Разностный манчестерский код													

Задание 4

Исходная последовательность	0	0	0	1	1	1	1	0	0	0	1	1	1
Двоичное кодирование													
Манчестерский код													
Разностный манчестерский код													

Задание 5

Исходная последовательность	0	1	0	0	0	0	1	1	0	1	0	0	0
Двоичное кодирование													
Манчестерский код													
Разностный манчестерский код													

Контрольные вопросы

1) Назначение протокола и его основные элементы?
2) Какие основные функции выполняет канальный протокол?
3) Основные принципы работы протокола HDLC.
4) От каких параметров зависят способы кодирования данных?
5) Проясните использование манчестерского и разностного манчестерского кодирования.

3.4. Занятие 4. Выбор конфигурации Fast Ethernet (2 часа)

3.4.1. Цель занятий

приобретение навыков:

- анализа характеристик кабелей, как среды передачи данных;
- классификации и методически обоснования схемы построения компьютерных сетей.

3.4.2. Методические указания по теме

Для определения работоспособности сети Fast Ethernet стандарт IEEE 802.3 предлагает две модели: Transmission System Model 1 и Transmission System Model 2. TSM1 основана на следующих правилах: все компоненты сети (в частности, кабели) имеют наихудшие из возможных временные

характеристики, поэтому всегда дает результат со значительным запасом. TSM2 использует систему точных расчетов с реальными временными характеристиками кабелей. В связи с этим ее применение позволяет иногда преодолеть жесткие ограничения TSM1.

Правила TSM1.

- Сегменты, выполненные на электрических кабелях (витых парах) не должны быть длиннее 100 метров. Это относится к кабелям всех категорий – 3, 4 и 5, к сегментам 100BASE-T4 и 100BASE-TX.

- Сегменты, выполненные на оптоволоконных кабелях, не должны быть длиннее 412 метров.

- Если используются адаптеры с внешними (выносными) трансиверами, то трансиверные кабели (MII) не должны быть длиннее 50 сантиметров.

TSM1выделяет три возможные конфигурации сети Fast Ethernet:

- Соединение двух абонентов (узлов) сети напрямую, без репитера или концентратора. Абонентами при этом могут выступать не только компьютеры, но и сетевой принтер, порт коммутатора, моста или маршрутизатора. Такое сопряжение называется соединением DTE—DTE или двухточечным.

- Соединение двух абонентов сети с помощью одного репитерного концентратора класса I или класса II .

- Соединение двух абонентов сети с помощью двух репитерных концентраторов класса II. При этом предполагается, что для связи концентраторов всегда используется электрический кабель длиной не более 5 метров. Концентраторы класса II имеют меньшую задержку, поэтому их может быть два. Использование трех концентраторов в соответствии с моделью 1 не допускается.

В случае выбора первой конфигурации (двухточечной) правила модели 1 предельно просты: электрический кабель не должен быть длиннее 100 метров, полудуплексный оптоволоконный – не более 412 метров, полнодуплексный оптоволоконный – 2000 метров (при этом задержка сигнала в кабеле не имеет значения, так как метод CSMA/CD не работает).

В случае применения второй конфигурации (с одним концентратором) надо ограничивать длину кабелей A и B сети в соответствии с табл.1. В случае выбора третьей конфигурации сети (с двумя концентраторами) надо ограничивать длину кабелей A и B в соответствии с табл.2. При этом по умолчанию предполагается, что кабель C имеет длину 5 метров. В обеих конфигурациях с концентраторами при использовании одновременно электрического и оптоволоконного кабелей можно за счет уменьшения длины электрического кабеля увеличить длину оптоволоконного. Причем уменьшению длины электрического кабеля на 1 метр соответствует увеличение длины оптоволоконного кабеля на 1,19 метра. Например, уменьшив кабель TX на 10 метров, можно увеличить кабель FX на 11,9 метра, и его предельная длина составит при двух концентраторах 128,1 метра. Немного увеличится и предельный размер сети (в нашем примере на 1,9 метра).

Таблица 1. Максимальная длина кабелей в конфигурации с одним концентратором

Вид кабеля А	Вид кабеля В	Класс концентратора	Макс. длина кабеля А, м	Макс. длина кабеля В, м	Макс. размер сети, м
TX, T4	TX, T4	I или II	100	100	200
TX	FX	I	100	160,8	260,8
T4	FX	I	100	131	231
FX	FX	I	136	136	272
TX	FX	II	100	208,8	308,8
T4	FX	II	100	204	304
FX	FX	II	160	160	320

Таблица 2. Максимальная длина кабелей в конфигурации с двумя концентраторами

Вид кабеля А	Вид кабеля В	Макс. длина кабеля А, м	Макс. длина кабеля В, м	Макс. размер сети, м
TX, T4	TX, T4	100	100	205
TX	FX	100	116,2	221,2
T4	FX	136,3	136,3	241,3
FX	FX	114	114	233

3.4.3. Задания для самостоятельного решения

Задание 1.

Имеется конфигурация из пяти сегментов 100BASE-T предельно допустимой длины (100 метров), соединенных между собой четырьмя концентраторами. Определить суммарную задержку. Задержки начального, конечного и трех промежуточных сегментов - в соответствии с табл.1.

Задание 2.

Подсчитать величину сокращения межпакетного интервала при конфигурации, описанной в задании 1 и с добавлением двух 100-метровых промежуточных сегментов.

Задание 3.

Имеется конфигурация из трех сегментов 100BASE-FX предельно допустимой длины (114 метров), соединенных между собой двумя концентраторами. Определить суммарную задержку. Задержки начального, конечного и двух промежуточных сегментов - в соответствии с табл.2.

Контрольные вопросы

1) Поясните основные модели расчета конфигурации Fast Ethernet.
2) Поясните суть метода управления обменом CSMA/CD.
3) Что понимается под термином «зона конфликта»?
4) Поясните условия, при которых сеть будет работоспособной.

3.5. Занятия 5. Особенности протоколов IEEE 802.3, 802.11, 802.16. Реализация мостов (2 часа)

3.5.1. Цель занятий

- выявление особенности протоколов IEEE 802.3, 802.11, 802.16;
- получение навыков реализации мостов.

