МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

А.Д. Береснев, А. И. Говоров, А. В. Чунаев

ПРАКТИЧЕСКИЕ РАБОТЫ ПО КУРСУ ИНФОРМАЦИОННЫЕ СЕТИ

Учебное пособие

Санкт-Петербург
2012

А.Д. Береснев, А. И. Говоров, А. В. Чунаев, Практические работы по курсу информационные сети. – СПб: НИУ ИТМО, 2011. – 66 с.

Учебное пособие содержит описания девяти практических работ по курсу «Информационные сети» и девять приложений с краткими теоретическими сведениями. В описаниях работ сформулированы цели их выполнения, дан перечень необходимых программных средств и краткие теоретические сведения, определены порядок выполнения и содержание отчетов по работе. В результате выполнения практических работ, подготовки и защите отчетов, обучающиеся должны сформировать умение выбирать технические решения по организации информационных сетей начального уровня, проектировать IP-сети с учетом архитектуры составной сети, работать с основными протоколами стека TCP\IP.

Пособие адресовано студентам специальностей 230202 - "Информационные технологии в образовании" и 230100 - "Информатика и вычислительная техника".

Рекомендовано к печати ученым советом факультета, 21.12.2011, протокол заседания № 9.

В 2009 году Университет стал победителем многоэтапного конкурса, в результате которого определены 12 ведущих университетов России, которым присвоена категория «Национальный исследовательский университет». Министерством образования и науки Российской Федерации была утверждена программа его развития на 2009–2018 годы. В 2011 году Университет получил наименование «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики»

Оглавление:

Практическая работа 1.
Структурированные кабельные системы

Цель работы:

- Получить представление о видах структурированных кабельных систем (СКС) и оборудовании, применяемом для их монтажа;
- Получить практические навыки монтажа кабельных систем на основе сетевых карт **Ethernet** / **FastEthernet**;
- Изучить назначение прямого и кроссированного соединения (**T568A** и **T568B**).

Необходимо:

- 2 компьютера с сетевыми картами Ethernet / FastEthernet;
- кабель **UTP Cat 5**, коннекторы **RJ45**, инструмент для монтажа кабеля;
- Программный пакет **Microsoft Visio**.

Краткие теоретические сведения:

Кабельная система (КС) – это совокупность линий связи и пассивного соединительного оборудования, предназначенная для передачи одного или нескольких типов сигналов. КС стандартизируются соответствующими типами документов: IEEE, ISO, ГОСТ. Структурированные кабельные системы - особые КС, удовлетворяющие таким требованиям как модифицируемость, надежность, емкость. СКС делят на горизонтальные, вертикальные и сети кампуса (табл. 1).

Таблица 1

Вид КС	Назначение	Требования
Горизонтальная	Соединение устройств в пределах помещения/этажа	модифицируемость, надежность, универсальность
Вертикальная	Соединение ГКС в пределах здания	Емкость, надежность
Сеть кампуса	Соединение ВКС между зданиями	Емкость, надежность

К пассивному оборудованию горизонтальной кабельной системы относятся:

- линии связи (кабели);
- розетки;

- Patch-panel (фактически розетки с большим количеством портов);
- patch-cord (кабели с установленными на них вилками, соединяющие активное и пассивное оборудование);
- прочее оборудование (стойки и кроссы).

Основной тенденцией развития ГКС является рост универсальности систем. По одним и тем же каналам могут передаваться сигналы аналоговой и цифровой телефонии, компьютерные данные, сигнал сетей вещания, видеосигнал, сигналы сетей сигнализаций. Достигается эта возможность за счет применения промежуточного пассивного оборудования – кроссов и Patch-panel.

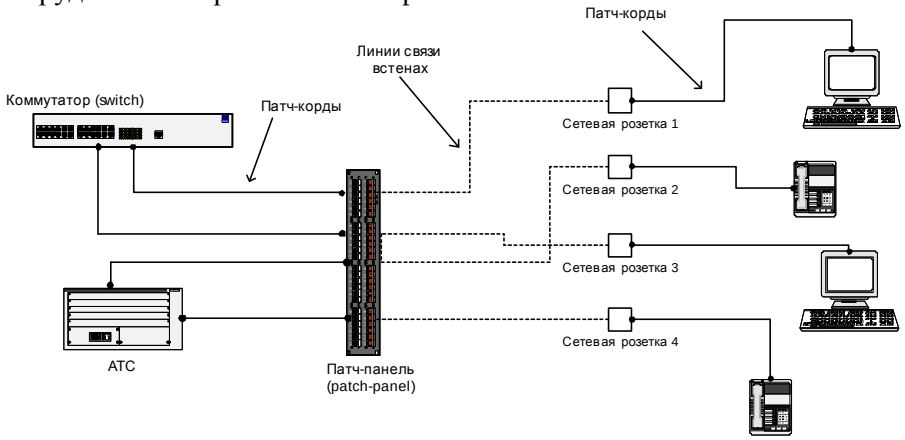
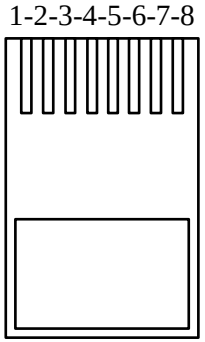


Рисунок 1

На рисунке 1 представлено использование одной кабельной системы для соединения разнородного активного оборудования. Так, заменяя только патч-корды, мы по одним и тем же линиям передаем по нашему желанию или голосовой аналоговый трафик, или компьютерные данные. Причем эта конфигурация может изменяться произвольно. При необходимости могут соединяться отдельные порты на патч-панели, для непосредственного соединения активного оборудования.

Патч-корд – кабель для соединения пассивного и активного оборудования. Различают патч-корды для компьютерных сетей (8 контактов) с вилкой RJ45, для телефонных сетей с вилками RJ16 (4 контакта) и RJ11 (2 контакта). Термин «RJ45» ошибочно применяется к восьмиконтактному разъему 8P8C. На самом деле настоящий RJ45 физически несовместим с 8P8C, так как использует схему 8P2C. Однако, общеупотребительным является термин «RJ45».

Для распределения контактов внутри коннектора существуют два стандарта T568A и T568B. Порядок проводов по цветам представлен на рисунке 2.



T568A: БЗ-З-БО-С-БС-О-БК-К
T568B: БО-О-БЗ-С-БС-З-БК-К

Рисунок 2

В соответствии с этими стандартами разводятся кабели на патч-панелях и розетках.

Для соединения двух разнородных устройств (компьютера и коммутатора) используется прямое соединение, то есть используется один стандарт на окончаниях всех соединениях.

Для соединения двух однородных устройств (компьютера и компьютера или коммутатора и коммутатора) используется перекрестное соединение, когда один раз по ходу линии используется соединение со сменой стандартов (A-B). Это может быть перекрестный патч-корд или перекрестное соединение розетки и патч-панели.

Порядок выполнения работы:

Часть 1.1. Соединение компьютеров на физическом уровне

1. Определить, какой стандарт соединения требуется для связи двух **однородных устройств**, например, компьютеров.
2. Удалить внешнюю оболочку кабеля на длину **12-13 мм** (1/2 дюйма). В обжимном инструменте имеется специальный **нож и ограничитель**.
3. Расплести кабель и расположить провода для **перекрёстного** соединения.
4. Повернуть вилку **металлическими контактами вверх** или пластмассовым «хвостиком» вниз и вставить в неё кабель. Проверить **правильность расположения** проводов и зубьев каждого контакта.

5. Используя обжимной инструмент, обжать вилку с кабелем.
6. С помощью кабельного тестера **проверить правильность** соединения коннекторов.

Часть 1.2. Соединение компьютеров на физическом уровне с помощью пач-панели

1. На **рисунке 3** представлена схема сети, которую необходимо собрать.
2. Составить **план сети**, определив и отметив на плане стандарты соединений.
3. Используя монтажный инструмент, собрать сеть.
4. Соединить два компьютера собранной сетью. Признаком наличия соединения будут горящие **индикаторы Link** на сетевых адаптерах.
5. В случае если сеть не работает, использовать кабельный тестер для **локализации неисправностей**.

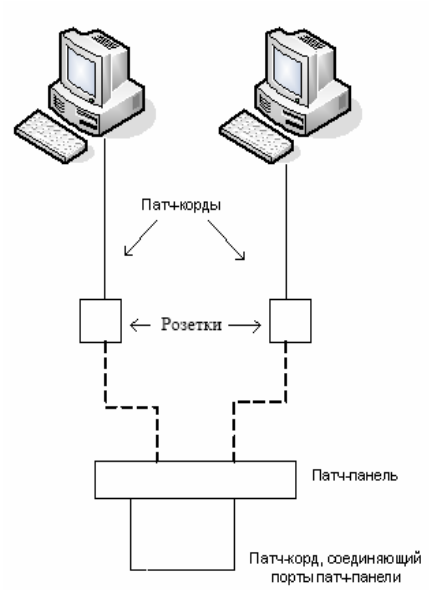


Рисунок 3

Часть 2. Разработка плана кабельной системы этажа (в соответствии с введенными стандартами)

Руководствуясь **файлом** «Пример выполненного задания» и положениями из **СНИП 2.09.04-87**, по данному плану помещения определить положение сетевых розеток (локальная сеть, телефония). Исходя из соответствующих **стандартов**, составить схему проводки кабелей, установки розеток, а также таблицу спецификаций материалов.

Содержание отчёта

В отчёте необходимо предоставить результат выполнения части 2, с использованием пакета **Microsoft Visio,** и ответы на контрольные вопросы. **Документ** Visio должен содержать страницы:

- Титульный лист;
- Пояснительная записка;
- Общие данные;
- Схема размещения розеток;
- Схема установки оборудования и монтажа розетки;
- Таблица соединений;
- Спецификация материалов.

