Оглавление

Приложение 1. Введение в Pacet Tracer

Packet Tracer — эмулятор сети передачи данных, выпускаемый фирмой Cisco Systems. Позволяет создавать работоспособные модели сетей, моделировать работу разнообразного сетевого оборудования, взаимодействовать между несколькими клиентами программы.

При первом запуске программы **Pacet Tracer** пользователь видит окно программы. Краткое описание интерфейса программы приведено далее (рисунок 1.):

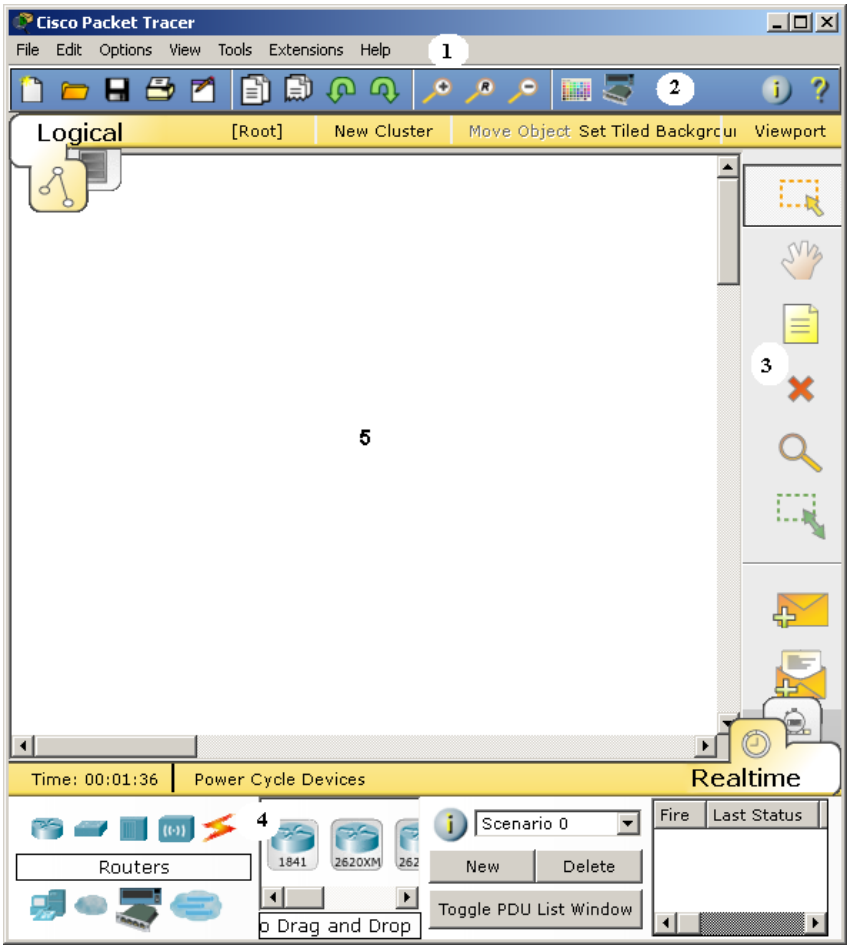

Рисунок 1

1. Стандартная строка опций;
2. Строка опций для работы с программой (сверху);
3. Строка действий над объектами, находящимися в рабочей области (справа);
4. Окно выбора оборудования;
5. Рабочая область.

Практическое задание 1

Создание простейшей локальной сети из коммутатора и двух компьютеров

1. Выбрать в меню оборудования компьютеры и перетащить два компьютера в рабочую область (рисунок 2).

Рисунок 2

2. Далее в меню оборудования войти в меню «Switches» (Коммутаторы) и выбрать устройство Generic Switch и перетащить его в рабочую область (рисунок 3).

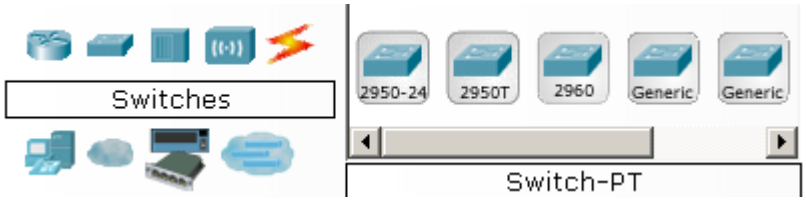

Рисунок 3

В меню коммутаторы, также как и в меню других устройств, пользователю предоставлена возможность выбрать существующую модель устройства или сгенерировать устройство самому, выбрав типы портов, требуемые для выполнения задачи. Для этого требуется открыть устройство двойным щелчком, войти в меню «Physical». В данном меню справа представлены типы портов, доступные для выбора пользователя. Для добавления требуется:

I. Выключить питание устройства, кнопкой «Power» на визуальной модели устройства.

II. Перетащить из меню «Modules» порты в устройство (рисунок 4).

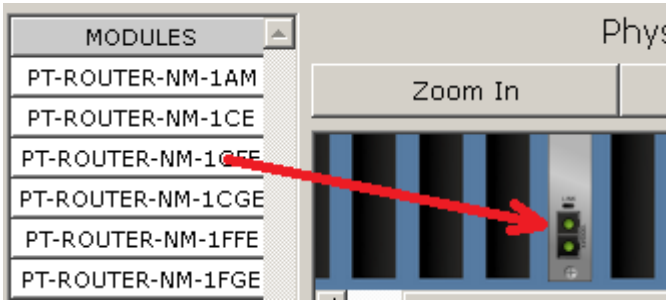

Рисунок 4

Для удаления порта из устройства, требуется перетащить его обратно в меню «Modules».

III. Включить устройство, с помощью той же кнопки «Power».

3. Далее стоит задача физически соединить устройства в сети.
Для этого в меню «Connections»(Соединения) выбрать Cooper Straight-Through(Витая пара) (рисунок 5), затем кликнуть на устройство (в нашем случае - компьютер или свитч) и выбрать порт, к которому нужно подключиться. Затем, выбрав порт на первом устройстве, выбрать порт на втором (рисунок 6). Таким образом, физическое соединение установлено.

Рисунок 5

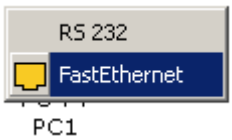

Рисунок 6

4. Далее следует задать сетевые параметры: для этого двойным щелчком открыть компьютер в рабочей области, в нем открыть вкладку «Desktop», и далее в открывшемся меню открыть вкладку «IP Configuration», ввести сетевые параметры и закрыть окно. Данную операцию проделать со всеми компьютерами сети.
5. Далее в меню любого из компьютеров открыть вкладку «Desktop» и открыть окно Command Prompt. Это командная строка. Затем с помощью команды ping проверить соединение между компьютерами сети. Имеется возможность проверять связь между

компьютерами, как в режиме реального времени, так и в режиме отслеживания пакета. Для этого в правой нижней части окна сменить Real time на Simulation (рисунок 7).