3.5.2. Методические указания по теме

IEEE 802.3 - Стандарт на метод коллективного доступа для локальных сетей CSMA/CD и на физический уровень. Он положен в основу ISO/IEC 8802-3. Иногда его называют стандартом Ethernet. **IEEE 802.11** - Стандарт на уровень MAC и спецификации физического уровня для беспроводных локальных сетей. Предлагаемый проект рассчитан на диапазон 2,4 ГГц. В протоколе 802.11 используется алгоритм доступа CSMA/CA (CSMA with Collision Avoidance). При этом производится прослушивание физического и виртуального каналов. CSMA/CA может работать в двух режимах. В первом - станция до начала передачи прослушивает канал. Если канал свободен, она начинает передачу данных. При передаче канал не прослушивается, и станция передает кадр полностью. Если канал занят, отправитель ждет его освобождения и только после этого начинает передачу. В случае коллизии станции, участвующие в этом событии, смогут начать передачу через псевдослучайный интервал времени (как в Ethernet). Второй режим CSMA/CA базируется на протоколе MACAW и использует контроль виртуального канала. **IEEE 802.16** - Стандарт широкополосной беспроводной связи. Безопасность в сети обеспечивается на уровне протокола 3-DES. В отличие от 802.11 он ориентирован для соединения стационарных, а не мобильных объектов. Его задачей является обеспечения сетевого уровня между локальными сетями (IEEE 802.11) и региональными сетями (WAN), где планируется применение разрабатываемого стандарта IEEE802.20. Эти стандарты совместно со стандартом IEEE 802.15 (PAN - Personal Area Network - Bluetooth) и 802.17 (мосты уровня MAC) образуют взаимосогласованную иерархию протоколов беспроводной связи. Стандарт покрывает диапазон частот от 2 до 11 ГГц. Базовая станция BS, следующая стандарту 802.16, размещается в здании или на вышке и осуществляет связь со станциями клиентов (SS - Subscriber Station) по схеме точка-мультиточка(PMP). Возможен сеточный режим связи (Mesh - сетка связей точка-точка - PTP).

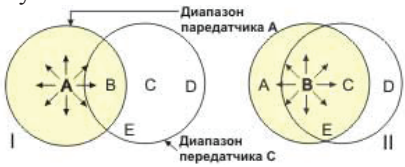
Мосты обеспечивают соединение двух или более локальных сетей (ЛВС) для образования единой логической сети. Исходные сети становятся при этом сегментами результирующей ЛВС. Сетевые мосты относятся к устройствам второго уровня сетевой модели OSI, которые способны распознавать приходящий на их порт сигнал, декапсулировать и перенаправлять последний на активные и соответствующие порты. Мосты выполняют три важных функции – анализ, фильтрация и пересылка. Основное назначение моста - организация обмена между сетями с разными стандартами обмена, например Ethernet, Token Ring, FDDI и т.д., а также между несколькими сегментами одной сети с целью разделения их нагрузок. Мосты используют два типа алгоритмов: алгоритм прозрачного моста (transparent bridge), либо алгоритм

моста с маршрутизацией от источника (source routing bridge). Прозрачные мосты незаметны для сетевых адаптеров конечных узлов, т.к. они самостоятельно строят специальную адресную таблицу, на основании которой можно решить, нужно передавать пришедший кадр в какой-либо другой сегмент или нет. Сетевые адаптеры работают точно так же, как при отсутствии прозрачного моста, т.е. не предпринимают никаких дополнительных действий, чтобы кадр прошёл через мост. Алгоритм прозрачного моста не зависит от технологии локальной сети, в которой устанавливается мост. Прозрачный мост строит свою адресную таблицу на основании пассивного наблюдения за трафиком, циркулирующим в подключённых к его портам сегментах. При этом мост учитывает адреса источников кадров данных, поступающих на порты моста. По адресу источника кадра мост делает вывод о принадлежности этого узла к тому или иному сегменту сети.

3.5.3. Задания для самостоятельного решения

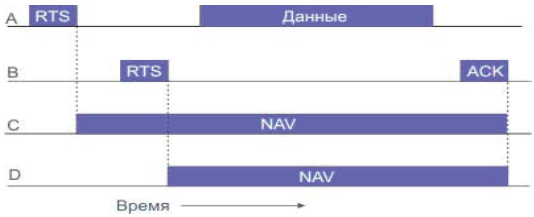
Задание 1.

Опишите схему взаимодействия узлов в беспроводной сети MACA: 1) Если передачу осуществляет узел A и узел C находится вне его радиуса действия; 2) если передачу ведет узел B.



Задание 2.

Опишите алгоритм прослушивание виртуального канала в протоколе CSMA/CA. Станция A намеревается передать данные станции B Станция C находится в зоне доступности стации A и, возможно, станции B. Станция D входит в зону доступности станции B, но пребывает в зоне зоны досягаемости станции A.



Задание 3.

Сегментировать локальную сеть для ликвидации узких мест в сетевом трафике. Схема локальной сети выдается индивидуально преподавателем.

Задание 4.

Сегмент Ethernet имеет 30 узлов, т.е. достигнут лимит на максимальное количество соединений. Разработать структуру, расширяющую локальную сеть.

Задание 5.

Расширить локальную сеть и обойти ограничения на длину сегментов, если нужно нарастить сегмент Ethernet на тонком кабеле, который уже имеет длину 185 м.

Контрольные вопросы

1) Поясните особенности протоколов IEEE 802.3, 802.11, 802.16.

2) Для чего используются мосты? Приведите их классификацию.

3) Поясните отличия алгоритмов transparent bridge и source routing bridge.

4) Поясните принцип построения адресной таблицы прозрачного моста

5) Какой алгоритм доступа используется в протоколе 802.11?

3.6. Занятия 6-7. Алгоритмы вычисления наикратчайшего пути в сетях (4 часа)

3.6.1. Цель занятий

- ознакомление алгоритмами маршрутизации;
- получение навыков построения маршрутов по заданному критерию.

3.6.2. Методические указания по теме

Изучение алгоритмов маршрутизации начнем со статического алгоритма, широко используемого на практике в силу его простоты. Идея этого алгоритма состоит в построении графа транспортной среды, где вершины - маршрутизаторы, а дуги - линии связи. Алгоритм находит для любой пары маршрутизаторов, а точнее абонентов, подключенных к этим маршрутизаторам, наикратчайший маршрут в этом графе.

Проиллюстрируем идею алгоритма нахождения наикратчайшего пути на рис. 3.3 (стрелками обозначены задействованные узлы).