Примечание: форма выполнения данных страниц приведена в файле «Пример выполненного задания».

Контрольные вопросы:

1. Зачем нужна смена стандартов при соединении однородных устройств?
2. Чем отличаются стандарты витой пары категорий 5, 5e, 6, 7?
3. Заполнить таблицу параметров кабельных сегментов в соответствии с их типом и назначением:

Тип кабеля	Названия стандартов, регламентирующих применение данных линий связи (ISO/IEC)	Основные области применения	Максимальная длина кабельного сегмента для сетей Ethernet (без использования повторителя)
Коаксиальный кабель			
Оптоволоконный кабель			
Витая пара категории 5			
Витая пара категории 5e			
Витая пара категории 6			
Витая пара категории 7			

Практическая работа 2.
Консольные утилиты настройки сетевых компонентов в MS Windows 2000/XP/2003 и Linux

Цель работы:

- Получить практические навыки по работе со средой виртуализации **ORACLE Virtual Box**;
- Получить практические навыки по конфигурированию сети в операционных системах **Windows** и **Linux**;
- Ознакомиться с утилитами командной строки, предназначенными для диагностики и настройки сети;
- Ознакомиться с форматом записи пути до сетевого ресурса **UNC**.
- Разработать **исполняемые файлы**, конфигурирующие сетевой интерфейс по заданным параметрам;

Необходимо:

- Установленная на компьютере среда виртуализации **ORACLE Virtual Box**
- Образы виртуальных жёстких дисков операционных систем **Windows 2003** и **Linux**

Краткие теоретические сведения:

Не смотря на то, что в состав современных операционных систем входят утилиты конфигурирования сети с графическим интерфейсом, задачи по диагностике и настройке сети удобнее решать с помощью консольных утилит.

В MS Windows к этим утилитам относят:

- **Ipconfig** – утилита отображения конфигурации IP,
- **Ping** – утилита диагностики сетевого соединения,
- **Net** – комплекс утилит для работы с сетью Microsoft,
- **Netsh** – утилита настройки всего стека протоколов MS Windows.

Справку по утилитам командной строки можно получить так:

command_name /? , а по команде net так: **net help имя_директивы**.

Linux – UNIX-подобная, многозадачная операционная система. Основным для нее является **текстовый интерфейс**, хотя для Linux разработаны (или портированы) графические оболочки, такие как **KDE** или **Gnome**.

В Linux запускаются **несколько** консолей, переключаться между которыми можно по кнопкам **Alt + Ctrl + F1** для перовой консоли, **Alt + Ctrl + F2** для второй и т.д.

Краткую справку по каждой команде можно получить с помощью команды **man**, краткую с помощью ключа **–h (--help)**. Например: **man ifconfig**. Также полезными для получения справки могут оказаться команды **apropos** и **whatis**.

В Linux, не смотря на то, что в разных дистрибутивах методы хранения конфигурационной информации разнятся, утилиты настройки сети идентичны:

- **ifconfig** – отображение настроек и конфигурирование сети,
- **route** – управление таблицей маршрутизации (и, соответственно, настройками шлюза по умолчанию).
- настройки **DNS** хранятся в текстовом файле **/etc/resolv.conf**

Сетевые интерфейсы в Linux именуются (для сетей Ethernet) **ethN**, где **N — номер** сетевого адаптера начиная **с нуля**.

Основными параметрами настройки сетевых интерфейсов являются:

- **IP-адрес**
- **Маска подсети**
- **Gateway (шлюз по умолчанию)**
- **DNS-сервер**

IP-адрес (*Internet Protocol Address*) — уникальный сетевой адрес узла в компьютерной сети, построенной по протоколу IP. Имеет длину 4 байта.

В терминологии сетей TCP/IP **маской подсети** или **маской сети** называется битовая маска, определяющая, какая **часть IP-адреса** узла сети относится **к адресу сети**, а какая **— к адресу самого узла** в этой сети.

Шлюз по умолчанию (*Default gateway*), шлюз последней надежды (*Last hope gateway*) — в маршрутизируемых протоколах — адрес маршрутизатора, на который отправляется трафик, для которого **невозможно определить маршрут** исходя из таблиц маршрутизации.

DNS (*Domain Name System*) — компьютерная распределённая система для получения **информации о доменах**. Чаще всего используется для **получения IP-адреса по имени хоста** (компьютера или устройства).

Все эти параметры можно настраивать вручную или при помощи специальной службы.

DHCP (*Dynamic Host Configuration Protocol*) — это **сетевой протокол**, позволяющий компьютерам **автоматически** получать IP-адрес и другие параметры, необходимые для работы **в сети TCP/IP**.

Порядок выполнения работы:

Часть 1. ORACLE Virtual Box

1. Запустить **Virtual Box** и ознакомиться с интерфейсом управления средой виртуализации. Основные элементы управления представлены в **приложении 1**.
2. Создать **три виртуальные машины**, используя образы жёстких дисков, предоставляемые преподавателем (**две с ОС Windows** и **одну с ОС Linux**).
   Примечание: Параметры создаваемых машин выбрать самостоятельно, учитывая технические характеристики компьютера.
3. Создать для любой виртуальной машины **снимок начального состояния**.
4. Используя виртуальные машины с **ОС Windows**, созданные в **пункте 2**, изучить различия следующих **типов подключения** при настройке сетевых адаптеров:
   - **NAT**
   - **Сетевой мост**
   - **Внутренняя сеть**

   Примечание 1: При использовании типа «**Сетевой мост**» требуется настроить виртуальные машины так, чтобы они имели доступ **к сети Internet** и всем локальным ресурсам **основного компьютера**.
   Примечание 2: При использовании типа «**Внутренняя сеть**» требуется создать **на каждой** виртуальной машине **сетевую папку**. Обе папки должны **быть доступны для обеих машин**.
5. Подключить к обеим виртуальным машинам сетевую папку **основной операционной системы**.
6. Создать для ранее выбранной виртуальной машины **снимок конечного состояния системы**, вернуть исходное состояние, используя ранее созданный снимок, а затем восстановить конечное. Убедиться в работоспособности виртуальной машины, проверив все установленные настройки.

Часть 2. MS Windows

1. Запустить виртуальную машину и авторизоваться в системе под администраторской учётной записью, используя заданное преподавателем **имя пользователя и пароль**. Проверить, активны ли следующие **пункты** в свойствах **используемого сетевого подключения**, и определить их назначение:
   - **Клиент для сетей Microsoft;**
   - **Служба доступа к файлам и принтерам Microsoft;**
   - **Протокол TCP/IP.**
2. Установить следующие параметры в свойствах протокола **TCP/IP**:
   - **IP 192.168.1.10;**
   - **mask 255.255.255.0;**
   - **gateway 192.168.1.1;**
   - **DNS 192.168.1.254.**
3. Используя знания, полученные в **пункте 1**, настроить сетевой интерфейс таким образом, чтобы **внешние пользователи не могли получить доступ** к ресурсам компьютера.
4. Разобраться в назначении параметров и ключей следующих утилит:
   - **ping**
   - **ipconfig**
   - **net** с директивами **use** и **view**
   - **netsh** с контекстом **interface**
5. С помощью утилиты **netsh** создать командные файлы для интерпретатора **CMD.exe**, с помощью которых можно было бы настраивать выбранный сетевой интерфейс двумя способами:
   - получение всех настроек через **DHCP-сервер** (автоматически) (**IP**, **mask**, **gateway**, **DNS**);
   - ввод всех настроек **вручную** (статически).

   Примечание: В качестве сетевых настроек использовать параметры из **пункта 2**.

Часть 3. Linux

1. Запустить виртуальную машину и авторизоваться в системе под администраторской учётной записью.
2. Разобраться в назначении параметров и ключей утилиты **ifconfig.**
3. Создать исполняемый файл, настраивающий выбранный сетевой

интерфейс двумя способами:

- получение всех настроек через **DHCP-сервер** (автоматически) (**IP**, **mask**, **gateway**, **DNS**)
- ввод всех настроек **вручную** (статически)

В качестве статических настроек использовать следующие данные:

- **IP 172.16.10.50**
- **Mask 255.255.0.0**
- **Gateway 172.16.0.1**
- **DNS 172.16.255.254**

Содержание отчёта:

В отчёт должны быть включены ответы на следующие вопросы:

1. Перечислите основные отличия **типов подключений** при настройке сетевых адаптеров в Virtual Box.
2. Что произойдёт, если у двух созданных виртуальных машин поменять местами **образы жёстких дисков**?
3. Для чего необходимы **«снимки»** виртуальных машин?
4. Как с помощью графической оболочки **Windows** можно запретить доступ через определенный сетевой интерфейс к ресурсам используемого компьютера? Как можно запретить используемому компьютеру доступ к ресурсам других компьютеров в сети **Microsoft**?
5. Как с помощью **ipconfig** узнать адрес **DNS**, на который настроен ваш компьютер?
6. Зачем нужна команда **net use**? Как с помощью этой утилиты подключить на локальный диск **R:** папку **TEST** на компьютере **SRV** (приведите командную строку)?
7. В чем назначение утилиты **ping**?

В отчёте необходимо предоставить тексты исполняемых файлов из пункта 5 части 2 и пункта 3 части 3 лабораторной работы, а также скриншоты с информацией о рабочей сессии для каждой из созданных виртуальных машин.