Рисунок 7

Практическое задание 2

Настройка маршрутизации

1. Создать две сети аналогично практической работе 1. (адрес первой сети – 192.169.56.0, адрес второй сети – 192.168.55.0).
2. Добавить в рабочую область два маршрутизатора. Для этого в меню выбора оборудования в меню «Routers» выбрать «Generic».
3. Далее требуется соединить устройства нашей сети физически. Для этого в меню «Connections» (Соединения) выбрать Cooper Straight-Through (Витая пара) и соединить кабелем пары маршрутизатор-коммутатор. Для соединения маршрутизаторов используется перекрестное соединение: в меню «Connections» (Соединения) выбрать «Cooper Cross-Over» (Витая пара с перекрестным соединением) и соединить порты Fast-Ethernet маршрутизаторов (рисунок 8).

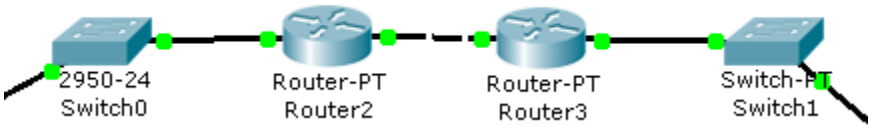

Рисунок 8

4. Теперь требуется настроить таблицы маршрутизации (рисунок 9).

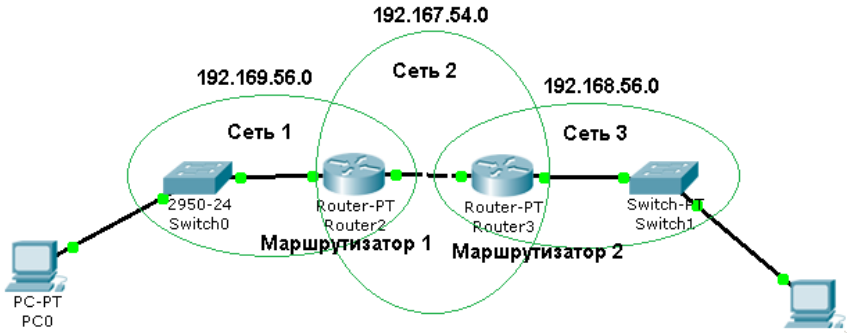

Рисунок 9

Зайти в меню маршрутизатора во вкладку «Static» (статическая маршрутизация). Затем требуется указать, в какую сеть пересылать следующий пакет. На каждом маршрутизаторе требуется указать пути для связи с сетями, в которых он не состоит. Например, из сети 1 требуется послать пакет в сеть 3. Для этого требуется указать на маршрутизаторе 1, куда отправлять пакет, адресованный сети 3. Здесь отправлять требуется на маршрутизатор 2, к которому в свою очередь и подключены компьютеры третей сети. Заполнить поля меню (Рисунок 10): Network – сеть, куда нужно отправить пакет, Mask – маска подсети в сети между маршрутизаторами, Next hope – следующий маршрутизатор для связи с сетью, в которую требуется отправить пакет. Не следует забывать о том, что для того чтобы пакет вернулся обратно в сеть 1, маршрутизатор 2 должен знать о том, как добраться в сеть 1.

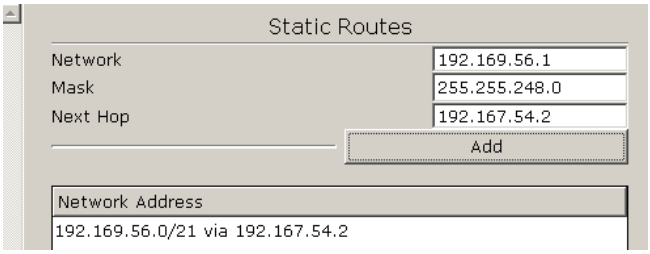

Рисунок 10

Приложение 2. Основы работы со средой виртуализации ORACLE VM VirtualBox

1. Основные концепции

ORACLE VM VirtualBox это виртуальная среда, относящаяся ко второму классу сред виртуализации - автономных эмуляторов компьютера, то есть для гостевой операционной системы эмулируется все оборудование, что позволяет запускать гостевую ОС без модификации ядра. Эмулируемые (виртуальные) жесткие диски физически хранятся в виде файлов с расширением .vdi и могут быть перенесены между реальными компьютерами. Состояние виртуальной машины может быть сохранено в виде «снимка». Позднее можно вернуться к сохраненному состоянию. Для переключения управления между виртуальной и базовой машинами используется специальная «хост-клавиша» (по умолчанию — правый <Ctrl>).

2. Основные элементы управления и меню.

Основное окно программы, служащее для создания, управления и удаления виртуальными машинами, представлено на рис. 1.

В верхней части расположены элементы управления, с помощью которых осуществляется процесс управления состоянием виртуальной машины.

В меню «Файл» доступны пункты манипуляций с конфигурациями виртуальных машин, управление виртуальными носителями и основными настройками программы (язык интерфейса, путь к папке, в которой будут храниться виртуальные машины, менеджер виртуальных носителей и т.п.).

Меню «Машина» (рис. 2) служит для управления существующими виртуальными машинами, их удаления и изменения конфигурации, а также для создания новых. Самые необходимые элементы данного меню вынесены в главное окно программы для увеличения удобства работы.

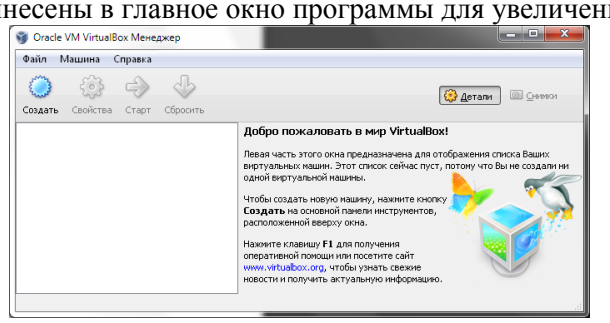

Рисунок 1

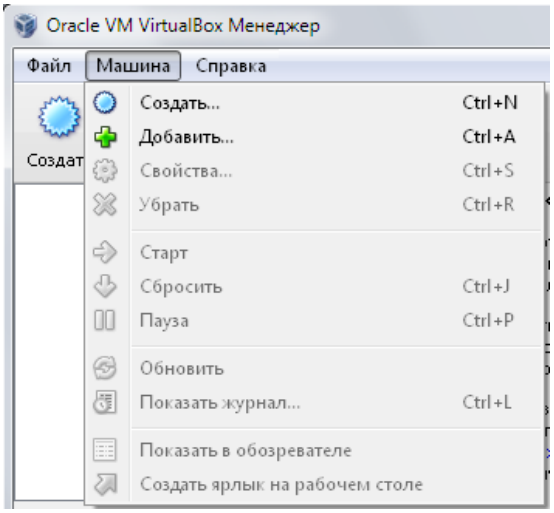

Рисунок 2

Меню «Справка» предоставляет стандартные функции по получению сведений о версии ORACLE VM VirtualBox, проверке актуальности текущей версии установленной программы, ссылку на официальный сайт и руководство пользователя на английском языке.

Важными элементами являются кнопки переключения режимов отображения параметров созданных виртуальных машин (рис. 3). Если активна кнопка «Детали», то в правой части основного окна приложения будет отображаться всю информация о виртуальной машине. При переключении в режим «Снимки», в правой части будут отображаться все созданные снимки выбранной виртуальной машины, появятся дополнительные элементы управления, необходимые для создания, удаления и использования имеющихся снимков.