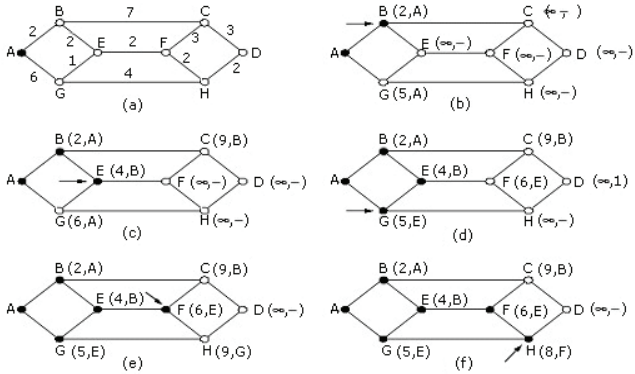


Рисунок 3.3 - Расчет кратчайшего пути от А к В

На дугах этого графа указаны веса, которые представляют расстояние между дугами. Расстояние можно измерять в переходах, а можно в километрах. Возможны и другие меры. Например, дуги графа могут быть размечены весами,

величина которых равна средней задержке пакетов в соответствующем канале. В графе с такой разметкой наикратчайший путь - наибыстрейший путь, хотя он не обязательно имеет минимальное число переходов или километров.

В общем случае веса на дугах могут быть функциями от расстояния, пропускной способности канала, среднего трафика, стоимости передачи, средней длины очереди в буфере маршрутизатора к данному каналу и других факторов. Изменяя весовую функцию, алгоритм будет вычислять наикратчайший путь в смысле заданной метрики.

Известно несколько алгоритмов вычисления наикратчайшего пути в графе. Один из них предложил голландский математик Эдсгер Дейкстра. Идею этого алгоритма заключается в следующем. Все вершины в графе, смежные исходной вершине, помечают расстоянием (оно указано в скобках) до исходной вершины. Изначально никакие пути не известны и все вершины помечены бесконечностью. По мере работы алгоритма и нахождения путей, метки могут меняться. Метки могут быть двух видов: либо пробными, либо постоянными. Изначально все метки пробные. Когда обнаруживается, что метка представляет наикратчайший путь до исходной вершины, она превращается в постоянную метку и никогда более не меняется.

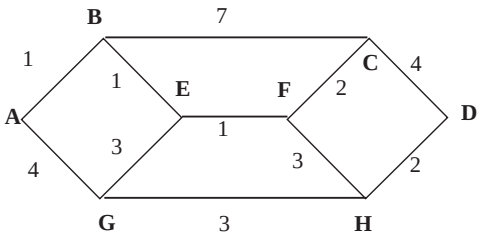
На рис.3.6 показан процесс построения маршрута из A в D. Помечаем вершину A как постоянную (вершина, закрашенная черным цветом). Все вершины, смежные A, помечаем как временные (эти вершины не закрашены), а также указываем в метке их вершину, из которой апробировали данную вершину. Это позволит впоследствии изменить маршрут, если надо. Кроме этого, все вершины, смежные A, помечаем расстоянием от A до этой вершины. Из всех смежных вершин выберем ту, расстояние до которой самое короткое, и ее объявляем рабочей. Таким образом, выберем на первом шаге вершину B, а затем E.

Особенности возникают на шаге (d). В соответствии с принципом наикратчайшего пути в качестве рабочей выберем вершину G. Теперь, на шаге (e), когда начнем искать вершины, смежные H, то увидим, что путь F до H короче, чем от G до H. Поэтому на шаге (e) в качестве рабочей возьмем вершину F, а затем H.. Надо сделать оговорку, что этот алгоритм строит наикратчайший путь, начиная от точки доставки, а не от точки отправления. Поскольку граф не ориентированный, то это никакого влияния на построение пути не оказывает.

3.6.3. Задания для самостоятельного решения

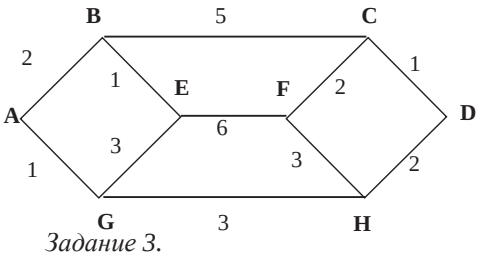
Задание 1.

Применить алгоритм наикратчайшего пути для маршрутизации пакета из точки A в точку D.



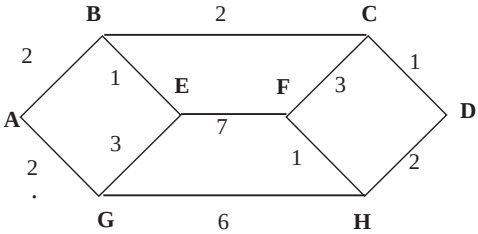
Задание 2.

Применить алгоритм наикратчайшего пути для маршрутизации пакета из точки A в точку D.



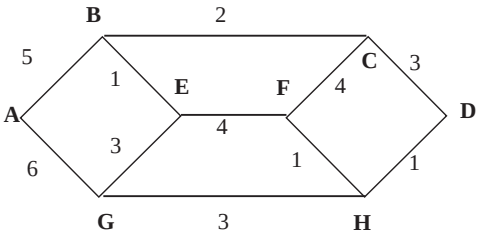
Задание 3.

Применить алгоритм наикратчайшего пути для маршрутизации пакета из точки A в точку D.



Задание 4.

Применить алгоритм наикратчайшего пути для маршрутизации пакета из точки A в точку D.



Контрольные вопросы

1) Что понимается под маршрутизацией?
2) Перечислите типичные случаи межсетевых подключений.
3) Перечислите основные виды и протоколы маршрутизации.
4) В чем отличие статической и динамической маршрутизации?
5) Поясните формат вывода таблицы маршрутов.
6) Поясните алгоритмы измерения кратчайшего пути

3.7. Занятия 8. Средства мониторинга сети. Утилиты тестирования сети (2 часа)

3.7.1. Цель занятий

- ознакомление со средствами мониторинга сети;
- получение навыков работы с утилитами TCP/IP.

3.7.2. Методические указания по теме

В состав TCP/IP входят диагностические утилиты, предназначенные для проверки конфигурации стека и тестирования сетевого соединения:

hostname	Выводит имя локального хоста. Используется без параметров.
ipconfig	Выводит значения для текущей конфигурации стека TCP/IP: IP-адрес, маску подсети, адрес шлюза по умолчанию, адреса WINS (Windows Internet Naming Service) и DNS (Domain Name System)
nbtstat	Выводит статистику и текущую информацию по NetBIOS, установленному поверх TCP/IP. Используется для проверки состояния текущих соединений NetBIOS.
netstat	Выводит статистику и текущую информацию по соединению TCP/IP
nslookup	Осуществляет проверку записей и доменных псевдонимов хостов, доменных сервисов хостов, а также информации операционной системы, путем запросов к серверам DNS.
ping	Осуществляет проверку правильности конфигурирования TCP/IP и проверку связи с удаленным хостом.
route	Модифицирует таблицы маршрутизации IP. Отображает содержимое таблицы, добавляет и удаляет маршруты IP.
tracert	Осуществляет проверку маршрута к удаленному компьютеру путем отправки эхо-пакетов протокола ICMP (Internet Control Message Protocol). Выводит маршрут прохождения пакетов на удаленный компьютер.