Практическая работа 3.
Анализ трафика в сетях Ethernet

Цель работы:

- Получить практические навыки по работе с **анализаторами сетевого трафика**;
- На практике ознакомиться **с различиями в принципах работы** активного сетевого оборудования;
- Уяснить **особенности взаимодействия** сетевого и канального уровней на примере **стека TCP/IP**;
- Выяснить **отличия** форматов кадров **Ethernet**.

Необходимо:

- Компьютер под управлением **MS Windows 2000/XP/2003** или **Linux**, подключенный к локальной сети;
- Пользователь с **администраторскими** правами;
- Сетевое подключение **по протоколу IP**;
- Доступ к глобальной **сети Интернет**.
- Программный пакет **Wireshark**.

Порядок выполнения работы:

1. Изучить назначение утилиты **arp**. Установить какие из **широковещательных** сообщений принадлежат протоколу ARP и для чего они предназначены.
2. Запустить программу **Wireshark** и получить сетевую статистику длительностью в **150 секунд**.
   Примечание: для увеличения интенсивности генерации кадров открыть любой **информационный** сайт в браузере.
3. Осуществить **визуализацию** полученных данных при помощи пункта меню построения графиков **Io Graphs**.
4. Используя сведения из пункта меню **Summary** определить длительность процесса анализа, количество захваченных пакетов, количество байтов, средний размер пакета, среднюю скорость передачи в Mbit/sec.
5. **Визуализировать** информационные потоки, образовавшиеся в результате работы при помощи пункта меню **Flow Graph.**
6. Выделить из общего числа **пакеты службы DNS.**
7. Определить **разницу во временах получения 1 и 2 пакетов** выделенных в предыдущем пункте.

8. Создать новый фильтр и захватить **5 Mb трафика**.
9. Создать **Display Filter** и выделить из общего числа пакеты по протоколам **TCP и UDP** предназначенные **для 80 порта**.
10. Создать **собственный фильтр**, захватывающий **30 пакетов** из трафика между используемым компьютером и сайтом **vkontakte.ru**.
11. Найти **широковещательные кадры и пакеты**. Изучить их **заголовки**. Выяснить их назначение. **Определить адреса**, на которые поступают данные кадры и пакеты **для канального и сетевого уровня.**
12. На основании собранной статистики определить, **к какому типу коммутационного оборудования** подключен используемый компьютер.
    Примечание: в качестве коммутационного оборудования могут выступать **хаб**, **коммутатор** или **маршрутизатор** (см. **Приложение 5**).

В отчет:

1. Предоставить скриншоты результатов выполнения пунктов 3 и 5.
2. Сведения, определённые в пункте 4.
3. Текст фильтра, созданного в пункте 10.

Также в отчёте предоставить ответы на вопросы:

1. Какие **типы кадров Ethernet** бывают, в чем **их отличия**?
2. Какой тип кадров Ethernet используется в анализируемой сети? Почему именно он?
3. Как можно определить тип используемого коммутационного оборудования, используя сетевую статистику?
4. На какие адреса **сетевого** уровня осуществляются широковещательные рассылки?
5. На какой **канальный** адрес осуществляются широковещательные рассылки?
6. Для чего применяются **перехваченные** широковещательные рассылки?
7. Как с помощью утилиты **arp** просмотреть **arp-кэш** и как его **очистить**. В каких случаях может понадобиться последняя операция.

Практическая работа 4.
Выбор коммутационного оборудования

Цель работы:

- Получить практические навыки подбора коммутационного оборудования по критериям различной степени формализации;
- Приобрести опыт работы с описаниями и техническими спецификациями оборудования.

Необходимо:

- Доступ к сети интернет для поиска справочной информации

Порядок выполнения работы:

В соответствии с вариантом подобрать **активное сетевое оборудование**, способное обеспечить весь **необходимый функционал**, требуемый в задании (см. приложение №6).

Каждый вариант состоит из **трёх типов задач**, требующих различные методы и подходы для их решения.

При подборе оборудования необходимо соблюдать принцип **минимизации финансовых затрат**.

Ограничения по производителям оборудования нет, однако рекомендуется обратить внимание на оборудование **LinkSys**, **CISCO**, **D-LINK**, **ASUS**, **HP**.

Вариант 1

1. Подобрать коммутатор с **48 портами** Fast Ethernet и **двумя портами** Gigabit Ethernet, поддерживающий технологию управления потоком **IEEE 802.3x**.
2. Подобрать коммутационное оборудование для сети **небольшого офиса**. В состав сети входят **15 компьютеров с равным уровнем доступа**. Максимальная нагрузка на сеть возможна при одновременном доступе к файловой базе данных **объемом 96 Мб**. Обеспечить возможность подключения существующей **IDS** (системы обнаружения вторжения), осуществляющей **мониторинг** всего передаваемого внутри локальной сети **трафика**.
3. Подобрать коммутационное оборудование для сети **крупного автосервиса**. Требуется создать инфраструктуру для обслуживания **6 ремонтных боксов**. Необходимо обеспечить работоспособность **специализированного программного**

обеспечения и **доступность** всех сетевых **ресурсов** пользователям. Каждый сотрудник имеет **коммуникационное устройство** с беспроводным интерфейсом, которое служит для оповещения о поступивших заказах и контроля за их выполнением. Каждое из них должно строго **контролироваться** и работать **на всей территории** автосервиса. Расстояние между наиболее удаленными точками территории автосервиса 340 метров.

Вариант 2

1. Подобрать **неуправляемый** коммутатор с **16 портами** 10/100/1000Base-T и поддержкой технологии **IEEE 802.1p QoS**.
2. Подобрать коммутационное оборудование для проведения **чемпионата России по киберспорту.** Необходимо обеспечить совместную работу **минимум 90 компьютеров**. Следует избежать ситуации задержек в игре из-за недостаточной производительности коммутационного оборудования. Пиковый трафик, генерируемый средней современной сетевой игрой, составляет **10Мб\с**. Предусмотреть возможность компактной установки коммутационного оборудования **в стойку**.
3. Подобрать коммутационное оборудование для **телевизионной компании**. Требуется обеспечить раздельную работу 4 студий, каждая из которых должна работать в собственной VLAN сети. Количество компьютеров в студиях 40.

Вариант 3

1. Подобрать коммутатор с возможностью подключения **7 IP-видеокамер** по проводной сети Fast Ethernet с возможностью обеспечивать **электропитание камер по линии связи** (Power over Ethernet).
2. Подобрать коммутационное оборудование для сети **крупного предприятия**. Требуется организовать **изолированные потоки данных** для разных отделов. Также необходимо создать высокоскоростной **back-bone (выделенную магистральную сеть)** для связи отделов между собой с возможностью **доступа к ресурсам и сервисам** предприятия. На предприятии **25 отделов**. В каждом отделе до **30 компьютеров**.
3. Подобрать коммутационное оборудование для сети **общеобразовательной школы,** в которой имеется **несколько небольших компьютерных классов**. Требуется учесть

дальнейшее **увеличение парка машин** и **возможность удалённого управления** всем сетевым оборудованием. Также необходимо обеспечить **распределение нагрузки сети** таким образом, чтобы исключить возможность **намеренного блокирования** каналов связи.

Вариант 4

1. Подобрать коммутатор третьего уровня с минимум **44 портами** FastEthernet с поддержкой протокола **OSPF**, **зеркалирование портов** в режиме Many-to-one.
2. Подобрать коммутационное оборудование для сети **студии видеомонтажа**. В студии создан вычислительный кластер для обсчета цифрового видео из **4 компьютеров**. Оборудование должно быть гарантированно **неблокирующим**, то есть обладать внутренней шиной такой производительности, чтобы гарантированно обработать максимально возможные потоки между всеми нагруженными портами коммутатора.
3. Подобрать коммутационное оборудование для **загородного ресторанного комплекса**. Комплекс состоит из **5 залов** и **2 открытых веранд**. В каждом зале находятся **4 терминала** для управления заказами, а на верандах **по 2**. Требуется обеспечить работу терминалов управления заказами во всех помещениях, доступность терминалам **10 сетевых принтеров** и возможность работы **трём компьютерам менеджеров**.

Вариант 5

1. Подобрать управляемый коммутатор второго уровня с **минимум 8 портами** FastEthernet и **двумя оптическими** портами **SFP**.
2. Подобрать коммутационное оборудование для ядра крупной **корпоративной сети**. Обеспечить коммутацию **18 каналов** от подразделений, каждый из которых имеет пропускную способность в **100 Мб\с**. Необходимо реализовать фильтрацию на основе **IP адресов** и автоматический **мониторинг** состояния оборудования.
3. Подобрать коммутационное оборудование для **городской больницы**. Требуется обеспечить доступ к **общей больничной базе** во всех кабинетах и к глобальной **сети интернет**. Необходимо предусмотреть возможность **блокирования доступа** к базе из **внешней сети** и **доступ в интернет по WiFi** для посетителей на всей территории больницы.

Вариант 6

1. Подобрать управляемый коммутатор **второго уровня** с минимум **16 портами** FastEthernet и поддержкой **Spanning Tree**.
2. Подобрать коммутационное оборудование для использования в качестве **узловых точек** растущей сети кабельного **интернет-провайдера**. Необходимо обеспечить **удаленное управление** устройством и **возможность подключения** к нему точек доступа **WiFi** без прокладки к ним линий **электропитания**.
3. Подобрать коммутационное оборудование для информационной сети **студенческого общежития**. Необходимо обеспечить высокоскоростную передачу данных между всеми узлами сети. Общежитие имеет **4 этажа**, следовательно, необходима **магистраль передачи данных** между этажами. На каждом этаже по **100 комнат**, в каждой из которых должен быть доступ к сети. Необходимо обеспечить **контроль** распределения адресов в сети и **мониторинг** сетевого трафика.