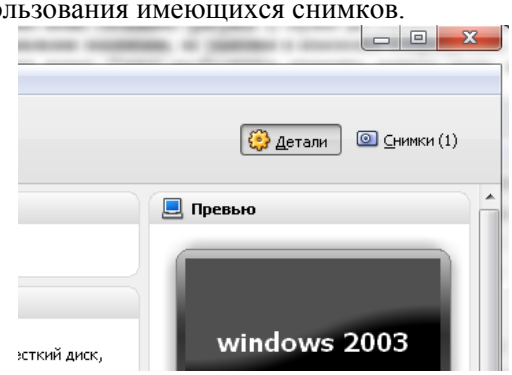

Рисунок. 3

3. Создание новой виртуальной машины.

Для создания новой виртуальной машины необходимо воспользоваться «Мастером создания новой виртуальной машины», который доступен под кнопкой «Создать» главного окна или из меню «Машина».

Для создания новой виртуальной машины необходимо последовательно указать следующие параметры:

- Имя машины и тип Операционной Системы
- Объём оперативной памяти для создаваемой машины
- Определить тип виртуального жёсткого диска, его местоположение и размер (если необходимо использовать существующий виртуальный диск, то его файл нужно подключить в менеджере виртуальных носителей).

4. Настройка виртуальной машины.

Параметры всех виртуальных машин можно изменять в любой момент, но они должны быть выключены. Окно изменений доступно с помощью кнопки «Свойства» в главном меню программы или в пункте меню «Машина». Доступ к окну изменений выбранного параметра возможен через нажатие на заголовок соответствующего пункта параметров в правой части экрана.

Пункт «Общие» позволяет изменить название машины, тип операционной системы, путь к папке для хранения снимков системы, параметры буфера обмена и её описание.

Пункт «Система» служит для изменения параметров связанных с оперативной памятью и процессором, а также позволяется задать порядок загрузочных устройств.

Пункт «Дисплей» определяет количество видео памяти и возможность подключения к данной виртуальной машине через протокол RDP.

Пункт «Носители» даёт возможность управлять всеми устройствами хранения данных с интерфейсами IDE и SATA.

Пункт «Аудио» предоставляет выбор аудио-драйвера и аудио-контроллера.

Пункт «Сеть» обеспечивает весь основной функционал в рамках сетевого взаимодействия с другими компьютерами. Он служит для активизации сетевых адаптеров и их настройки. Настройка включает в себя следующие параметры:

- Тип подключения
- Название используемого сетевого адаптера
- Тип сетевого адаптера
- MAC-адрес сетевого адаптера
- Управление портами

При желании подключить какое-либо USB-устройство следует воспользоваться средствами пункта «USB», но для корректной работы необходимо установить соответствующий программный компонент.

Пункт «Общие папки» позволяет подключить сетевые папки с реальных машин на виртуальные, что может служить связью между ними. В настройках VirtualBox указывается существующая папка и ее псевдоним для виртуальной машины. Внутри виртуальной машины доступ к общей папке осуществляется через «Сетевое окружение» в ОС Windows и через пункт «Сеть» в ОС Linux..

Приложение 3. Эталонная модель OSI

Для описания способов коммуникации между сетевыми устройствами организацией ISO в 1978 г. была разработана эталонная модель взаимосвязи открытых систем ЭМВОС — OSIBRM (Open Systems Interconnection Basic Reference Model). Она основана на уровневых протоколах, что позволяет обеспечить логическую декомпозицию сложной сети на обозримые части — уровни; стандартные интерфейсы между сетевыми функциями; симметрию в отношении функций, реализуемых в каждом узле сети (аналогичность функций одного уровня в каждом узле сети). Функции любого узла сети разбиваются на уровни, для конечных систем их семь.

Любой протокол модели OSI должен взаимодействовать либо с протоколами своего уровня, либо с протоколом выше или ниже своего уровня. Взаимодействия с протоколами своего уровня называются горизонтальными, а с уровнями выше или ниже — вертикальными. Любой протокол модели OSI может выполнять только функции своего уровня и не может выполнять функций другого уровня, что не выполняется в протоколах альтернативных моделей.

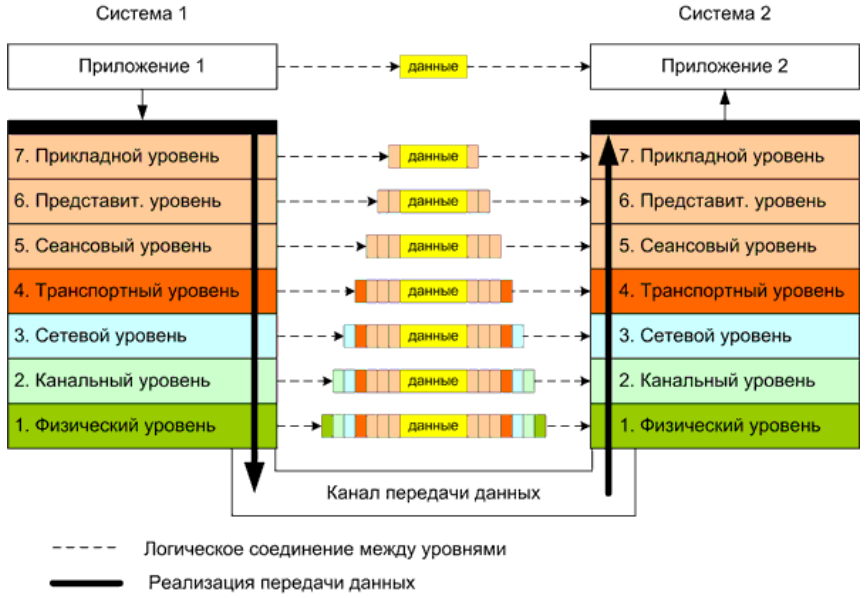

Рис. 1 Передача данных между двумя приложениями по стеку OSI

Внутри каждого узла взаимодействие между уровнями идет по вертикали. Взаимодействие между двумя узлами логически происходит по горизонтали — между соответствующими уровнями. Реально же из-за отсутствия непосредственных горизонтальных связей производится спуск до нижнего уровня в источнике, связь через физическую среду и подъем до соответствующего уровня в приемнике информации. Уровень, с которого посылается запрос, и симметричный ему уровень в отвечающей системе формируют свои блоки данных. Данные снабжаются служебной информацией (заголовком) данного уровня и спускаются на уровень ниже. На этом уровне к полученной информации также присоединяется служебная информация, и так происходит спуск до самого нижнего уровня, сопровождаемый увеличением количества заголовков. По нижнему уровню вся сформированная информация достигает получателя, где по мере подъема вверх освобождается от служебной информации соответствующих уровней. В итоге сообщение, посланное источником, достигает соответствующего уровня системы-получателя. Служебная информация управляет процессом передачи и служит для контроля его успешности и достоверности. В случае возникновения проблем может быть сделана попытка их уладить на том уровне, где они обнаружены. Если уровень не может решить проблему, он сообщает о ней на вызвавший его вышестоящий уровень.