При устранении неисправностей и проблем в сети TCP/IP следует сначала проверить правильность конфигурации TCP/IP. Для этого используется утилита ipconfig. Данная команда полезна на компьютерах, работающих с DHCP (Dynamic Host Configuration Protocol), так как дает пользователям возможность определить, какая конфигурация сети TCP/IP и какие величины были установлены с помощью DHCP.

Синтаксис:

ipconfig [/all | /renew[adapter] | /release]

Параметры:

all - выдает весь список параметров. Без этого ключа отображается только IP-адрес, маска и шлюз по умолчанию;

renew[adapter]

обновляет параметры конфигурации DHCP для указанного сетевого адаптера;

release[adapter] -

освобождает выделенный DHCP IP-адрес; adapter – имя сетевого адаптера;

displaydns - выводит информацию о содержимом локального кэша клиента DNS, используемого для разрешения доменных имен.

Таким образом, утилита ipconfig позволяет выяснить, инициализирована ли конфигурация и не дублируются ли IP-адреса:

- если конфигурация инициализирована, то появляется IP-адрес, маска, шлюз;

- если IP-адреса дублируются, то маска сети будет 0.0.0.0;

- если при использовании DHCP компьютер не смог получить IP-адрес, то он будет равен 0.0.0.0 .

Утилита ping (Packet Internet Grouper) используется для проверки конфигурирования TCP/IP и диагностики ошибок соединения. Она определяет доступность и функционирование конкретного хоста. Использование ping позволяет проверить факт, что между локальным компьютером и сетевым хостом существует маршрут. Хостом называется любое сетевое устройство (компьютер, маршрутизатор), обменивающееся информацией с другими сетевыми устройствами по TCP/IP.

Команда ping проверяет соединение с удаленным хостом путем посылки к этому хосту эхо-пакетов ICMP и прослушивания эхо-ответов. Ping ожидает каждый посланный пакет и печатает количество переданных и принятых пакетов. Каждый принятый пакет проверяется в соответствии с переданным сообщением. Если связь между хостами плохая, из сообщений ping станет ясно, сколько пакетов потеряно.

По умолчанию передается 4 эхо-пакета длиной 32 байта (периодическая последовательность символов алфавита в верхнем регистре). Ping позволяет изменить размер и количество пакетов, указать, следует ли записывать маршрут, который она использует, какую величину времени жизни (ttl) устанавливать, можно ли фрагментировать пакет и т.д..При получении ответа в поле time указывается, за какое время (в миллисекундах) посланный пакет доходит до удаленного хоста и возвращается назад. Так как значение по умолчанию для ожидания отклика равно 1 секунде, то все значения данного поля будут меньше 1000 миллисекунд. Если получаете сообщение «Request time out» (Превышен интервал ожидания), то, возможно, если увеличить время ожидания отклика, пакет дойдет до удаленного хоста. Это можно сделать с помощью ключа –w.

Ping можно использовать для тестирования как имени хоста (DNS или NetBIOS), так и его IP-адреса. Если ping с IP-адресом выполнилась успешно, а с именем – неудачно, это значит, что проблема заключается в распознавании соответствия адреса и имени, а не в сетевом соединении.

Утилита ping используется следующими способами:

1) Для проверки того, что TCP/IP установлен и правильно сконфигурирован на локальном компьютере, в команде ping задается адрес петли обратной связи (loopback address):

ping 127.0.0.1

Если тест успешно пройден, то вы получите следующий ответ:

Reply from 127.0.0.1
Reply from 127.0.0.1
Reply from 127.0.0.1
Reply from 127.0.0.1

2) Чтобы убедиться в том, что компьютер правильно добавлен в сеть и IP-адрес не дублируется, используется IP-адрес локального компьютера:

ping IP-адрес_локального_хоста

3) Чтобы проверить, что шлюз по умолчанию функционирует и что можно установить соединение с любым локальным хостом в локальной сети, задается IP-адрес шлюза по умолчанию:

ping IP-адрес_шлюза

4) Для проверки возможности установления соединения через маршрутизатор в команде ping задается IP-адрес удаленного хоста:

ping IP-адрес_удаленного хоста

Синтаксис утилиты ping:

ping [-t] [-a] [-n count] [-l length] [-f] [-i ttl]
[-v tos] [-r count] [-s count] [[-j host-list] |
[-k host-list]] [-w timeout] destination-list

Параметры:

-t - выполняет команду ping до прерывания. Control-Break - посмотреть статистику и продолжить. Control-C - прервать выполнение команды;

-a - позволяет определить доменное имя удаленного компьютера по IP-адресу;

-n count - посылает количество пакетов ECHO, указанное параметром count;

-l length - посылает пакеты длиной length байт (максимальная длина 8192 байта);

-f - посылает пакет с установленным флагом «не фрагментировать». Этот пакет не будет фрагментироваться на маршрутизаторах по пути своего следования;

-i ttl - устанавливает время жизни пакета в величину TTL (каждый маршрутизатор уменьшает ttl на единицу);

-v tos - устанавливает тип поля «сервис» в величину tos;

-r count - записывает путь выходящего пакета и возвращающегося пакета в поле записи пути. Count - от 1 до 9 хостов;

-s count - позволяет ограничить количество переходов из одной подсети в другую (хопов). Count задает максимально возможное количество хопов;

-j host-list - направляет пакеты с помощью списка хостов, определенного параметром host-list. Последовательные хосты могут быть отделены промежуточными маршрутизаторами (гибкая статическая маршрутизация). Максимальное количество хостов в списке, позволенное IP, равно 9;

-k host-list - направляет пакеты через список хостов, определенный в host-list. Последовательные хосты не могут быть разделены промежуточными маршрутизаторами (жесткая статическая маршрутизация). Максимальное количество хостов – 9;

-w timeout - указывает время ожидания (timeout) ответа от удаленного хоста в миллисекундах (по умолчанию – 1сек);

destination-list - указывает удаленный хост, к которому надо направить пакеты ping.

Пример использования утилиты ping.