Вариант 7

1. Подобрать коммутатор **третьего уровня** с возможностью **объединения в стек**, минимум с **30 портами** FastEthernet и **фильтрацией по IP** адресам.
2. Подобрать коммутационное оборудование для **DATA-центра** хостинговой компании. Через сеть в среднем передается **4 Терабайта** в день. Необходимо обеспечить соединение сетей с **разными канальными протоколами** (**FastEthernet**, **GigabitEthernet** на витой паре и **FastEthernet** по оптическим каналам), обеспечить масштабируемость решения.
3. Подобрать коммутационное оборудование для проведения **выставки информационных технологий**. Требуется обеспечить **зону покрытия WiFi** на всей территории выставки, а также возможность **удалённого управления** цифровыми проекторами. Координация выставки будет происходить и **специального центра**, который представляет собой **несколько компьютеров**. Все они должны иметь **доступ к сети,** и **только они** должны иметь **доступ к управлению** проекторами.

Вариант 8

1. Подобрать неуправляемый коммутатор минимум с 7 портами 10/100Base-TX и 1 оптическим портом 100Base-FX.

2. Подобрать коммутационное оборудование для **локальной сети**, компьютеры в которой расположены двумя группами в **двух помещениях**, которые в настоящий момент удалены друг от друга на расстояние (по кабельной трассе) **90 м**. В каждом помещении находятся **20 компьютеров**. При подборе оборудования необходимо учесть **скорый переезд одного отдела** в соседнее здание на расстояние по кабельной трассе **1800 м**. Необходимо обеспечить **минимальные финансовые затраты** и не приобретать оборудование, которое может не понадобиться.
3. Подобрать коммутационное оборудование для **главного узла** компании, занимающейся **продажей трафика** через свою сетевую инфраструктуры. Требуется обеспечить максимально возможную **пропускную способность** и **полезную скорость** передачи данных, компактность и масштабируемость решения.

Содержание отчёта:

В отчёт входит краткая пояснительная записка, в которой обосновывается выбор того или иного активного оборудования. В ней указывается:

- модель выбранного оборудования;
- характеристики, обеспечивающие решение поставленных задач;
- стоимость устройства;
- дополнительные параметры и характеристики, говорящие в пользу вашего выбора;
- рекомендации по организации разработанной сетевой структуры.

Практическая работа 5.
Работа с адресами IP сетей

Цель работы: получить практические навыки по работе с пространством IP-адресов, масками и управления адресацией в IP сетях.

Необходимо:

- Навыки по переводу чисел из десятичной в двоичную систему и наоборот, в ручном режиме или с помощью компьютера.

Краткие теоретические сведения

Все пространство IP адресов делится на логические группы – IP-сети. Они нужны для организации иерархической адресации в составной IP-сети, например Интернете. Каждой локальной сети присваивается своя IP-сеть, маршрут до IP-узлов, находящихся в этой локальной сети строится на маршрутизаторах как маршрут до их IP-сети. И только после того, как пакет попал в конкретную IP-сеть, решается задача его доставки на отдельный узел.

В IP-адресе выделяются две части – адрес сети и адрес узла. Деление происходит с помощью маски – 4-х байтного числа, которое поставлено в соответствие IP-адресу. Макса содержит двоичные 1 в тех разрядах IP-адреса, которые определяют адрес сети и двоичные 0 в тех разрядах IP адреса, которые определяют адрес узла.

Адресом IP-сети считается IP-адрес из этой сети, в котором в поле адреса узла содержатся двоичные 0. Этот адрес обозначает сеть целиком в таблицах маршрутизации. Есть еще служебный IP-адрес – адрес ограниченного широковещания – в поле адреса узла он содержит двоичные 1. Оба эти адреса не используются для адресации реальных узлов сети, однако входят в диапазон адресов IP-сети.

Рассмотрим пример: есть адрес 192.168.170.15 с маской 255.255.252.0. Определим адрес сети, адрес широковещания и допустимый для данной IP-сети диапазон адресов.

DEC IP	192	168	170	15
DEC MASK	255	255	252	0
BIN IP	11000000	10101000	10101010	00001111
BIN MASK	11111111	11111111	11111100	00000000
	С фоном – адрес сети, без фона – адрес узла			

BIN IP сети	11000000	10101000	10101000	00000000
	скопируем сетевую часть IP и заполним узловую часть 0			
DEC IP сети	192	168	168	0
BIN IP	11000000	10101000	10101011	11111111
	Адрес широковещания (скопируем сетевую часть IP и заполним узловую часть 1)			
DEC IP широковещания	192	168	171	255
Начало диапазона IP-адресов для узлов	192	168	168	1
	(значение поля узла +1 к IP адресу сети)			
Окончание диапазона IP-адресов для узлов	192	168	171	254
	(значение поля узла -1 от IP-адреса широковещания)			

Если имеется сеть, составленная из нескольких локальных сетей, соединенных между собой маршрутизаторами, то нужно каждой из этих локальных сетей назначить отдельную IP-сеть. В случае, если для такой сети выдается большая IP-сеть в управление (например, такую сеть может назначить провайдер Интернет), то эту сеть необходимо разделить с помощью масок на части.

Примечание: необходимо отметить, что подобная ситуация может иметь место, если вам необходимо назначить узлам вашей сети реальные IP адреса, для того чтобы ваши компьютеры были «видны» из Интернета каждый под своим адресом.

Порядок выполнения работы:

В работе даны 4 варианта задания (таблица 2). Необходимо сделать все варианты. На приведенной схеме представлена составная локальная сеть. Отдельные локальные сети соединены маршрутизаторами. Для каждой локальной сети указано количество компьютеров. Провайдер выдал IP-сеть (данные о сети представлены в таблице 2). Необходимо установить IP-адрес сети и допустимый диапазон адресов. Разделить

сеть на части, используя маски. Маску надо выбирать так, чтобы в отделяемой IP подсети было достаточно адресов.

Примечание: порт маршрутизатора, подключенный к локальной сети, имеет IP адрес!

Выделять диапазоны следует, начиная с самой большой сети. Некоторые маски представлены в таблице 3.

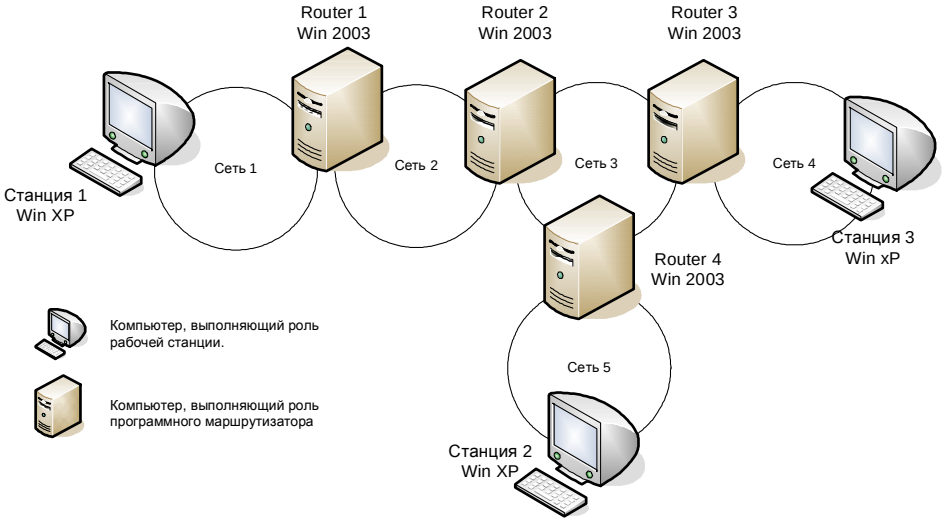


Таблица 2

Вар	IP- адрес из сети маска	Количество компьютеров в сети				
		Сеть 1	Сеть 2	Сеть 3	Сеть 4	Сеть 5
1	192.169.168.70 255.255.248.0	500	16	19	200	100
2	172.21.25.202 255.255.255.224	30	3	2	12	6
3	83.14.53.9 255.255.255.128	10	12	8	3	8
4	190.23.23.23 255.255.255.192	5	3	3	3	3

Таблица 3

Маска	Количество двоичных 0	Количество всех адресов в IP сети с такой маской
255.255.255.252	00	4

255.255.255.248	000	8
255.255.255.240	0000	16
255.255.255.224	00000	32
255.255.255.192	000000	64
255.255.255.128	0000000	128
255.255.255.0	00000000	256
255.255.254.0	0.00000000	512

В отчет:

В качестве отчета предоставить результаты расчетов в табличной форме

Вариант:					
Сеть	Сеть 1	Сеть 2	Сеть 3	Сеть 4	Сеть 5
IP-сети, маска					
Количество IP адресов в IP-сети					
Начальный и конечный адреса сети, пригодные для адресации портов маршрутизаторов и компьютеров.					

Практическая работа 6.
Конфигурирование межсетевого экрана

Цель работы:

- получить представление о работе классического межсетевого экрана.
- Закрепить понимание адресации на сетевом и транспортном уровне стека **TCP/IP**.

Необходимо:

- Установленная на компьютере среда виртуализации **ORACLE Virtual Box;**
- Образы виртуальных жёстких дисков операционных систем **Windows 2003** и **Linux;**
- Доступ к сети Интернет;
- Учетные записи пользователей с **администраторскими** правами.

Краткие теоретические сведения:

Под межсетевым экраном или брандмауэром понимают **фильтр IP пакетов** предназначенный для формального ограничения соединений клиентов и серверов работающих «поверх» **стека TCP\IP**.