Назначение уровней модели OSI и примеры протоколов, функции которых совпадают с функциями конкретных уровней модели OSI приведены в табл.1

Таблица 1

Прикладной уровень (application layer)
Основные функции: Передача служебной информации приложений, предоставляет

приложениям информацию об ошибках,
Примеры протоколов: FTP (File Transfer Protocol), Telnet (TErminaL NETwork), HTTP (HyperText Transfer Protocol), POP3 (Post Office Protocol Version 3), SMTP (Simple Mail Transfer Protocol).
Уровень представления данных (presentation layer)
Основные функции: Сжатие данных, шифрование данных, перекодировка данных
Примеры протоколов: SSL (Secure Socket Layer), RDP — Remote Desktop Protocol
Сеансовый уровень (session layer)
Основные функции: обеспечивает установление, поддержание и завершение сеанса связи, позволяя приложениям взаимодействовать между собой длительное время.
Примеры протоколов: L2TP (Layer 2 Tunneling Protocol), NetBIOS (Network Basic Input Output System), PAP (Password Authentication Protocol), PPTP (Point-to-Point Tunneling Protocol), RPC (Remote Procedure Call Protocol)
Транспортный уровень (transport layer)
Основные функции: Обеспечивает надежную доставку данных, подтверждение приема и сегментацию потока, получаемого от сеансового уровня.
Примеры протоколов: TCP (Transmission Control Protocol), UDP (User Datagramm Protocol)
Сетевой уровень (network layer)
Основные функции: Решает задачу доставки данных по составной сети, межсетевую адресацию, трансляцию физических адресов в сетевые.
Примеры протоколов: IP/IPv4/IPv6 (Internet Protocol), IPX (Internetwork Packet Exchange), IPsec (Internet Protocol Security), ICMP (Internet Control Message Protocol), RIP (Routing Information Protocol), OSPF (Open Shortest Path First), ARP (Address Resolution Protocol).
Канальный уровень (data link layer)
Основные функции: Обеспечивает формирование фреймов (frames) — кадров, передаваемых через физический уровень, контроль ошибок и управление потоком данных (data flow control). Логическое кодирование данных.
Примеры протоколов: ATM, Ethernet, EAPS (Ethernet Automatic Protection Switching), FDDI (Fiber Distributed Data Interface), MPLS (Multiprotocol Label Switching), PPP (Point-to-Point Protocol), SLIP (Serial Line Internet Protocol)
Физический уровень (physical layer)
Основные функции: обеспечивающий физическое кодирование бит кадра в электрические (оптические) сигналы и передачу их по линиям связи. Определяет тип кабелей и разъемов, назначение контактов и формат физических сигналов.
Примеры протоколов: IEEE 802.15 (Bluetooth), IRDA, EIA RS-232, EIA-422, Ethernet, DSL, ISDN, IEEE 802.11.

Приложение 4. Межсетевая передача между двумя узлами на примере взаимодействия сетевого и канального уровня.

Рассмотрим процесс передачи сообщения между двумя узлами по составной сети, ограничившись описанием взаимодействия сетевого и канального уровней. Под составной сетью будем понимать сеть, состоящую из локальных сетей, объединенных между собой маршрутизаторами, то есть через общий сетевой уровень.

Введем необходимые соглашения и условные обозначения.

1. Введем два вида адресов канального уровня (аналог MAC-адресов). Адрес первого типа будет формироваться из трех строчных букв латинского алфавита, адрес второго – из трех прописных букв. Наличием двух разных типов адресов мы указываем на то, что составная сеть может состоять из локальных сетей с разными канальными протоколами. Если адрес состоит из трех букв "z", то это будет широковещательный адрес канального уровня и кадр, отправленный на этот адрес принимают все узлы в локальной сети.
2. Общий для составной сети сетевой протокол будет иметь адреса (аналоги IP адресов), состоящие из двух цифр разделенных тире. Первая цифра указывает на адрес сети, вторая на адрес узла. Причем если в поле адреса узла стоит ноль, то это адрес сети целиком. При конфигурации узла будем указывать адрес шлюза в круглых скобках.
3. На рисунке1 приведем условные обозначения (a - узел сети, b - сеть, c – маршрутизатор, d - сетевое сообщение с адресом отправителя и адресом получателя, e - пример инкапсуляции сетевого сообщения (пакета) в сообщение канального уровня (в кадр).

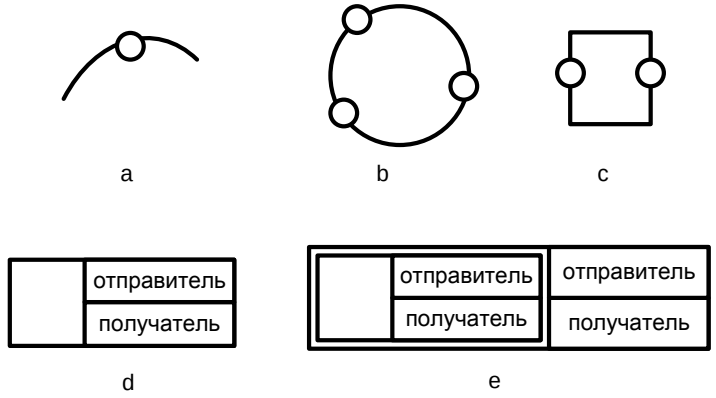

Рисунок 1

4. Примем упрощенные таблицы маршрутизации, в которых указывается адрес сети назначения, порт и шлюз. При передаче сообщения маршрутизатор по адресу назначения, содержащегося в заголовке пакета, определяет адрес сети назначения и по таблице маршрутизации определяет, через какой порт и на какой шлюз необходимо передавать его на следующем этапе маршрута.

На рисунке 2 показана составная сеть с адресной информацией.

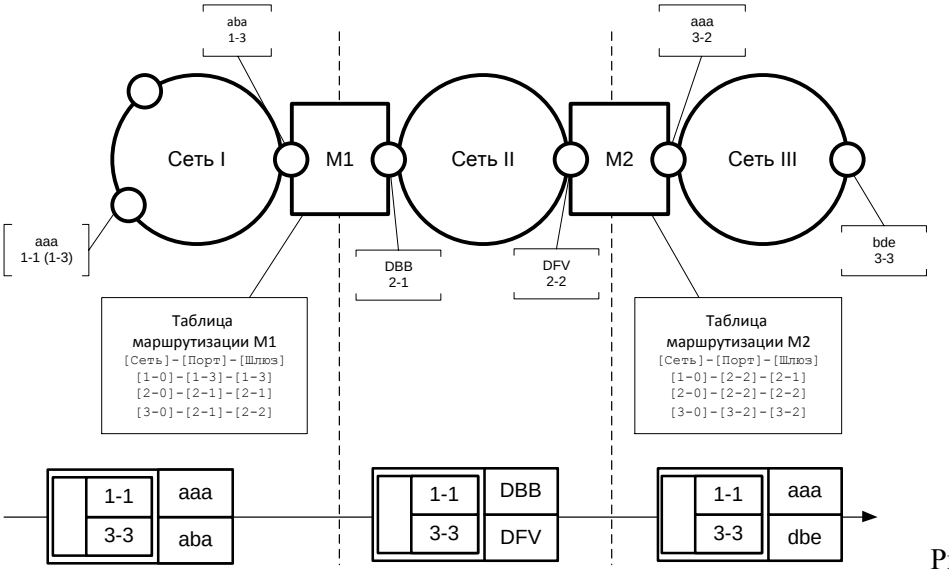

Рисунок 2

Опишем этапы передачи.