C:\WINDOWS>ping –n 10 www.netscape.com
Обмен пакетами с www.netscape.com [205.188.247.65] по 32 байт:
Ответ от 205.188.247.65: число байт=32 время=194мс TTL=48
Ответ от 205.188.247.65: число байт=32 время=240мс TTL=48
Ответ от 205.188.247.65: число байт=32 время=173мс TTL=48
Ответ от 205.188.247.65: число байт=32 время=250мс TTL=48
Ответ от 205.188.247.65: число байт=32 время=187мс TTL=48
Ответ от 205.188.247.65: число байт=32 время=239мс TTL=48
Статистика Ping для 205.188.247.65:
Пакетов: послано = 10, получено = 10, потеряно = 0 (0% потерь)
Приблизительное время передачи и приема:
Наименьшее = 173мс, наибольшее = 406мс, среднее =236мс

Изучение маршрута между сетевыми соединениями выполняется с помощью утилиты tracert. Tracert - это утилита трассировки маршрута. Она использует поле TTL (time-to-live, время жизни) пакета IP и сообщения об ошибках ICMP для определения маршрута от одного хоста до другого.

Утилита tracert может быть более содержательной и удобной, чем ping, особенно в тех случаях, когда удаленный хост недостижим. С помощью нее можно определить район проблем со связью (у Internet-провайдера, в опорной сети, в сети удаленного хоста) по тому, насколько далеко будет отслежен маршрут. Если возникли проблемы, то утилита выводит на экран звездочки (*), либо сообщения типа «Destination net unreachable», «Destination host unreachable», «Request time out», «Time Exeeded».

Утилита tracert работает следующим образом: посылается по 3 пробных эхо-пакета на каждый хост, через который проходит маршрут до удаленного хоста. На экран при этом выводится время ожидания ответа на каждый пакет (Его можно изменить с помощью параметра -w). Пакеты посылаются с различными величинами времени жизни. Каждый маршрутизатор, встречающийся по пути, перед перенаправлением пакета уменьшает величину TTL на единицу. Таким образом, время жизни является счетчиком точек промежуточной доставки (хопов). Когда время жизни пакета достигнет нуля, предполагается, что маршрутизатор пошлет в компьютер-источник сообщение ICMP “Time Exeeded” (Время истекло). Маршрут определяется путем посылки первого эхо-пакета с TTL=1. Затем TTL увеличивается на 1 в каждом последующем пакете до тех пор, пока пакет не достигнет удаленного хоста,

либо будет достигнута максимально возможная величина TTL (по умолчанию 30, задается с помощью параметра -h).

Маршрут определяется путем изучения сообщений ICMP, которые присылаются обратно промежуточными маршрутизаторами.

Примечание: некоторые маршрутизаторы уничтожают пакеты с истекшим TTL и не будут видны утилите tracert.

Синтаксис:

tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout] имя_целевого_хоста

Параметры:

-d - указывает, что не нужно распознавать адреса для имен хостов;

-h maximum_hops - указывает максимальное число хопов для того, чтобы искать цель;

-j host-list - указывает нежесткую статическую маршрутизацию в соответствии с host-list;

-w timeout - указывает, что нужно ожидать ответ на каждый эхо-пакет заданное число мсек.

Основная задача протокола ARP – трансляция IP-адресов в соответствующие локальные адреса. Для этого ARP-протокол использует информацию из ARP-таблицы (ARP-кэша). Если необходимая запись в таблице не найдена, то протокол ARP отправляет широковещательный запрос ко всем компьютерам локальной подсети, пытаясь найти владельца данного IP-адреса. В кэше могут содержаться два типа записей: статические и динамические. Статические записи вводятся вручную и хранятся в кэше постоянно. Динамические записи помещаются в кэш в результате выполнения широковещательных запросов. Для них существует понятие времени жизни. Если в течение определенного времени (по умолчанию 2 мин.) запись не была востребована, то она удаляется из кэша.

Синтаксис:

arp [-s inet_addr eth_addr] | [-d inet_addr] | [-a]

Параметры:

-s -занесение в кэш статических записей;

-d - удаление из кэша записи для определенного IP-адреса;

-a - просмотр содержимого кэша для всех сетевых адаптеров локального компьютера;

inet_addr - IP-адрес;

eth_addr - MAC-адрес.

Утилита netstat позволяет получить статическую информацию по некоторым из протоколов стека (TCP, UDP, IP, ICMP), а также выводит сведения о текущих сетевых соединениях. Особенно она полезна на брандмауэрах, с ее помощью можно обнаружить нарушения безопасности периметра сети.

Синтаксис:

netstat [-a] [-e] [-n] [-s] [-p protocol] [-r]

Параметры:

-a -выводит перечень всех сетевых соединений и прослушивающихся портов локального компьютера;

-e - выводит статистику для Ethernet-интерфейсов (например, количество полученных и отправленных байт);

-n выводит информацию по всем текущим соединениям (например, TCP) для всех сетевых интерфейсов локального компьютера. Для каждого соединения выводится информация об IP-адресах локального и удаленного интерфейсов вместе с номерами используемых портов;

-s выводит статистическую информацию для протоколов UDP, TCP, ICMP, IP. Ключ «/more» позволяет просмотреть информацию постранично;

-r выводит содержимое таблицы маршрутизации.

3.7.3. Задания для самостоятельного решения

Задание 1.

Получить справочную информацию по командам ipconfig, ping, tracert, hostname.

Задание 2.

Получить имя хоста.

Задание 3.

Изучить утилиты ipconfig.

Задание 4.

Протестировать связь с помощью утилиты ping.

Задание 5.

Определить путь IP-пакета.

Задание 6.

Просмотреть ARP-кэш.

Задание 7.

Получить информацию о текущих сетевых соединениях и протоколах стека TCP/IP.

Контрольные вопросы

1) Какие утилиты можно использовать для проверки правильности конфигурирования TCP/IP?
2) Каким образом каманда ping проверяет соединение с удаленным хостом?
3) Что такое хост?
4) Сколько промежуточных маршрутизаторов сможет пройти IP-пакет, если его время жизни равно 30?
5) Как работает утилита tracert?
6) Каково назначение протокола ARP?

3.8. Занятия 9-10. Адресация в IP-сетях. Преобразование адресов (4 часа)

3.8.1. Цель занятий

- ознакомление принципами адресации сетей;

- изучение CIDR-нотации;
- получение навыков преобразования адресов.

3.8.2. Методические указания по теме

Структура адресов IPv4 и IPv6 подробно изложены в [2]. В данном пособии рассмотрим использование масок подсетей.