В основу работы классического firewall положен **контроль формальных признаков.** В общем случае фильтрация осуществляется по:

- **IP адресам** отправителя и получателя в заголовке IP пакета;
- **номерам портов** приложения-получателя и приложения-отправителя инкапсулированным в IP протокол транспортного (TCP, UDP) и сетевого уровней (ICMP).

Правила фильтрации формируются **в виде списка**. Все проходящие пакеты проверяются по списку **последовательно**, до первого срабатывания. Последующие правила к пакету **не применяются**.

Для конфигурирования firewall **в Linux** необходимо сформировать набор правил **iptables**. В iptables реализовано несколько цепочек правил **INPUT** для входящего трафика, **OUTPUT** для исходящего и **FORWARD** для пересылаемого. Управление цепочками производится с помощью консольной команды iptables.

Примеры:

- iptables -A INPUT -s ws.mytrust.ru -j ACCEPT включает прием всех пакетов с хоста ws.mytrust.ru

- iptables -A OUTPUT -d mail.ifmo.ru --dport 25 -j DROP запрещает отправку всех пакетов на хост mail.ifmo.ru на порт 25
- iptables -A INPUT -j DROP запрещает прием всех сообщений.

В протоколах **TCP** и **UDP** (семейства TCP/IP) **порт** — идентифицируемый номером системный ресурс, выделяемый приложению, выполняемому на некотором сетевом хосте, для связи с приложениями, выполняемыми на других сетевых хостах (в том числе с другими приложениями на этом же хосте).

Порядок выполнения работы:

Часть 1. Windows

1. Разобраться в назначении параметров и ключей следующих утилит:
   - **sc;**
   - **netsh** с директивами **firewall** и **diag;**
   - **netstat.**
2. Создать **скрипт**, который:
   - включает **автоматическую** загрузка **Брандмауэра Windows;**
   - запускает брандмауэр;
   - включает протоколирование **входящих** соединений;
   - настраивает службу **Telnet** на **ручной запуск;**
   - добавляет **правило**, разрешающее доступ с **IP** адресов сети компьютерного класса к службе **Telnet;**
   - разрешает системе отвечать на запросы **echo-request ICMP;**
   - запускает службу **Telnet;**
3. Запустить сеанс **Telnet** из реального компьютера в **гостевую ОС.**
4. В гостевой ОС вывести на экран данные только об **установленных соединениях** со службой **Telnet**, с указанием **IP адресов** и **портов** в **численной** форме.
5. В гостевой ОС проверить **доступность** службы **Telnet** на виртуальной машине.

Часть 2. Linux

1. Разобраться в назначении параметров и ключей утилит **iptables** и **iptables-save**
2. Сконфигурируйте **firewall** в **Linux** следующим образом:

- Должен быть доступен **DNS сервер** с адресом **194.85.32.18**
- Должны быть доступны все **наружные Web сервера и HTTP прокси** с адресом шлюза **proxy.ifmo.ru**
- Должен быть доступен **FTP сервер ftp.ifmo.ru**,
- Должны быть доступны **все наружные POP3 сервера**
  Примечание: Связь по открытым портам в этом правиле можно устанавливать только из защищаемой системы.
- Должен быть доступен **SMTP сервер mail.ifmo.ru**,
- Со всех узлов подсети **83.0.0.0/16** должен быть доступен **SSH сервер** на используемом компьютере
- Отдельно должен быть **заблокирован доступ** к системе с хоста **10.10.11.173**
- Используемая система не должна отвечать на запросы команды **PING**
- Работа с **остальными сервисами** должна быть **блокирована**

Содержание отчёта:

В отчёт должны быть включены ответы на следующие вопросы:

1. От чего не способен защитить **классический firewall**?
2. Можно ли организовать доступ к **Web серверу**, если у клиентов закрыт доступ к **80 порту**?
3. В чем отличие правил **Deny** и **Drop**?
4. Каким образом осуществляется **оптимизация правил**, используемых в работе **firewall**?
5. Перечислите ограничения брандмауэра Windows относящиеся к фильтрации трафика TCP/IP.

Также в отчёте необходимо предоставить:

1. Список правил **iptables**
2. Команды по созданию правил Windows-брандмауэра.

Практическая работа 7.
Маршрутзация в IP сетях

Цель работы:

- Получить представление о работе IP маршрутизатора;
- Порпактиковаться в составлении таблиц маршрутизации и работе протоколов внутренней маршуртизации;
- Дополнительной целью работы является приобретение опыта работы в средах виртуализации.

Необходимо:

- Семь компьютеров, объединенных локальной сетью.
- Установление на них программа **ORACLE Virtual Box**.
- Виртуальные машины **Windows 2003 Server** и **Windows XP**.
- Понимание структуры IP адресов и принципов маршрутизации.

Краткие теоретические сведения:

Маршрутизаторы (аппаратные или программные) выполняют задачу выбора оптимального маршрута следования **IP пакета** и его отправки по этому маршруту. Для принятия решения анализируется адрес получателя и устанавливается маршрут следования на основе неких формализованных записей о структуре составной сети. Эти записи называются **таблицами маршрутизации**.

В таблице маршутизации присутствуют как минимум следующие поля: **адрес назначения** (адрес IP-сети или IP адрес хоста), **идентификатор порта**, через который пакет идет до сети назначения (порт обозначается IP-адресом или внутренним номером), **шлюз** (IP адрес на который необходимо пойти после того как пакет покинет порт), **метрика** (показатель качества маршрута).

На каждом маршрутизаторе сети присутствует таблица, полностью описывающая структуру всей сети и, иногда, содержащая записи о маршрутах по умолчанию.

Таблицы маршрутизации составляются вручную или с помощью протоколов маршрутизации, автоматизирующих этот процесс. Одним из таких протоколов является протокол **RIP2**.

Использование виртуальных машин в этой работе обусловлено исключительно соображениями удобства развертывания нескольких операционных систем на одном компьютере и не связано напрямую с главной целью работы.

В работе операционные системы **Microsoft ©** могут заменяться на любые другие при условии, что последние поддерживают программную маршрутизацию **IP**.

Так же в работе сделано **еще одно допущение**: реально, маршрутизатор объединяет несколько локальных сетей, имея по интерфейсу (порту) в каждой локальной сети. В случае программного маршрутизатора, работающего в составе ОС, **в качестве портов** выступают **сетевые карты** на компьютере.

В работе программный маршрутизатор использует единственный интерфейс с двумя IP адресами для доступа к единой локальной сети, маршрутизируя фактически изолированные IP потоки.

Порядок выполнения работы:

Часть 1. Packet Tracer:

1. Реализовать схему, приведенную на рисунке 1, смоделировав ее в программе **Packet Tracer**.
2. Воспользовавшись расчетами адресов из **лабораторной работы номер 5** (любым из четырех вариантов), назначить адреса компьютеров, коммутационного оборудования и настроить **статическую маршрутизацию** в данной сети.
3. Проверить возможность передачи пакетов данных между узлами модели.

Часть 2:

1. Используя виртуальные машины реализовать работу составной IP сети, схема которой представлена на рисунке 1.
   Примечание: В сети **7 компьютеров**. **Три** рабочих станции и **четыре** маршуртизатора. В качестве этих компьютеров будут выступать **гостевые операционные системы**. Рабочие станции будут работать под **windows XP или Windows 2003**, а программные маршрутизаторы под управлением **Windows 2003** (XP не поддерживает программную маршрутизацию). Все компьютеры будут подключены к одной локальной сети, а в ней организуются изолированные по адресам IP сети. Одновременная работа одного сетевого интерфейса в разных IP сетях достигается назначением двух и более IP адресов одному сетевому интерфейсу (Свойства сетевого соединения Свойства TCP\IP \ Дополнительно).

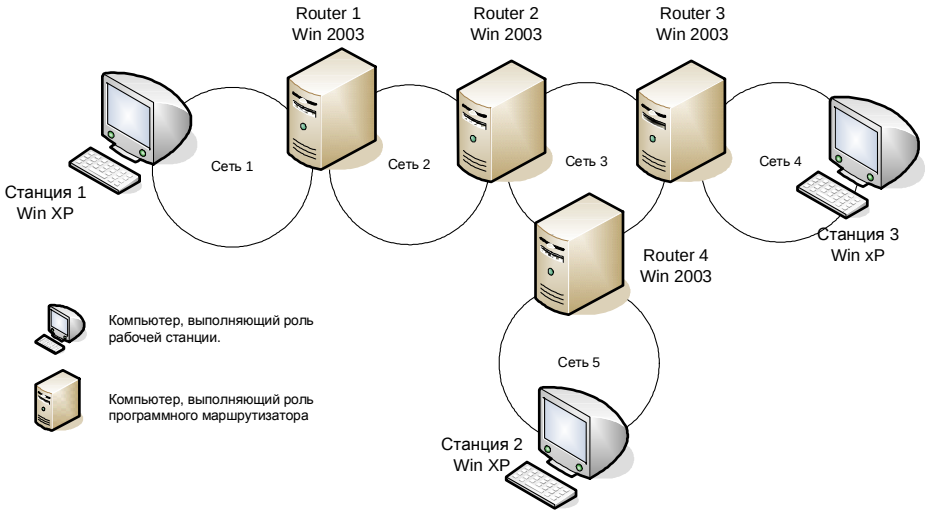


Рис. 1

2. На первом этапе необходимо **составить план сети** (заранее выбрать IP-адреса для сетей, рабочих станций и портов маршрутизаторов). Можно использовать результаты моделирования в первой части работы. Задать уникальное имя для каждого виртуального компьютера.
3. В виртуальной операционной системе:
   - Поменять **MAC адрес сетевой платы** на новый уникальный (Свойства сетевого соединения / Настройка сетевого адаптера / Сетевой адрес). Делать это необходимо из-за того, что виртуальные машины созданы из одной копии и, следовательно, обладают идентичными MAC адресами, что приводит к неправильной работе коммутатора локальной сети.
   - Изменить имя компьютера (панель управления / свойства системы / имя)
   - Установить все необходимые IP адреса.