1. Перед началом передачи сетевой уровень передающей стороны сформирует пакет с адресом отправителя 1-1 и адресом получателя 3-3. Оставим за рамками рассмотрения откуда узел 1-1 "узнал" сетевой адрес получателя. Обычно такие задачи решаются с помощью систем, подобных DNS.
2. Перед инкапсуляцией сетевого пакета в кадр канального уровня сетевой уровень устанавливает, что адрес назначения лежит в другой локальной сети и передавать пакет надо через шлюз, указав его канальный адрес в поле адреса назначения кадра канального уровня.
3. В конфигурации узла адрес шлюза (1-3) дан в виде сетевого адреса, поэтому узел 1-1 генерирует широковещательное сообщение на канальном уровне адресованное на адрес "zzz" с запросом "у кого адрес 1-3?". Это сообщение получают все узлы сети 1, но отвечает на него только узел 1-3 со своего адреса канального уровня. Так узел 1-1 определяет канальный адрес назначения для первого шага.
4. Сетевой пакет инкапсулируется в кадр канального уровня, где в поле адреса отправителя стоит "aaa", а в поле получателя –канальный адрес шлюза "aba".
5. Этот кадр приходит на порт маршрутизатора-шлюза M1. Его канальный уровень принимает кадр для обработки, деинкапсулирует пакет сетевого уровня и передает его на свой сетевой уровень.
6. Сетевой уровень решает задачу маршрутизации. Сначала определяется адрес сети назначения по адресу назначения в сетевом пакете (адрес сети 3-0). По таблице маршуртизации по адресу сети назначения определяется порт, через который надо передать пакет и сетевой адрес следующего шлюза.
7. Сетевой пакет инкапсулируется в кадр канального уровня сети 2. При этом канальный адрес отправителя будет соответствовать аресу порта ("DBB"), а канальный адрес шлюза определяется по его сетевому адресу так же как и в п.3.
8. Сетевой пакет инкапсулированный в новый кадр канального уровня попадает на маршуртизатор M2. принимается им и обрабатывается так же как в п.5,6 и 7. С той разницей, что M2 определяет, что он непосредственно подключен к сети с адресом 3-0 (сеть 3) и определяет канальный адрес получателя не для следующего шлюза, а для узла назначения 3-3. Сетевой пакет инкапсулируется в новый кадр канального уровня в сети 3 и отправляется уже на узел 3-3. канальный уровень узла назначения принимает кадр, так как в адресе назначения стоит его адрес, деникапсулирует пакет сетевого уровня и передает его выше по стеку на сетевой уровень для обработки. Так как сетевой адрес назначения соответствует собственному адресу узла, то пакет принимается, деникапсулируется вложенное сообщение и передается выше по стеку.

Обратите внимание:

1. в сетях 1 и 3 есть узлы с одинаковыми адресами канального уровня. Это возможно, так как область действия адресации канального уровня – локальная сеть.
2. В составной сети адреса сетевого уровня из одной локальной сети должны иметь одинаковую сетевую часть. Это нужно для решения задачи маршрутизации.
3. В составной сети адреса сетевого уровня должны быть уникальными.
4. За счет процедуры инкапсуляции межсетевое взаимодействие не завит от природы канальных протоколов в локальных сетях.

Приложение 5. Коммутационное оборудование локальных сетей

В этом разделе рассмотрим некоторые виды коммутационного оборудования локальных сетей и особенности их работы.

Концентратор (анг. hub)

Назначение: объединение устройств в сеть.

Принцип работы: объединяет узлы на физическом уровне, усиливает сигнал, некоторые концентраторы могут согласовывать параметры сигнала. Поступающие сообщения концентратор копирует во все порты, предоставляя подключенным устройствам фильтровать трафик по назначению. Концентратор фактически предоставляет узлам общую среду передачи данных.

Особенности передачи трафика: никакого анализа трафика или его обработки не производится. Производит усиление сигнала.

Обработка широковещательных сообщений: рассылаются без ограничений.

Коммутаторы 2-го уровня (анг. L2 switch)

Назначение: объединение устройств в сеть.

Принцип работы: объединяет узлы на канальном уровне. Проходящие кадры фильтруются и продвигаются согласно адресной информации (MAC-адресам), содержащейся в их заголовках. Упрощенно принцип работы коммутатора 2-го уровня сводится к составлению и поддержанию в актуальном состоянии таблицы принадлежности адресов устройств к портам коммутатора и последующей фильтрации проходящего трафика согласно таблице.

Особенности передачи трафика: поступающий на порт коммутатора кадр записывается только в тот порт, к которому подключено устройство с адресом назначения. Остальные порты коммутатора свободны и могут участвовать в обмене данными между друг другом. В случае, если в таблице нет данных об адресе назначения, кадр записывается во все порты устройства. Адресная информация в заголовке кадра канального уровня не изменяется.

Обработка широковещательных сообщений: рассылаются без ограничений.

Маршрутизатор (анг. router)

Назначение: объединение устройств в сети, работа в качестве узловых точек сети, объединение сегментов сетей в составную сеть.

Принцип работы: объединяет устройства на сетевом уровне. Входящий кадр при поступлении на принимающий порт маршрутизатора подвергается деинкапсуляции на канальном уровне. Адресная информация, содержащаяся в заголовке сетевого пакета, используется для выбора маршрута передачи (порта маршрутизатора через который и шлюза, на который необходимо передать сетевой пакет). Решение принимается на основе записей таблицы маршрутизации, которые могут заносится в нее в ручную или с использованием специальных протоколов маршрутизации. Пакет инкапсулируется в новый кадр канального уровня.

Особенности передачи трафика: единицей передачи данных выступает сетевой пакет. Он передается в порт, определенный по таблице маршрутизации и подвергается инкапсуляции в кадр канального уровня. В качестве адреса назначения канального уровня выступает MAC адрес шлюза.

Обработка широковещательных сообщений: широковещательный трафик канального уровня не передается.

Коммутатор 3-го уровня (анг. L3 switch)

Назначение: объединение устройств в сети, работа в качестве узловых точек сети, объединение сегментов сетей в составную сеть.

Принцип работы: может работать в режиме коммутатора 2-го уровня. В режиме коммутатора 3-го уровня осуществляет коммутацию на основе таблиц коммутации, составленных относительно адресов сетевого уровня. Эти таблицы могут составляться автоматически, путем наблюдения трафика, вручную или с использованием протоколов маршрутизации. За счет аппаратной реализации большинства операций и отсутствие необходимости деинкапсуляции-инкапсуляции сетевых сообщений, в большинстве случаев работает быстрее маршрутизатора.

Особенности передачи трафика: кадр может передаваться без изменения адресной информации.