Поля номеров сети и подсети образуют расширенный сетевой префикс. Для выделения расширенного сетевого префикса используется маска подсети (subnet mask). Маска подсети – это 32-разрядное двоичное число (по длине IP-адреса), в разрядах расширенного префикса содержащая единицу; в остальных разрядах находится ноль. Расширенный сетевой префикс получается побитным сложением по модулю два (операция XOR) IP-адреса и маски подсети.

При таком построении, очевидно, что число подсетей представляет собой степень двойки - 2^n, где n - длина поля номера подсети. Таким образом, характеристики IP-адреса полностью задаются собственно IP-адресом и маской подсети.

Для упрощения записи применяют следующую нотацию (так называемая CIDR-нотация): IP-адрес/длина расширенного сетевого префикса. Например, адрес 192.168.0.1 с маской 255.255.255.0 будет в данной нотации выглядеть как 192.168.0.1/24 (очевидно, что 24 – это число единиц, содержащихся в маске подсети).

В следующей таблице показаны стандартные маски подсетей для классов адресов Интернет:

Класс адреса	Биты маски подсети	Маска подсети
Класс A	11111111 00000000 00000000 00000000	255.0.0.0
Класс B	11111111 11111111 00000000 00000000	255.255.0.0
Класс C	11111111 11111111 11111111 00000000	255.255.255.0

Но для каждого класса возможны и другие маски подсети. Рассмотрим пример для класса A:

- 255.0.0.0 - маска для сети класса A; длина расширенного сетевого префикса - 8;
- 255.255.0.0 - маска для сети класса A; длина расширенного сетевого префикса -1 6;
- 255.255.255.0 - маска для сети класса A; длина расширенного сетевого префикса - 24.

В 1987 году документом RFC 1009 был определен порядок использования в сети, разделённой на подсети, нескольких масок подсети. В этом случае расширенные сетевые префиксы имеют разную длину и маски подсетей называются масками подсетей переменной длины (Variable Length Subnet Mask). Таким образом, можно разбить сеть на подсети разного размера.

Маска подсети переменной длины позволяет более эффективно использовать выделенное организации адресное пространство протокола IP.

Главная трудность связана с тем, что ранее каждая сеть могла иметь только одну маску подсети, а это, в свою очередь, ограничивало возможности организации в выборе размера подсети.

Предположим, например, что администратор намеревается настроить выделенную организации сеть класса B 130.5.0.0 на использование расширенного сетевого префикса/22. Номер подсети задается с помощью шести бит.Сеть класс B с расширенным сетевым префиксом /22 позволяет организовать 64 подсети ($2^6 = 64$), каждая из которых поддерживает максимум до 1022 ($2^{10} - 2 = 1022$) индивидуальных адресов хостов. Такой вариант может устроить администратора, если организации требуется некоторое число подсетей с большим количеством хостов в них. Однако если организации нужны подсети с числом хостов не более 30, то при фиксированной маске подсети администратору придется эксплуатировать подсети, рассчитанные на большое количество хостов, но содержащие всего несколько пользователей. В результате невостребованными могут оказаться около 1000 возможных адресов хостов в подсетях. Как видно, ограничение на использование только одной маски подсети значительно снижает эффективность распределения адресного пространства.

Основное решение данной проблемы состоит в введении маски подсети переменной длины. Предположим, что администратор хочет использовать расширенный сетевой префикс /26. Адрес класса B с таким расширенным сетевым префиксом позволит иметь до 1024 подсетей ($2^{10} = 1024$), каждая из которых может поддерживать до 62 ($2^6-2= 62$) индивидуальных адресов хостов. Такой расширенный сетевой префикс идеально подходит к небольшим подсетям, с числом хостов порядка 60, в то время как префикс /22 лучше подходит большим подсетям, с тысячами хостов.

Как видно, применение разных расширенных сетевых префиксов /22 и /26 позволяет получить два типа подсетей с резко отличающимся количеством поддерживаемых хостов. Введение маски подсети переменной длины дает возможность администратору создавать в рамках своей организации подсети требуемого размера. Это происходит следующим образом. Сначала сеть делится на подсети, затем некоторые из них делятся, в свою очередь, еще на подсети и т. д. - происходит своего рода рекурсия подсетей.

Таким образом, рекурсивное разбиение адресного пространства организации может быть выполнено с учетом пожеланий администратора сети. Кроме рекурсии адресов подсетей введение маски подсети переменной длины позволяет значительно уменьшить объем таблиц маршрутизации на маршрутизаторах в организации.

Предположим, что сеть организации охватывает несколько удаленных филиалов. Если организация имеет три удаленные сети, то ей понадобится выделить 3 бита для формирования подсетей - этого ей хватит как сегодня, так и в обозримом будущем ($2^3 = 8$). Второй уровень в иерархии подсетей образуют отдельные подсети внутри каждого филиала. Кроме того, каждой рабочей

группе также требуется выделить отдельные подсети. Следуя приведенной иерархической модели, верхний уровень определяется числом удаленных филиалов, второй - числом зданий внутри каждого филиала, а третий - максимальным числом подсетей в каждом здании и максимальным числом хостов в каждой из подсетей.

В следующих двух таблицах показано разбиения класса B на 5 подсетей, а также маски подсетей и broadcast. Рассмотрим адрес класса B 172.16.0.0/16 с маской подсети 255.255.0.0. Сначала разобьём его на 4 подсети с маской подсети /18.

172.16.0.0/16

IP-адрес подсети/префикс	Broadcast в десятичном представлении	Broadcast в двоичном представлении
172.16.0.0/18	172.16.63.255	10101100.00010000.00111111.11111111
172.16.64.0/18	172.16.127.255	10101100.00010000.01111111.11111111
172.16.128.0/18	172.16.191.255	10101100.00010000.10111111.11111111
172.16.192.0/18	172.16.255.255	10101100.00010000.11111111.11111111
Маска подсети	255.255.192.0 11111111.11111111.11000000.00000000	

Теперь разобьём, например, третью подсеть на 2 подсети, т.е. выделим ещё 1 бит (2^1) в расширенный префикс сети. Т.о. мы получим 2 подсети с маской /19.

172.16.128.0/18

IP-адрес подсети/префикс	Broadcast в десятичном представлении	Broadcast в двоичном представлении
172.16.128.0/19	172.16.31.255	10101100.00010000.00011111.11111111
172.16.160.0/19	172.16.63.255	10101100.00010000.00111111.11111111
Маска подсети	255.255.224.0 11111111.11111111.11100000.00000000	

Современные протоколы маршрутизации, такие как OSPF и IS-IS, позволяют использовать маску подсети переменной длины. Это достигается за счет передачи маски подсети в каждом сообщении об обновлении маршрутов, так что каждую подсеть можно рекламировать с соответствующей маской. Если протокол маршрутизации не рассчитан на это, то маршрутизатор будет либо предполагать, что ему следует использовать маску подсети своего локального порта, либо произведет поиск в статически настроенной таблице, содержащей всю информацию о масках подсетей. Первое решение не может гарантировать выбора корректной маски подсети, а статическая таблица плохо масштабируется, кроме того, она сложна в управлении и выполнении коррекции ошибок.

