

ГОУВПО
ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ И ЗАДАНИЯ
к лабораторным работам по дисциплине
«Организация компьютерных сетей»

(для студентов направления подготовки 09.03.04 “Программная инженерия”)

Донецк-ДонНТУ-2016

ГОУВПО
ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ И ЗАДАНИЯ
к лабораторным работам по дисциплине
«Организация компьютерных сетей»

(для студентов направления подготовки 09.03.04 “Программная инженерия”)

Рассмотрено на заседании кафедры
программной инженерии
Протокол № 1 от 30.08.2016
Утверждено на заседании
учебно-издательского Совета ДонНТУ
протокол № от

Донецк –2016

УДК 681.3

Методические указания и задания к лабораторным работам по дисциплине «Организация компьютерных сетей» для студентов направления подготовки 09.03.04 «Программная инженерия», Сост.: Чернышова А.В., Донецк, ДонНТУ, 2016 - 121 стр.

Приведены методические указания и задания к выполнению лабораторных работ по дисциплине «Организация компьютерных сетей» для студентов направления подготовки 09.03.04 «Программная инженерия». Излагаются вопросы, связанные с проектированием локальной сети и выбором сетевого оборудования, с изучением сетевых настроек и утилит для проверки правильности работы сети. Излагаются вопросы, касающиеся изучения содержимого структур пакетов протоколов стека TCP/IP, использование масок подсети, установки и настройки контроллера доменов, а также проектирования AD.

Методические указания предназначены для усвоения теоретических основ и формирования практических навыков по курсу «Организация компьютерных сетей».

Составители: ст.преп. каф. ПИ Чернышова А.В.

Лабораторная работа №1

Тема: Проектирование локальной сети.

Цель: Научиться создавать проект локальной сети с учетом предлагаемых требований. Обосновать выбор сетевого оборудования.

Методические указания к лабораторной работе

В настоящее время довольно часто бывает необходимым проявить знания и умения выполнения проектов локальной сети. Обычно, для проектирования сети в крупных фирмах и организациях приглашают сотрудников фирм, занимающихся проектированием и монтажом сетей. Если фирма небольшая, то иногда целесообразно проводить проектирование и монтаж сети «своими» силами. Поэтому, рассмотрим основные этапы проектирования локальной сети для небольшой фирмы, состоящей из определенного количества сотрудников, которая занимает определенное количество комнат и этажей.

Основные этапы проектирования локальной сети:

- 1 Определение количества сотрудников, использующих компьютеры.
- 2 Определение планируемого расширения штата фирмы (при проектировании локальной сети необходимо предусмотреть планируемое расширение фирмы, чтобы в дальнейшем была возможность подключения дополнительных узлов к сети).
- 3 Определение количества комнат и этажей, занимаемых фирмой с возможностью дальнейшего расширения.
- 4 Выбор физической топологии сети.
- 5 Выбор оптимального сетевого оборудования (коммутаторов, маршрутизаторов) с учетом планируемого расширения и бюджета фирмы.
- 6 Выбор сетевого кабеля и предварительный подсчет метража в соответствии с метражом комнат.

7 Возможность использования сетевых коробов, пач-панелей, патчкордов, розеток, коммуникационных шкафов для размещения свитчей, управляемых свитчей, маршрутизаторов, серверов, если необходимо ограничить физический доступ к оборудованию сотрудников фирмы.

8 Выбор типа сети – одноранговая сеть, сеть на основе сервера, комбинированная сеть.

9 Определение типов серверов для сети на основе сервера и комбинированной сети (файловый сервер, сервер приложений, сервер-маршрутизатор, почтовый сервер, принт-сервер). Возможность совмещения услуг, предоставляемых серверами (например, можно объединить почтовый сервер и сервер-маршрутизатор, или файловый сервер и принт-сервер).

10 Определить уровень безопасности, необходимый для нормального функционирования фирмы и хранения коммерческой информации, исходя из этого, выбрать, под какой операционной системой будут работать рабочие станции локальной сети и сервера.

11 Выбрав коммуникационное оборудование и дополнительное оборудование для монтажа сети, произвести с учетом текущих цен на сетевое оборудование расчет примерной сметы расходов проекта локальной сети фирмы (прайсы по сетевому оборудованию можно найти на сайтах фирм, например, «Компьютерные технологии»).

Основные рекомендации к выполнению лабораторной работы.

1 При выполнении проектирования локальной сети в соответствии с вариантом заданий для проводной сети рекомендуется:

- при выборе физической топологии использовать «звезду» или иерархическую звезду» (с несколькими коммутаторами);
- для обеспечения возможности фильтрации трафика на канальном уровне и обеспечения дополнительных средств безопасности использовать управляемый коммутатор;

- если предполагается выход в Internet или соединение с другими сетями, использовать маршрутизатор.