Обработка широковещательных сообщений: сообщения могут передаваться или фильтроваться в зависимости от настроек.

Приложение 6. Функции коммутаторов

Функции коммутаторов 2 уровня

Spanning Tree Protocol (приблизительный перевод - связующее дерево) – описывается стандартами IEEE 802.1d (STP), IEEE 802.1w Rapid Spanning Tree Protocol (RSTP), IEEE 802.1s Multiple Spanning Tree Protocol (MSTP). Технология позволяет использовать сложносвязанные топологии сетей основанных на коммутаторах. STP снимет ограничение на использование только древовидных топологий в таких сетях. Принцип работы заключается в выделении логического древовидного графа в сложносвязанном графе реальной сети. Технология применяется для повышения отказоустойчивости ЛВС или для реализации резервных каналов связи между несколькими ЛВС.

Автоопределение типа кабеля MDI/MDI-X – позволяет автоматически определить тип соединения в подключенном кабеле витая пара (прямой или кроссовый).

Автосогласование между режимами Full-duplex или Half-duplex – автоматическое определение возможного режима передачи данных по линии. В режиме Full-duplex данные передаются в двух направлениях одновременно по разным парам. При режиме Half-duplex данные могут передаваться только в одну сторону одновременно. Функция автосогласования между режимами позволяет избежать проблем с использованием разных режимов на разных устройствах.

Агрегация каналов (анг. Link aggregation for parallel links или pool) – описывается стандартом IEEE 802.3ad и предназначена для повышения пропускной способности канала за счет объединения нескольких портов в один высокоскоростной порт с суммарной скоростью объединенных портов. Максимальная скорость определенная стандартом составляет 8 Гбит/сек.

Виртуальные локальные сети (анг. VLAN) – описывается стандартом IEEE 802.1q и позволяет внутри одной физической локальной сети построить несколько отдельных логических сетей (виртуальных сетей), узлы которых изолированы от остальных участков сети.

Возможность установки в стойку (анг. rackmount) – возможность установки коммутатора в стойку или в коммутационный шкаф. Наибольшее распространение получили 19 дюймовые шкафы и стойки, которые стали для современного сетевого оборудования стандартом де-факто.

Возможность установки дополнительных модулей – эта возможность подразумевать наличие слотов расширения или портов подключения внешних модулей, позволяющие разместить дополнительные интерфейсы. В качестве дополнительных интерфейсов выступают гигабитные модули, использующие витую пару, и оптические интерфейсы, способные передавать данные по оптоволоконному кабелю.

Диагностика кабеля – технология, позволяющая контролировать состояние подключенных кабелей на основе медной витой пары или оптических линий. При помощи этой функции может быть определено местонахождение коротких замыканий, разрывов, несовпадений волнового сопротивления.

Зеркалирование портов (анг. Port Mirroring)- технология, позволяющая перенаправлять весь трафик с одного (One-to-One) или с нескольких (Many-to-One) портов на единственный порт коммутатора. Технология применяется для содержательного анализа сетевого трафика, проходящего через коммутатор.

Объединение в стек – технология, позволяющее объединять через специальные физические интерфейсы нескольких коммутаторов в одно логическое устройство. Стекирование целесообразно производить, когда в итоге требуется получить коммутатор с большим количеством портов (больше 48 портов). Различные производители коммутаторов используют свои фирменные технологии стекирования, к примеру, Cisco использует технологию стекирования StackWise (шина между коммутаторами 32 Гбит/сек) и StackWise Plus (шина между коммутаторами 64 Гбит/сек).

Приоритетизация трафика по тегам (анг. Priority tags) – описывается стандартом IEEE 802.1p и позволяет отсортировать кадры по степени важности, выставив приоритеты. Более приоритетные кадры будут отправляться в первую очередь, например, высокий приоритет выставляется пакетам VoIP и низкий — пакетам FTP.

Сбор статистики – одна из основных функций сетевого оборудования, дающая возможность анализировать трафик, тем самым выявлять уязвимые места инфраструктуры и в кратчайшие сроки ликвидировать их. Сбор статистики может осуществляться средствами самого сетевого оборудования или специально установленными серверами («примеры»).

Удаленное управление – возможность конфигурирования устройства через сетевое соединение, например средствами протокола SNMP (Simple Network Management Protocol), через встроенный в устройство Web-сервер или через консольный доступ, осуществляемый через ssh или telnet. Консольный доступ может осуществляться через локальные интерфейсы, такие как RS232 (COM-порт).

Управление потоком (анг. Flow Control) – описывается стандартов IEEE 802.3x и обеспечивает защиту от потерь пакетов при их передаче по сети. Принцип действия упрощенно заключается в согласовании работы взаимодействующих устройств, когда передающее и принимающее устройство согласуют интенсивность потока кадров в случае переполнения буфера приемника.

Управляемое питание по витой паре (Power over Ethernet/PSE) – описывается стандартом IEEE 802.af. Функция позволяет обеспечить питание (до 15,4 Ватт на порт) подключенных к коммутатору устройств таких, как IP-камеры, Wi-Fi точки доступа, IP-телефоны или многофункциональные терминалы.

Фильтрация многоадресных рассылок – технология, позволяющая фильтровать широковещательные рассылки канального уровня, которые обычно передаются без ограничений по всем портам коммутатора. Применяется для оптимизации трафика в крупных сетях.

Фильтрация трафика по MAC адресам – технология, позволяющая составлять ACL (списки контроля доступа) по отношения к адресам канального уровня. Используется для привязки подключенных устройств к порту коммутатора или для разрешения передачи трафика от определенных устройств на выбранный порт.

Функции коммутаторов 3-го уровня

L3 коммутация – упрощенно, возможность коммутатора проводить продвижение пакетов не на основе MAC адресов, а на основе IP адресов.

Поддержка протоколов маршрутизации – составление таблиц коммутации с помощью протоколов маршрутизации.

Фильтрация по параметрам IP и TCP\UDP – осуществление фильтрации трафика по алгоритмам формального межсетевого экарана, т.е. основываясь на значении IP адресов или портов TCP \ UDP.

Приложение 7. Протоколы стека TCP\IP

Стек TCP\IP состоит из четырех уровней. По реализуемым функциям уровни могут быть соотнесены с уровнями стека OSI. На рисунке 1 приведена структура стека TCP\IP с перечислением основных протоколов, относящихся к этим уровням.

Рисунок 1

Перечислим эти протоколы и дадим их краткую характеристику.

FTP (англ. File Transfer Protocol — протокол передачи файлов) – работает по протоколу TCP, порты 20 и 21. Предназначен для передачи файлов межу сервером и клиентом. Поддерживает авторизацию по имени пользователя и паролю. Не защищен.

SMTP (англ. Simple Mail Transfer Protocol — простой протокол передачи почты) – работает по 25 порту TCP, предназначен для передачи сообщений электронной почты между клиентским программным обеспечением и сервером, а также между серверами. Не содержит стандартных средств авторизации отправителя (кроме расширений ESMTP для авторизации клиента).

POP3 (англ. Post Office Protocol Version 3 - протокол почтового отделения, версия 3) – работает по 110 порту TCP. Предназначен для получения клиентом почтовых сообщений с сервера. Поддерживает авторизацию по имени пользователя и паролю. Не защищен.