   Примечание: На рабочих станциях необходимо указывать шлюз по умолчанию – IP адрес порта маршрутизатора из IP сети.
   На маршрутизаторах делать этого не следует – маршрута по умолчанию нет. Адреса **DNS** остаются **пустыми**.

4. С помощью команды **PING** проверить **видимость** ближайших соседей по локальной сети.
5. На маршрутизаторах запустить службу **Routing and Remote Access** (Панель управления / Администрирование / Routing and Remote Access). С помощью мастера сконфигурируйте службу, как LAN Router.
6. С помощью консольной команды **ROUTE** изучить таблицу маршрутизации по умолчанию.
7. С помощью консольной команды ROUTE (рекомендуемый способ) или с помощью графической консоли службы Routing and Remote Access **дополнить таблицу** необходимыми записями.
8. С помощью команды ping проверить достижимость рабочих станций друг с друга, а с использованием команд **tracert** и **pathping** проверить **путь** следования IP пакетов.
9. **Сохранить** таблицы маршрутизации **в текстовом файле**.
10. **Удалить** созданные вручную записи в таблицах маршрутизации. Примечание: рабочие станции **не должны** определять друг друга в сети.
11. Добавить **в консоли службы** Routing and Remote Access на маршрутизаторах протокол **RIP2** (General / Add new routing protocol). В нем добавить **интерфейс**, через который будет происходить обмен векторами маршрутизации. Установить интервал обмена **60 секунд**.
12. Обновить **консоль** и убедиться, что пошли **рассылки** таблицы. После получения **чужих** таблиц вывести таблицу **динамической** маршрутизации (IP-routing / Routing tables)
13. После того, как **будут получены** все необходимые записи, с помощью команды ping проверить достижимость рабочих станций, а с использованием команд tracert и pathping проверить путь следования IP пакетов.
14. Отключить службу Routing and Remote Access и установить **автоматическое получение IP** адресов на витуальных системах.

В отчет:

1. Скриншот схемы сети из Packet Tracer (часть 1).
2. Таблицы маршрутизации всех маршрутизаторов (часть 2).
3. Скриншот таблиц маршрутизации из Packet Tracer (часть 1).

Ответы на вопросы:

1. Как в таблице маршрутизации MS отличить маршрут на хост от маршрута на сеть?
2. Как в таблице маршрутизации MS отличить маршрут по умолчанию?
3. Как с помощью команды route вывести таблицу маршрутизации, добавить и удалить маршрут?
4. Какие методы предотвращающие возникновение ложных маршрутов в RIP2 включены на маршрутизаторе MS по умолчанию?

Для тех, кто решит выполнить работу в иной ОС (например Linux) следует готовить этот отчет в терминах и применительно к другой ОС.

Практическая работа 8.
DNS

Цель работы:

- Получить представление о работе DNS сервера.
- Получить практические навыки использования утилит работы с серверами системы DNS и конфигурирования системы.

Необходимо:

- Установленная система виртуализации;
- Виртуальные машины Windows 2003 Server;
- Представление о работе системы DNS;
- Доступ в Web.

Краткие теоретические сведения:

Система DNS – распределенная база данных хранящая **соответствие** между **IP адресом** и **доменным именем** компьютера.

Система DNS – **клиент - серверная**. DNS-клиент получает в качестве конфигурационного параметра IP адрес обслуживающего DNS-сервера и получает к нему доступ напрямую.

На сервере DNS могут присутствовать множество записей разных типов и назначения.

Диагностику работы DNS с клиента можно выполнять с помощью команд ping (формальная проверка разрешения имени) и с помощью консольной утилиты **nslookup** (работа с DNS сервером в режиме запрос-ответ).

Порядок выполнения работы:

Часть 1. Освоение утилиты nslookup

1. Используя встроенную справку и доступные материалы в Web выяснить:
   - Назначение и формат следующих типов записей DNS: **SOA, A, NS, MX, CNAME**;
   - Значение и взаимосвязь терминов **«домен»** и **«доменная зона»**;
   - Значение термина **«зона обратного просмотра»**;
   - Значение термина **«делегирование домена»**.
2. С помощью консольной утилиты **nslookup**:
   - Определить адреса хостов обслуживающих почтовый домен yandex.ru
     Примечание: запрос необходимо выполнить к NS северу сети RunNet (домен runnet.ru), для чего необходимо выяснить имена или адреса DNS серверов зоны runnet.ru.
   - Определить каноническое имя (CNAME) для хоста www.ifmo.ru.
   - Определить e-mail администратора DNS сервера зоны ifmo.ru (запрос можно к DNS серверу зоны ifmo.ru).

Часть 2. Управление и настройка DNS-сервера под Windows Server

1. Подготовить **два** (Б и Д) компьютера с **Windows Server**. Согласовать настройку сети с преподавателем. Проброс сети в виртуальной машине должен быть настроен на режим **«сетевой мост»**.
2. Установить пакет **support tools** (он содержит необходимую для работы утилиту dnscmd.exe). В конфигурации TCP/IP установить согласованный с преподавателем **IP адрес и адрес DNS** равный IP.

3. Разработать план доменного дерева со следующими условиями:
   - **Сервер Б** должен содержать зону, поддерживающую домен **инициалы.local** (например adb.local);
   - **Сервер Б** должен содержать зону **обратного просмотра** для IP сети, в которой будут находится сервера Б и Д;
   - В зоне **прямого просмотра сервера Б** должна быть заведена запись **типа А** для сервера Б;
   - В зоне **прямого просмотра сервера Б** должен быть создан поддомен **sub1.инициалы.local**, все записи которого хранятся в зоне сервера Б;
   - В зоне прямого просмотра сервера Д должен быть создан поддомен **sub2.инициалы.local**;
   - В зоне **прямого просмотра сервера Б** должно быть назначено **делегирование** домена **sub2.инициалы.local** в зону сервера Д;
   - Все ссылки в **SOA** на **DNS** серверах должны быть сделаны через **псевдонимы** с именем **ns**;
   - **Сервер Д** должен содержать **дополнительную** зону обратного просмотра для зоны обратного просмотра с сервера Б, должно быть включено **уведомление** об изменениях и **ограничено** предоставление копии зоны только для сервера Д;
   - В доменах **инициалы.local**, **sub1.инициалы.local** и **sub2.инициалы.local** должны быть **А записи** на хосты с именами **srv** и ip равными **ip-адресам** сервера, поддерживающего домен, в котором создается запись.
4. Установить и настроить **DNS сервера** на компьютерах Д и Б согласно **п.5.**
5. Установить, на каких **номерах портов** и по каким **протоколам транспортного уровня** работает DNS сервер.
6. Изучить **содержимое** файлов зон (сохранить их для отчета).
7. С помощью утилит **dnscmd** получить **список всех зон** на обоих серверах, и **содержимого** зоны инициалы.loc (сохранить их для отчета).
8. Разобраться в назначении **других ключей** утилиты dnscmd.
   Убедиться, что на сервере Б корректно разрешается имена:
   - srv.инициалы.local;
   - srv.sub1.инициалы.local;

- srv.sub2.инициалы.local.

Сохранить для отчета вывод команд.

Часть 3. Рекурсивный поиск по дереву DNS

1. Перенастроить **DNS сервер Б**, поменяв IP адрес по указаниям преподавателя и переключив проброс сети в виртуальной машине на **режим «NAT».**
2. Настроить DNS-сервер так, чтобы он запрашивал **внешний сервер** с адресом **194.85.32.18** в случаях, когда сам **не способен** разрешить имена. (Параметр Forwarders в Свойствах сервера).
3. Проверить **корректность** разрешения имени **www.google.ru** при работе через DNS.
4. **Удалить** настройку Forwarders и **очистить кэш** сервера не перезагружая его.
5. Проверить корректность разрешения имени www.google.ru при работе через DNS в **новой конфигурации**.
6. С помощью любой программы анализатора трафика (например, **wireshark**) установить этапы работы алгоритма разрешения имени в п.3 и п. 5. **Сохранить** перехваченные сообщения для отчета.

В отчет:

1. Консольный вывод команды nslookup части 1 п. 2.
2. Файлы зон с среверов Б и Д из части 2 п.6.
3. Вывод команд из части 2 п. 7, 8.
4. Перехваченные сообщения разрешения имени из части 3 п. 6.

Ответы на вопросы:

1. Для чего предназначены основные типы записей DNS?
2. В каком режиме работал DNS-сервер в части 3 п. 3 и в п. 5 (рекурсивном или нет)?
3. Что такое корневые ссылки? Привести несколько адресов корневых DNS серверов «известных» созданному DNS-серверу по умолчанию.
4. Разрешение имени в части 3 п. 3 и п. 5 происходило с разной скоростью. Почему?
5. В чем назначение зоны обратного просмотра?

6. Как определить, какие хосты обрабатывают почту, направленную в домен yandex.ru?

Для тех, кто решит выполнить работу в иной ОС (например Linux) следует готовить этот отчет в терминах и применительно к другой ОС.

Практическая работа 9.
Работа с прикладными протоколами из командной строки

Цель работы: получить представление о принципах работы и практические навыки работы с типичными высокоуровневыми протоколами через текстовые консоли.