Таким образом, если требуется использование маски подсети переменной длины в сложной сетевой топологии, наилучшим выбором является

применение протоколов маршрутизации OSPF, IS-IS, а не RIP-1 IP. Однако при этом нужно учитывать, что вторая версия протокола RIP (RIP-2 IP), описанная в документе RFC 1388, расширяет возможности первой версии протокола, в том числе за счет возможности переноса маски подсети.

3.8.3. Задания для самостоятельного решения

Задание 1.

Определить к какому классу принадлежат следующие IP-адреса, а также рассчитать максимально возможное количество сетей и хостов для данного класса:

131.107.2.89

3.3.57.0

Задание 2.

Определить к какому классу принадлежат следующие IP-адреса, а также рассчитать максимально возможное количество сетей и хостов для данного класса:

200.200.5.2

191.107.2.10

Задание 3.

Определить к какому классу принадлежат следующие IP-адреса, а также рассчитать максимально возможное количество сетей и хостов для данного класса:

162.213.7.67

82.90.162.0

Задание 4.

Определить к какому классу принадлежат следующие IP-адреса, а также рассчитать максимально возможное количество сетей и хостов для данного класса:

208.55.90.0

171.12.106.31

Задание 5.

Определить к какому классу принадлежат следующие IP-адреса, а также рассчитать максимально возможное количество сетей и хостов для данного класса:

121.15.179.3

176.12.16.131

Задание 6.

Определить адрес выделенной подсети, если задан адрес хоста и расширенный сетевой префикс.

192.168.240.147/26

Задание 7.

Определить адрес выделенной подсети, если задан адрес хоста и расширенный сетевой префикс.

131.29.213.87/18

Задание 8.

Определить адрес выделенной подсети, если задан адрес хоста и расширенный сетевой префикс.

16.194.0.9/12

Задание 9.

Определить адрес выделенной подсети, если задан адрес хоста и расширенный сетевой префикс.

169.0.249.147/22

Задание 10.

Определить адрес выделенной подсети, если задан адрес хоста и расширенный сетевой префикс.

192.168.240.147/26

Контрольные вопросы

1) Что понимается под термином «маска подсети», каково ее назначение?
2) Для чего используется CIDR-нотация?
3) Перечислите основные типы адресов IPv4 и IPv6.
4) Назовите основные характеристики протоколов IPv4 и IPv6.
5) Перечислите протоколы маршрутизации, позволяющие использовать маску подсети переменной длины.

3.9. Занятие 11. Работа с удаленным компьютером на FTP-серверах, использование сервиса telnet для доступа к удаленному компьютеру (2 часа)

3.9.1. Цель занятий

- получение практических навыков и умений работы на FTP-серверах,
- использования сервиса telnet для доступа к удаленному компьютеру.

3.9.2. Методические указания по теме

Модель и принципы работы протоколов FTP и telnet подробно изложены в [2,4]. Для получения файлов, хранящихся на различных компьютерах Internet, пользователи сети прежде всего используют программы, основанные на протоколе FTP.

FTP - File Transfer Protocol - протокол передачи файлов, определяющий правила передачи файлов в сети с одного компьютера на другой.

Для работы с FTP на удаленном компьютере необходимо ввести пароль и (или) назвать себя (свое сетевое имя). В качестве имени пользователя при работе с FTP-серверами используется слово anonymous или ftp, а в качестве пароля берется адрес электронной почты.

Доступ должен быть как минимум типа dial-up (по вызову). Для запуска FTP-клиента, нужно подать команду ftp с указанием имени рабочей машины, с которой вы хотите провести сеанс.

Синтаксис команды ftp следующий:

ftp [-name] [hostname]

Параметры для пересылаемых файлов:

-d [level] - Переход в режим отладки.

-f <filename> - Выполнять только команды файла filename.

-g - Блокировка автоматического расширения имени файла.

-h <filename> - Указать файл конфигурации.

-i - Блокировка приглашений для групповых переносов файлов.

-m - Включить программу more.

-n - Блокировка режима автоматической регистрации.

-p <filename> - Выполнить команды, содержащиеся в файле filename. Эти команды выполнятся сразу после того, как вы зарегистрируетесь.

-r - Отключить переадресацию вывода.

-s - Отключить переключение слэша.

-v - Отображать любые сообщения удаленного компьютера.

Команды для передачи файлов:

account [password] - Получить пароль доступа к дополнительным ресурсам сервера FTP.

ascii ASCII - режим передачи данных.

bell - переслать данные и издать писк.

bget - бинарный режим передачи данных. Аналог команды get.

binary - бинарный режим передачи данных.

bput - переслать файл в бинарном режиме. Аналог команды put.

bye или quit - закончить выполнение ftp.

close - закрыть соединение с сервером FTP и выйти в DOS.

delete -удалить файл на удаленном компьютере.

debug [mode] - активизировать режим отладки, то есть ставить перед каждой командой, посланной на удаленный компьютер символы ->.

dir [other_directory][my_file] - распечатать содержимое локального каталога на удаленном компьютере. Если команда без аргумента other_directory, то на удаленном компьютере будет выведен текущий каталог. Если отсутствует аргумент my_file, то вся информация отобразится на экране локального монитора.

get other_file [my_file] - получить файл с удаленного компьютера и сохранить его на локальном компьютере. При отсутствии аргумента имя сохраняемого файла на локальном компьютере такое же, каким оно было на удаленном.

glob - посредством этой команды можно оперировать расширениями файлов, то есть использовать команды mdelete, mget и mput не только вместе с именами файлов, но и с их расширениями. Разрешается применять стандартные символы (* и &) в расширениях передаваемых файлов.

hash - активизировать режим печати символов # при передаче блоков, размер каждого из которых равен 1024 байта.

help [command] - отобразить описание команды.

interactive - активизировать режим выдачи сообщений во время приема или передачи файла.

lcd [directory] - Перейти в другой каталог локального компьютера. Если у этой команды отсутствует аргумент, то вы перейдете в каталог по умолчанию.

lls [directory] - просмотреть каталог локального компьютера.