IMAP4 (англ. Internet Message Access Protocol) — протокол прикладного уровня для доступа к электронной почте. Работает по 143 порту TCP. Предназначен для получения клиентом почтовых сообщений с сервера. Отличается возможностью хранения почтовых сообщений на сервере, их структурирование по каталогам и т.п.

HTTP (сокр. от англ. HyperText Transfer Protocol — протокол передачи гипертекста). Работает по портам 80, 8080 TCP. Предназначен для передачи текстовых и мультимедийных данных от сервера к клиенту по запросу последнего. В настоящее время используется как транспорт для других протоколов прикладного уровня.

RDP (англ. Remote Desktop Protocol — протокол удалённого рабочего стола). Работает по порту 3389 TCP. Протокол терминального доступа Microsoft. Существуют клиенты для различных операционных систем. Поддерживается отображение устройств клиентской стороны в терминальную сессию (принтеров, com-портов, аудиоустройств, смарткарт и дисковых устройств).

SSH (англ. Secure SHell — «безопасная оболочка») — сетевой протокол сеансового уровня

Telnet (англ. TErminaL NETwork — протокол терминального сетевого доступа). Работает по 21 порту TCP. Предназначен для организации полнодуплексного сетевого терминала между клиентом и сервером. Команды выполняются на стороне сервера. Поддерживает авторизацию по имени пользователя и паролю. Не защищен.

DNS (англ. Domain Name System — система доменных имён). Работает по портам 53 UDP для взаимодействия клиента и сервера и 53 TCP для AFXR запросов, поддерживающих обмен между серверами. DNS – протокол поддерживающий работу одноименной распределённой системы, осуществляющей отображение множества доменных имен и множества IP адресов хостов.

LDAP (англ. Lightweight Directory Access Protocol — облегчённый протокол доступа к каталогам). Работает по портам 389 TCP и UDP. Предназначен для чтения, добавления и изменения данных, хранящимся в службе каталогов. Используется в Active Directory от Microsoft, Open LDAP и др.

TCP (анг. Transmission Control Protocol - протокол управления передачей). Протокол транспортного уровня, обеспечивающий установку двунаправленного соединения между процессами, идентифицирующимися по сокету (комбинации IP адреса и порта), передачу потока сегментов внутри соединения с подтверждением приема, управление и завершение соединения. Сообщение TCP содержит в заголовке адреса сегментов в направленном потоке и контрольную сумму при расчете которой используется поле данных и заголовок. Для оптимизации передачи и предотвращения перегрузок сети используется механизм переменного окна, позволяющий вести передачу без получения подтверждения приема каждого сообщения. В качестве адресной информации использует порт.

UDP (англ. User Datagram Protocol — протокол пользовательских дейтаграмм). Протокол транспортного уровня, обеспечивающий передачу сообщений между процессами, идентифицирующимися по сокету (комбинации IP адреса и порта). Сеанс не устанавливается, подтверждения приема не осуществляется. В качестве адресной информации использует порт.

XTP (анг. Xpress transport protocol – быстрый транспортный протокол). Проектировался как замена TCP. Реализует раздельное управление потоком и подтверждением приема. В качестве адресной информации использует порт.

ICMP (англ. Internet Control Message Protocol — протокол межсетевых управляющих сообщений). Является диагностическим протоколом стека TCP\IP. Предназначен для запроса и оповещении о состояниях связи по протоколу IP и TCP, UDP. При передаче инкапсулируется в IP.

Оповещение реализовано конечным количеством кодов запроса и кодов ответа. Пример ответов: код 3 — Порт недостижим, код 5 — Неверный маршрут от источника. Пример запросов: 8 — Эхо-запрос, 30 — Трассировка маршрута (RFC-1393).

ARP (англ. Address Resolution Protocol — протокол определения адреса). Используется для определения MAC адреса по известному IP адресу. Соотнесение реализуется путем широковещательных рассылок. Область действия ограничена локальной сетью.

RARP (англ. Reverse Address Resolution Protocol — Обратный протокол преобразования адресов). Решает задачу обратную ARP – определение MAC по известному IP.

IP (анг. Internet Protocol — межсетевой протокол). Предназначен для доставки сообщений по составной сети. Реализует доставку данных в пределах локальной сети как подмножество основной задачи. Не гарантирует доставку. Существует в двух версиях IPv4 и IPv6. В качестве адресной информации используется IP адреса, имеющие разный формат в разных версиях протокола.

DHCP (англ. Dynamic Host Configuration Protocol — протокол динамической конфигурации узла). Предназначен для автоматического конфигурирования сетевого узла. В качестве конфигурационных параметров могут быть переданы: IP, mask, gate, адреса DNS, адрес сервера загрузки, сервера времени и т.п. Идентифицирует клиентов по MAC адресу к которому привязывается назначенный IP.

BOOTP (англ. Bootstrap Protocol –протокол сетевой загрузки) — сетевой протокол, используемый для автоматического получения клиентом IP-адреса. Является аналогом DHCP, но предназначен для загрузки бездисковых рабочих станций.

ESP (анг. Encapsulating Security Payload - инкапсуляция защищенных данных). Подпротокол IPSec. Предназначен для шифрования поля данных IP пакета. Реализуется за счет добавление служебного заголовка в поле данных IP пакета.

AH (анг. Authentication Header - идентификационный заголовок). Подпротокол IPSec. Предназначен для шифрования инкапсулированного IP пакета в IP пакете внешней сети. Реализуется за счет добавление служебного заголовка в поле данных IP пакета. Применяется дополнительно с ESP.

RIP (англ. Routing Information Protocol – протокол маршрутизации IP). Предназначен для автоматического составления таблиц маршрутизации. Является протоколом дистанционно-векторного типа. Алгоритм заключается в рассылке таблиц маршрутизации по соседям. Использует метрику маршрута, равную количеству промежуточных маршрутизаторов до сети назначения. Максимальное значение метрики – 15. Существует в двух вариантах RIP1 и RIP2. Последний является актуальным. Является внутренним протоколом маршрутизации, т.е. ориентирован на работу внутри автономных систем.

OSPF (англ. Open Shortest Path First – открытие кратчайшего пути первым). Предназначен для автоматического составления таблиц маршрутизации. Основан на технологии отслеживания состояния канала. Использует для нахождения кратчайшего пути Алгоритм Дейкстры. Использует метрики, учитывающие пропускную способность канала. Является внутренним протоколом маршрутизации, т.е. ориентирован на работу внутри автономных систем.

BGP (англ. Border Gateway Protocol - протокол граничного шлюза). Работает через 179 порт TCP. Предназначен для автоматического составления таблиц маршрутизации. Является внешним протоколом маршуртизации. BGP поддерживает бесклассовую адресацию, при которой маршрутизаторы обмениваются уменьшенными таблицами маршрутизации полученными суммированием маршрутов.

EGP (англ. Exterior Gateway Protocol - протокол внешнего шлюза). Устаревший вариант BGP.