Необходимо:

- Установленная на компьютере среда виртуализации **ORACLE Virtual Box;**
- Образ виртуального жёсткого диска операционной системы **Windows 2003;**
- Доступ в глобальную сеть Интернет по протоколам **Web** и **FTP;**
- Терминальные клиенты **ftp** и **telnet**;
- Сгруппироваться **по двое.**

Краткие теоретические сведения:

Для передачи электронных писем необходим почтовый сервер (сервер электронной почты, мейл-сервер), который в системе пересылки электронной почты называется агентом пересылки сообщений (**mail transfer agent, MTA**). Это компьютерная программа, которая передаёт сообщения от одного компьютера к другому. Обычно пользователи работают с почтовыми системами через клиент электронной почты (**mail user agent, MUA**), например Outlook Express или Thunderbird.

Когда пользователь набрал сообщение и посылает его получателю, почтовый клиент взаимодействует с почтовым сервером, используя протокол **SMTP** (Simple Mail Transfer Protocol). Почтовый сервер отправителя взаимодействует с почтовым сервером получателя (напрямую или через промежуточный сервер — **релей**). На почтовом

сервере получателя сообщение попадает в почтовый ящик, откуда при помощи агента доставки сообщений (**mail delivery agent, MDA**) доставляется клиенту получателя. Часто последние два агента совмещены в одной программе (к примеру, **sendmail**), хотя есть специализированные MDA, которые в том числе занимаются фильтрацией спама. Для финальной доставки полученных сообщений используется протокол **POP3** (Post Office Protocol Version 3).

Порядок выполнения работы:

Часть 1. Консольное управление электронной почтой

1. Разобраться в назначении параметров и ключей следующих терминальных утилит:
   - **telnet.exe;**
   - **ftp.exe;**
2. Используя клиент **ftp.exe** получите с сервера **ftp://ftp.asus.com/** из каталога **pub/ASUS/DVR/** файл **e1351_drw-0402p_d.pdf.** Примечание: адрес FTP сервера и имя файла может быть выбрано самостоятельно.
3. Определить **адреса**, используемые для отправки и получения сообщений, **вашего** почтового ящика.
4. Выяснить **номера портов** для серверов отправки и получения электронной почты.
5. Разобраться в назначении и функционировании команд **telnet.exe**, используемых для управления почтой.
6. С помощью **telnet.exe** отправите сообщение со своего почтового ящика на почтовый ящик своего партнёра.
7. С помощью **telnet.exe** прочитайте полученное от вашего партнёра сообщение.

Часть 2. Создание, конфигурирование и тестирование серверов электронной почты

1. Ознакомиться с окном управления **ролями сервера** операционной системы Windows 2003.
2. Добавить роль сервера электронной почты по протоколам **POP3 и SMTP**.
   Примечание: В качестве имени домена использовать собственную фамилию.
3. Создать **двух локальных** пользователей с **уникальными**

адресами электронной почты.

4. Отправить письмо от одного локального пользователя другому и **убедиться** в его поступлении.

Содержание отчёта:

В отчёте необходимо предоставить текстовые команды из всех пунктов части 1 задания, а также скриншоты из 2 части задания с текстом полученного сообщения, адресом отправителя и адресом получателя.

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.):

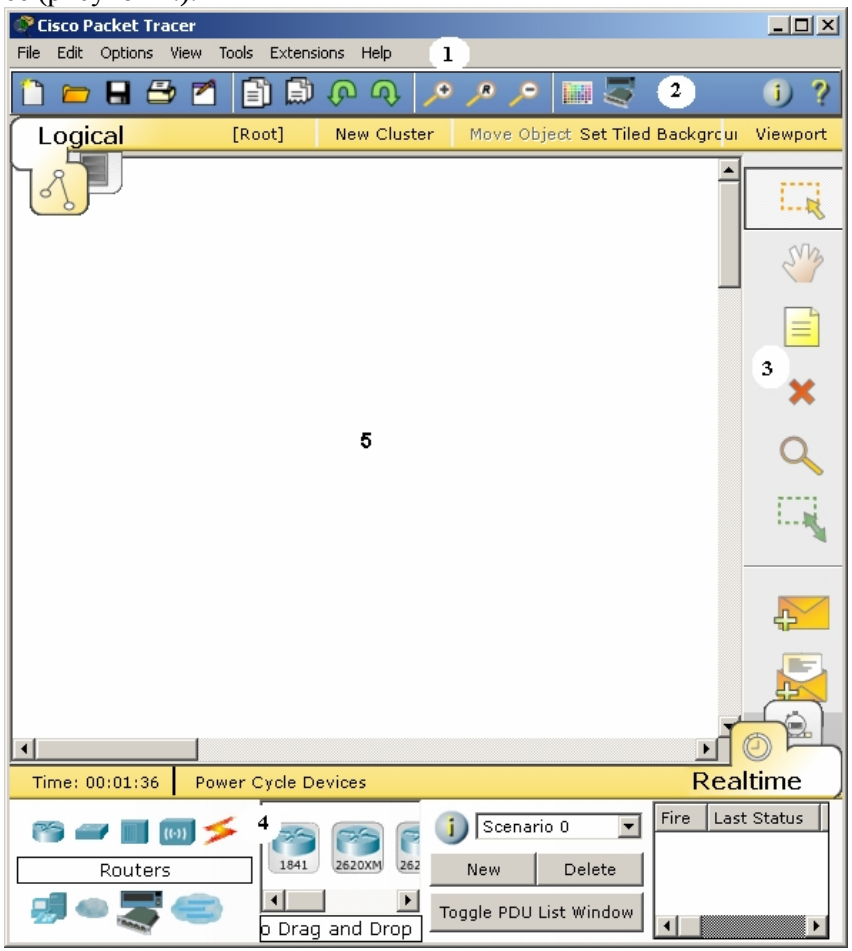


Рисунок 1

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).

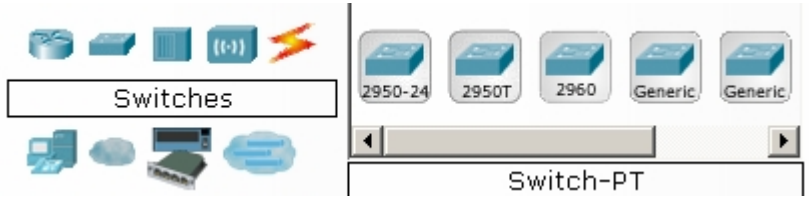


Рисунок 3

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).

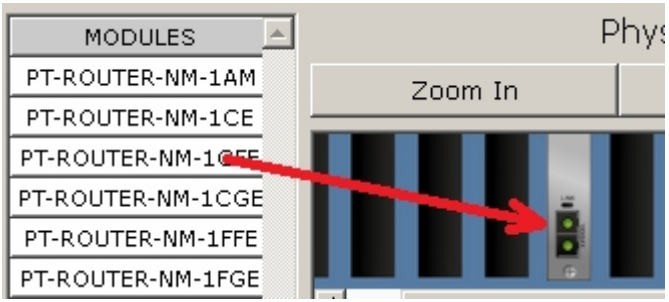


Рисунок 4

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

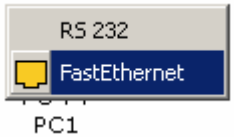


Рисунок 6

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).

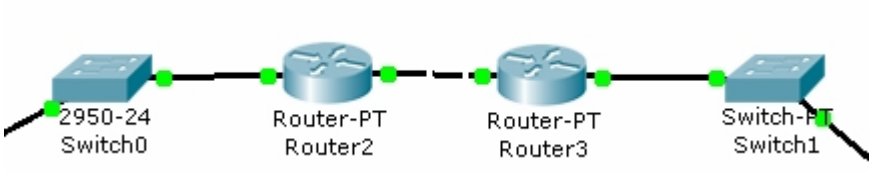


Рисунок 8

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

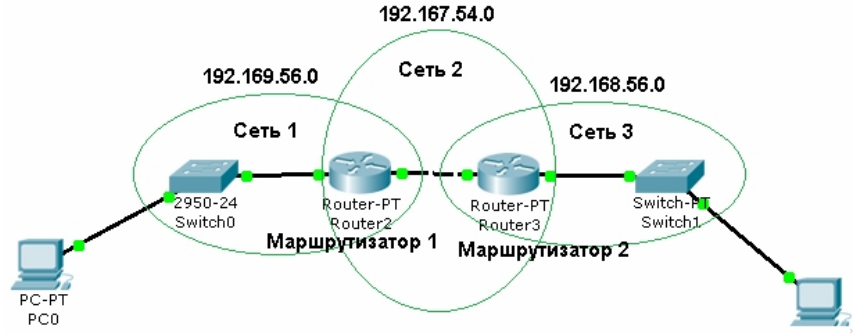


Рисунок 9

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.

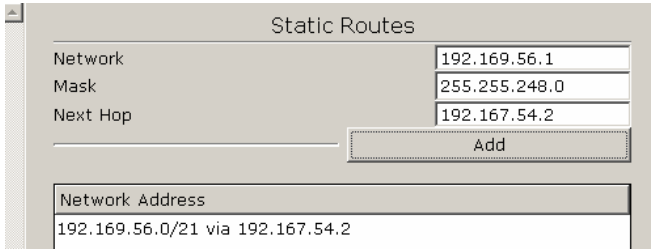


Рисунок 10

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.