ls [other_directory][my_file] - отобразить содержимое каталога удаленного компьютера. При отсутствии аргумента other_directory отображается содержимое каталога по умолчанию, при отсутствии аргумента my_file отображается файл, в который будет помещена информация с удаленного компьютера. Если вместо последнего аргумента стоит дефис, то вся информация с удаленного компьютера будет выведена на экран локального монитора.

mdelete [other_files] - удалить файлы other_files удаленного компьютера.

mdir other_files my_file - распечатать локальные файлы other_files на удаленном компьютере.

mget other_files - найти на удаленном компьютере файлы other_files, расшифровать и активизировать команду get для переноса этих файлов в рабочий каталог локального компьютера.

mkdir name_directory - создать каталог на удаленном компьютере.

mls other_files my_files - отобразить содержимое файлов удаленного компьютера.

mode [name_mode] - активизация режима переноса файлов в определенное место. По умолчанию установлен режим stream.

more - включить режим more, то есть через паузу разбивать содержимое каталогов на части.

mput files - найти и расшифровать локальные файлы files и запустить команду put для переноса этих файлов в рабочий каталог удаленного компьютера.

noninteractive - не выдавать сообщения во время пересылки или приема файлов.

open host [port] - соединиться с сервером FTP.

prompt - показывать интерактивные сообщения.

put my_file [other_file] -поместить локальный файл my_file на удаленный компьютер. При отсутствии аргумента other_file используется исходный файл.

pwd - распечатать имя текущего каталога на удаленном компьютере.

quote arg1 arg2 ... - передать аргументы arg1 arg2 ... на сервер FTP и получить только код ответа.

recv other_file [my_file] - аналог команды get.

remotehelp [name_command] - получить список доступных команд удаленного сервера FTP.

rename old_name new_name - дать другое имя файлу old_name удаленного компьютера.

rm other_file - Аналог команды delete.

rmdir name_directory - стереть каталог name_directory на удаленном компьютере.

send my_file [other_file] - аналог команды put.

sendport - активизация режима команд port, что позволяет ускорить пересылку файлов. Если port не работает, то по протоколу передачи файлы поступят на порт данных по умолчанию.

slashflip - изменить режим смены слэша.

status - отобразить состояние программы ftp в данный момент времени.

struct [name_struct] - установить соответствие между структурой файла и указанным именем. В установке по умолчанию имя структуры есть file.

type [name_type] - установить тип ascii для текстов и тип binary или image для графики. При отсутствии аргумента - тип по умолчанию ascii.

user name_user [password][access] – сообщение серверу FTP, кто вы есть. Если аргумент отсутствует, то выподится запрос на ввод пароля. Если указан только аргумент access, то после того, как Вы зарегистрируетесь, можно будет воспользоваться командой доступа account.

verbose - активизация режима сообщений, при котором можно получать полную информацию с сервера FTP (активизирован по умолчанию).

Telnet - сетевой протокол для реализации текстового интерфейса по сети. Основные команды режима командной строки telnet приведены ниже:

open host [port]	Начать telnet-сессию с машиной host по порту port. Адрес машины можно задавать как в форме IP-адреса, так и в форме доменного адреса
close	Завершить telnet-сессию и вернуться в командный режим. Однако в некоторых системах, если telnet был вызван с аргументом, close приведет к завершению работы telnet
quit	Завершить работу telnet
z	"Заморозить" telnet-сессию и перейти в режим интерпретатора команд локальной системы. Из этого режима можно выйти по команде Exit
mode type	Если значение type line, то используется буферизованный обмен данными, если character — то обмен не буферизованный
send argument	Данная команда используется для ввода команд и сигналов протокола TELNET, которые указываются в качестве аргумента (send ao-прервать выдачу на принтер NVT).

3.9.3. Задания для самостоятельного решения

Задание 1.

Напишите последовательность команд для авторизации на FTP- сервере и просмотра содержимого удаленного компьютера.

Задание 2.

Выполните запуск telnet –сервера.

Задание 3.

Написать команду: Начать telnet-сессию с машиной (номер host или IP-адрес) по порту 1326**.**

Задание 4.

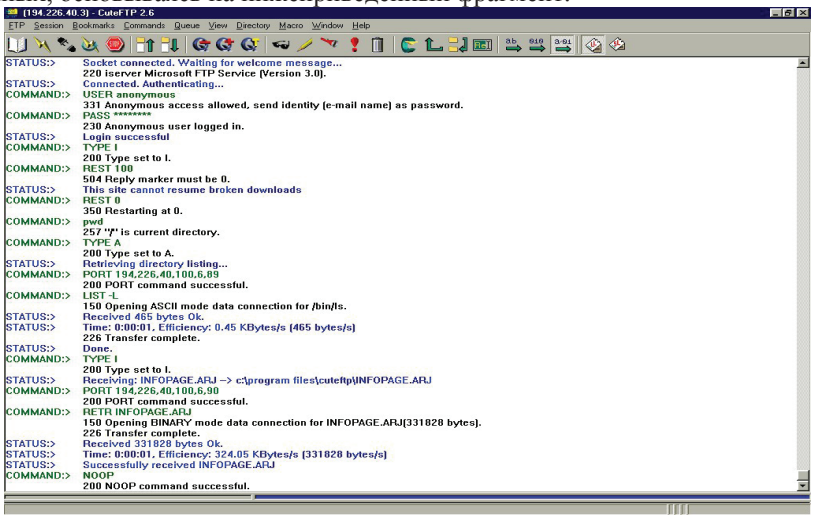
Написать команду: В режиме удаленного терминала работать с буферизацией (line-by line) и без нее (character-at-a-time).

Задание 5.

Выполните тестирование других протоколов: SMTP, HTTP и др.

Задание 6.

Укажите количество управляющих соединений и соединений передачи данных, основываясь на нижеприведенный фрагмент.



Контрольные вопросы

1) Назовите основные этапы работы в FTP.
2) Какие команды FTP используются для передачи файла?
3) Назовите опции FTP для операций по передаче файла.
4) Что означает тип файла в FTP? Что такое формат файла в FTP?
5) Как FTP использует протокол telnet на управляющем соединении?
6) Каково соотношение FTP и telnet?

Список рекомендуемой литературы

1. Э. Таненбаум. Компьютерные сети. 4-е изд.- М.:, 2007
2. Романчева Н.И.Компьютерные сети и интернет-технологии/Учебное пособие. - М.:МГТУ ГА.- 2014.-80с.
3. Уэнделл Одом. Компьютерные сети. Первый шаг-М.: Вильямс, 2005.-432 с.
4. Сидни Фейт TCP/IP: Архитектура, протоколы, реализация. – М.: ЛОРИ, 2000 – 756 с.