PPTP (англ. Point-to-Point Tunneling Protocol - туннельный протокол типа точка-точка). Предназначен для туннелирования трафика по логической топологии точка-точка. Позволяет устанавливать защищённое соединение между двумя узлами путем инкапсуляции кадры PPP в IP. PPTP использует дополнительное TCP-соединение для обслуживания туннеля.

L2TP (англ. Layer 2 Tunneling Protocol - протокол туннелирования второго уровня). Предназначен для организации туннеля в том числе и на втором уровне модели OSI. То есть он позволяет создавать туннель не только в сетях IP, но и в таких, как ATM, X.25 и Frame Relay. Реализуется за счет добавление служебного заголовка в поле данных внешнего кадра или IP Реализуется за счет добавление служебного заголовка в поле данных кадра или IP пакета в которые производится инкапсуляция.

Приложение 8. Заголовок IP-пакета.

<table>
<tr><td>Версия (4 бита)</td><td>IHL(4 бита)</td><td>Тип обслуживания(8 бит)</td><td colspan="2">Длина пакета(16 бит)</td></tr>
<tr><td colspan="3">Идентификатор(16 бит)</td><td>Флаги(3 бита)</td><td>Смещение фрагмента</td></tr>
<tr><td colspan="2">Время жизни(8 бит)</td><td>Протокол(8 бит)</td><td colspan="2">Контрольная сумма заголовка</td></tr>
<tr><td colspan="5">IP-адрес отправителя (32 бита)</td></tr>
<tr><td colspan="5">IP-адрес получателя (32 бита)</td></tr>
<tr><td colspan="5">Параметры (от 0 до 10-ти 32-х битных слов)</td></tr>
<tr><td colspan="5">Данные (до 65535 байт минус заголовок)</td></tr>
</table>

Заголовок IP

Версия(Version) - для ip-протокола версии 4 значение поля должно быть равно 4.
IHL - длина заголовка IP-пакета в 32-битных словах (dword), указывающая начало блока данных в пакете.
Тип обслу́живания (Type of Service) - байт, содержащий информацию о типе обслуживания IP-пакетов.
Длина пакета(Total Lenght) – поле указывающее общую длину пакета в байтах.
Идентификатор(ID) - значение, назначаемое отправителем пакета и предназначенное для определения корректной последовательности фрагментов при сборке датаграммы. Для фрагментированного пакета все фрагменты имеют одинаковый идентификатор.
Флаги(Flags) - первый бит всегда равен нулю, второй бит определяет возможность фрагментации пакета и третий бит показывает, не является ли этот пакет последним в цепочке пакетов.
Смещение фрагмента(Fragment Offser) - значение, определяющее позицию фрагмента в потоке данных.
Время жизни (Time To Live) – параметр определяющий время существования пакета в сети. Представляет собой численное поле в заголовке пакета, значение которого уменьшается при прохождении очередного маршрутизатора минимум на единицу, если передача данных через устройство заняла больше времени, то на величину этой задежки. Если значения этого поля равно нулю то, пакет должен быть отброшен.
Протокол(Protocol) - идентификатор интернет-протокола следующего уровня указывает, данные какого протокола содержит пакет, например, TCP или ICMP.
Контрольная сумма заголовка(Header Checksum) - контрольная сумма заголовка пакета. Пересчитывается каждый раз при смене заголовка - например, если он проходит через очередной маршрутизатор.
Адрес отправителя(Source Address) - IP-адрес источника, отославшего пакет.
Адрес получателя(Destination Address) - IP-адрес назначения, куда был послан пакет.
Поле опций (Options) – необязательное поле, задающее дополнительные параметры пакета.

Приложение 9. Заголовки TCP-сегмента и дейтаграммы UDP.

<table>
<tr><td colspan="3">Порт источника(16 бит)</td><td>Порт назначения(16 бит)</td></tr>
<tr><td colspan="4">Номер последовательности(32 бита)</td></tr>
<tr><td colspan="4">Номер подтверждения(32 бита)</td></tr>
<tr><td>Смещение данных (4 бита)</td><td>Зарезервировано (4 бита)</td><td>Флаги (4 бита)</td><td>Размер окна(16 бит)</td></tr>
<tr><td colspan="3">Контрольная сумма(16 бит)</td><td>Указатель важности(16 бит)</td></tr>
<tr><td colspan="4">Опции(32 бита)</td></tr>
<tr><td colspan="4">Данные</td></tr>
</table>

TCP заголовок

Порт источника - идентифицирует приложение клиента, с которого отправлены пакеты. Порт назначения - идентифицирует порт, на который отправлен пакет.
Номер последовательности - выполняет две задачи:
Если установлен флаг SYN, то это начальное значение номера последовательности — ISN (Initial Sequence Number). Первый байт данных, который будут передан в следующем пакете, будет иметь номер последовательности, равный ISN + 1.
В противном случае, если SYN не установлен, первый байт данных, передаваемый в данном пакете, имеет этот номер последовательности.
Поскольку поток TCP в общем случае может быть длиннее, чем число различных состояний этого поля, то все операции с номером последовательности должны выполняться по модулю 2^32. Это накладывает практическое ограничение на использование TCP. Если скорость передачи коммуникационной системы такова, чтобы в течение MSL (максимального времени жизни сегмента) произошло переполнение номера последовательности, то в сети может появиться два сегмента с одинаковым номером, относящихся к разным частям потока, и приёмник получит некорректные данные.
Номер подтверждения - если установлен флаг ACK, то это поле содержит номер последовательности, ожидаемый получателем в следующий раз.
Смещение данных - поле определяющее размер заголовка пакета TCP в 4-байтных словах. Минимальный размер составляет 5 слов, а максимальный — 15, что составляет 20 и 60 байт соответственно. Смещение считается от начала заголовка TCP.
Зарезервировано – шести битное поле, для будущего использования, должно устанавливаться в ноль. Из них два (5-й и 6-й) уже определены:
CWR (Congestion Window Reduced) — Поле «Окно перегрузки уменьшено» — флаг установлен отправителем, чтоб указать, что получен пакет с установленным флагом ECE (RFC 3168)
ECE (ECN-Echo) — Поле «Эхо ECN» — указывает, что данный узел способен на ECN (явное уведомление перегрузки) и для указания отправителю о перегрузках в сети (RFC 3168)
Флаги (управляющие биты) - поле содержит 6 битовых флагов:
URG(англ. Urgent pointer field is significant) - поле «Указатель важности».
ACK(англ. Acknowledgement field is significant) - поле «Номер подтверждения».
PSH(англ. Push function) - сообщает о данных, накопившиеся в приемном буфере, в приложениях пользователя.
RST(англ. Reset the connection) – обрывает соединения, сбрасывает буфер.
SYN(англ. Synchronize sequence numbers) - Синхронизация номеров последовательности
FIN(англ. FIN bit used for connection termination) - флаг, будучи установлен, указывает на завершение соединения.
Окно - в этом поле содержится число, определяющее в байтах размер данных, которые получатель готов принять.

<table>
<tr><td>Порт отправителя(16 бит)</td><td>Порт получателя(16 бит)</td></tr>
<tr><td>Длина датаграммы(16 бит)</td><td>Контрольная сумма(16 бит)</td></tr>
<tr><td colspan="2">Данные</td></tr>
</table>

UDP заголовок