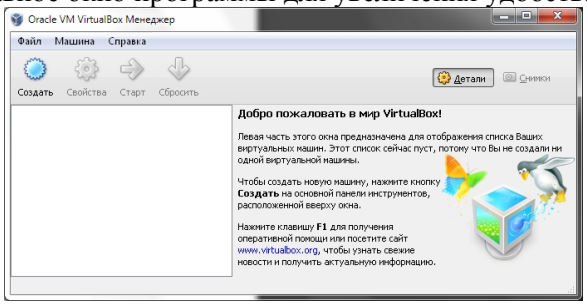


Рисунок 1

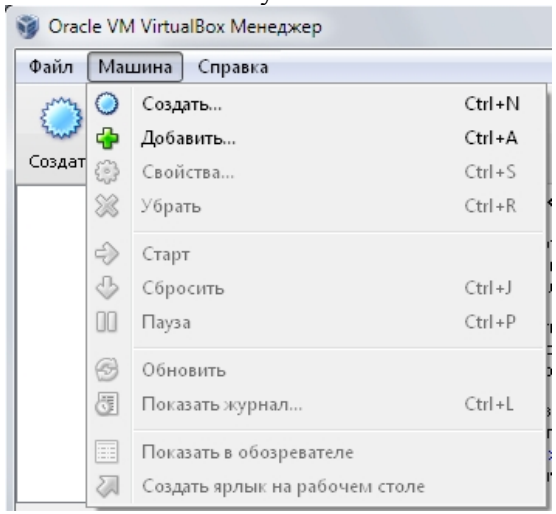


Рисунок 2

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.

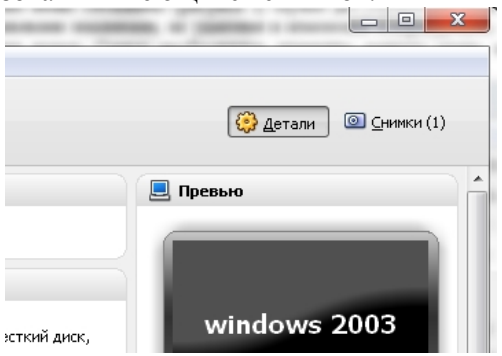


Рисунок. 3

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.

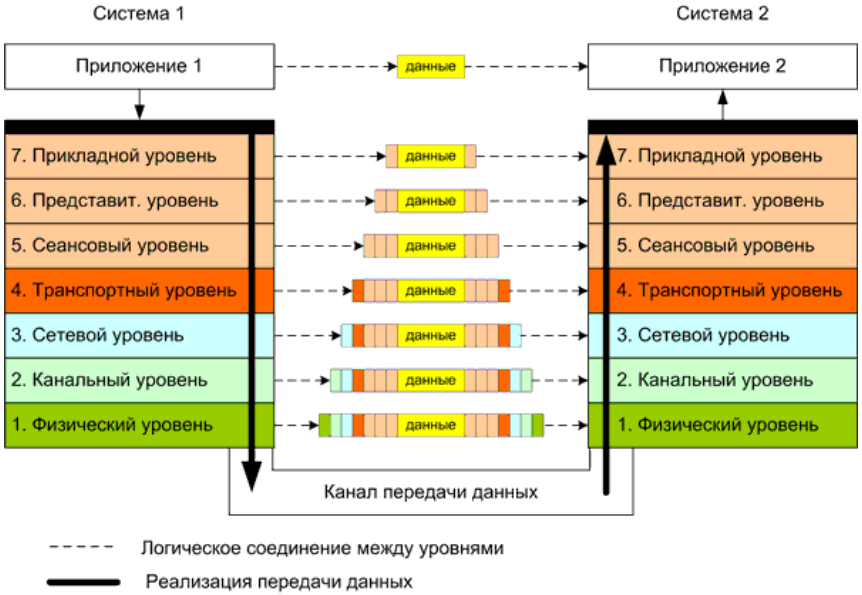


Рис. 1 Передача данных между двумя приложениями по стеку OSI

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).

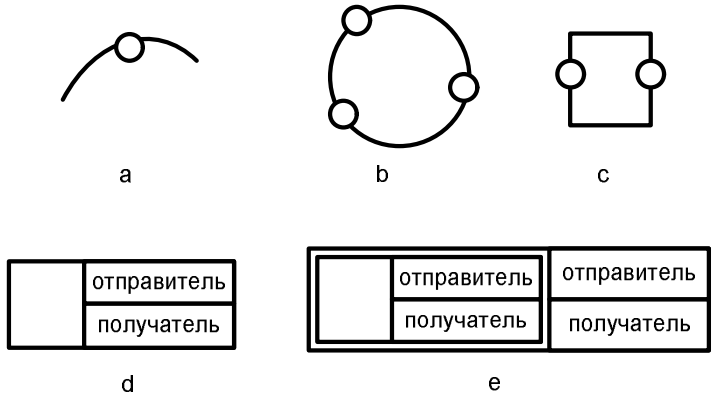


Рисунок 1

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.

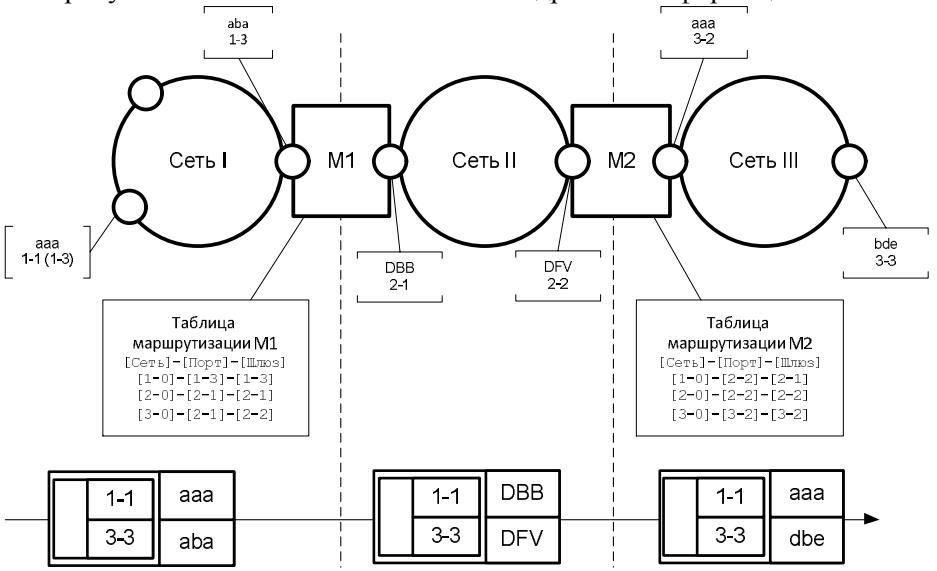


Рисунок 2

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 М1. 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 М2. принимается им и обрабатывается так же как в п.5,6 и 7. С той разницей, что М2 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.

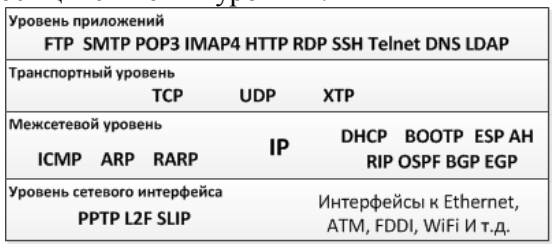


Рисунок 1

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>Время жизни(8 бит)</td><td colspan="2">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.

Порт источника(16 бит)			Порт назначения(16 бит)
Номер последовательности(32 бита)			
Номер подтверждения(32 бита)			
Смещение данных (4 бита)	Зарезервировано (4 бита)	Флаги (4 бита)	Размер окна(16 бит)
Контрольная сумма(16 бит)			Указатель важности(16 бит)
Опции(32 бита)			
Данные			

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.

Порт отправителя(16 бит)	Порт получателя(16 бит)
Длина датаграммы(16 бит)	Контрольная сумма(16 бит)
Данные	

UDP заголовок

В 2009 году Университет стал победителем многоэтапного конкурса, в результате которого определены 12 ведущих университетов России, которым присвоена категория «Национальный исследовательский университет». Министерством образования и науки Российской Федерации была утверждена программа его развития на 2009–2018 годы. В 2011 году Университет получил наименование «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики»

КАФЕДРА ТЕХНОЛОГИЙ ПРОФЕССИОНАЛЬНОГО ОБУЧЕНИЯ

Кафедра технологий профессионального обучения (ТПО) создана на базе учебно-методической секции технологий профессионального обучения кафедры физики (заведующий кафедрой - декан ЕНФ, профессор Н. А. Ярышев). В секцию входили профессор М. И. Потеев (руководитель), а также доценты Н. Н. Горлушкина и Н. Ф. Гусарова. Секция ТПО была преобразована в кафедру технологий профессионального обучения по решению Ученого совета Университета от 28 апреля 1998 года.

С 2003 года кафедра ведет подготовку инженеров специальности 230202 – «Информационные технологии в образовании» со специализацией Управление проектами в информационных образовательных системах.

За первые 10 лет существования кафедры число ее выпускников составило 152, из них 40 получили дипломы с отличием, а 16 работают в Университете.

Инициатор создания кафедры, ее организатор и первый заведующий - декан факультета повышения квалификации преподавателей, профессор М. И. Потеев.

С 2009 года осуществляется подготовка магистров по направлению 230200.68 «Информационные системы».

С 2011 года кафедра перешла на обучение по направлению бакалаврской подготовки 230400.68 «Информационные системы и технологии», а также по новому направлению бакалаврской подготовки 036000.62 «Интеллектуальные системы в гуманитарной сфере».

Артем Дмитриевич Береснев,
Антон Игоревич Говоров,
Антон Владимирович Чунаев

ПРАКТИЧЕСКИЕ РАБОТЫ ПО КУРСУ

«ИНФОРМАЦИОННЫЕ СЕТИ»

Учебное пособие

В авторской редакции
Редакционно-издательский отдел НИУ ИТМО
Зав. РИО Н.Ф. Гусарова
Лицензия ИД 00408 от 05.11.99
Подписано к печати 9.02.12
Заказ 2443
Тираж 120 экз.
Отпечатано на ризографе