- при выборе сетевого кабеля обратить внимание на то, необходим ли экранированный кабель, или достаточно выбрать неэкранированную витую пару;

- кабель рекомендуется выбирать также с учетом того, будет ли использоваться у вас для укладки кабеля сетевые короба, патчпанели, сетевые розетки, или это оборудование не будет использоваться.

- при расчете примерной сметы расходов самостоятельно определить метраж комнат, чтобы в дальнейшем рассчитать метраж сетевого кабеля;

- при расчете сметы расходов на проект локальной сети обратить внимание на конфигурацию серверов и конфигурацию рабочих станций. Объяснить необходимость закупки серверов и рабочих станций выбранной вами конфигурации;

- обосновать выбор операционных систем для компьютеров сотрудников фирмы и серверов.

- обосновать использование коммутационных шкафов в каждой комнате под коммутаторы, сервера; для удобства подключения в напольных шкафах использовать патч-панели. Коммутационные шкафы, патч-панели, сетевые розетки, инструмент для монтажа локальной сети учесть в смете расходов.

Задание к лабораторной работе (часть 1)

Небольшую фирму, состоящую из «А» сотрудников, занимающую «В» этажей в одном здании, размещающуюся в «С» комнатах (количество комнат на этажах выбрать из указанного количества самостоятельно), необходимо обеспечить локальной сетью.

Последнее время увеличился объем работы и в будущем планируется расширение штата (D человек).

У каждого сотрудника есть компьютер. Информация конфиденциальна. Одновременно с установкой сети планируется установка лазерного принтера (выбрать оптимальное количество принтеров для нормальной работы фирмы). Планируется, что будет использоваться сетевая база данных, необходим сервер для хранения информации.

Предложите проект локальной сети для этой фирмы. Необходимо привести примерный план размещения сотрудников по комнатам, перечислить сетевое оборудование, обосновать выбор данного сетевого оборудования, необходимого для нормальной работы сети, описать топологию, которой Вы будете придерживаться, проектируя сеть, обосновать выбор. Описать обязанности сотрудников по отношению к сети (будет ли ими производиться настройка адаптеров и т.д.). Какие меры безопасности Вы бы предложили для сохранения конфиденциальности информации. Посчитать стоимость проекта с учетом выбранного сетевого оборудования.

Варианты лабораторной работы приведены таблице 1.

Таблица 1 – Варианты заданий

| № варианта | «А» сотрудники | «В» этажи | «С» комнаты | «Д» расширение |
|------------|----------------|-----------|-------------|----------------|
| 1 | 10 | 2 | 3 | 5 |
| 2 | 12 | 1 | 4 | 5 |
| 3 | 12 | 2 | 3 | 8 |
| 4 | 10 | 1 | 2 | 5 |
| 5 | 7 | 1 | 2 | 3 |
| 6 | 8 | 1 | 4 | 5 |
| 7 | 9 | 1 | 3 | 7 |
| 8 | 10 | 2 | 2 | 5 |
| 9 | 12 | 2 | 5 | 5 |

| № варианта | «А» сотрудники | «В» этажи | «С» комнаты | «Д» расширение |
|---------------|-------------------|--------------|----------------|-------------------|
| 10 | 12 | 1 | 2 | 8 |
| 11 | 10 | 1 | 4 | 5 |
| 12 | 7 | 1 | 2 | 3 |
| 13 | 8 | 1 | 2 | 5 |
| 14 | 9 | 1 | 2 | 7 |
| 15 | 15 | 2 | 4 | 8 |
| 16 | 15 | 2 | 4 | 10 |
| 17 | 17 | 2 | 4 | 12 |
| 18 | 20 | 3 | 5 | 12 |
| 19 | 20 | 3 | 5 | 10 |
| 20 | 17 | 2 | 3 | 12 |
| 21 | 16 | 1 | 4 | 5 |
| 22 | 16 | 2 | 5 | 6 |
| 23 | 18 | 1 | 4 | 7 |
| 24 | 22 | 2 | 5 | 8 |
| 25 | 22 | 1 | 4 | 9 |
| 26 | 17 | 2 | 3 | 10 |
| 27 | 30 | 2 | 4 | 5 |
| 28 | 31 | 2 | 5 | 5 |
| 29 | 32 | 2 | 4 | 7 |
| 30 | 33 | 1 | 2 | 8 |

Задание к лабораторной работе (часть 2)

Предложите проект локальной сети для этой фирмы, план размещения сотрудников которой приведен на рисунке 1. Необходимо перечислить сетевое оборудование, обосновать выбор данного сетевого оборудования, необходимого для нормальной работы сети, описать

топологию, которой Вы будете придерживаться, проектируя сеть, обосновать выбор. Описать обязанности сотрудников по отношению к сети (будет ли ими производиться настройка адаптеров и т.д.). Какие меры безопасности Вы бы предложили для сохранения конфиденциальности информации.

Исходные данные взять из рисунка 1.

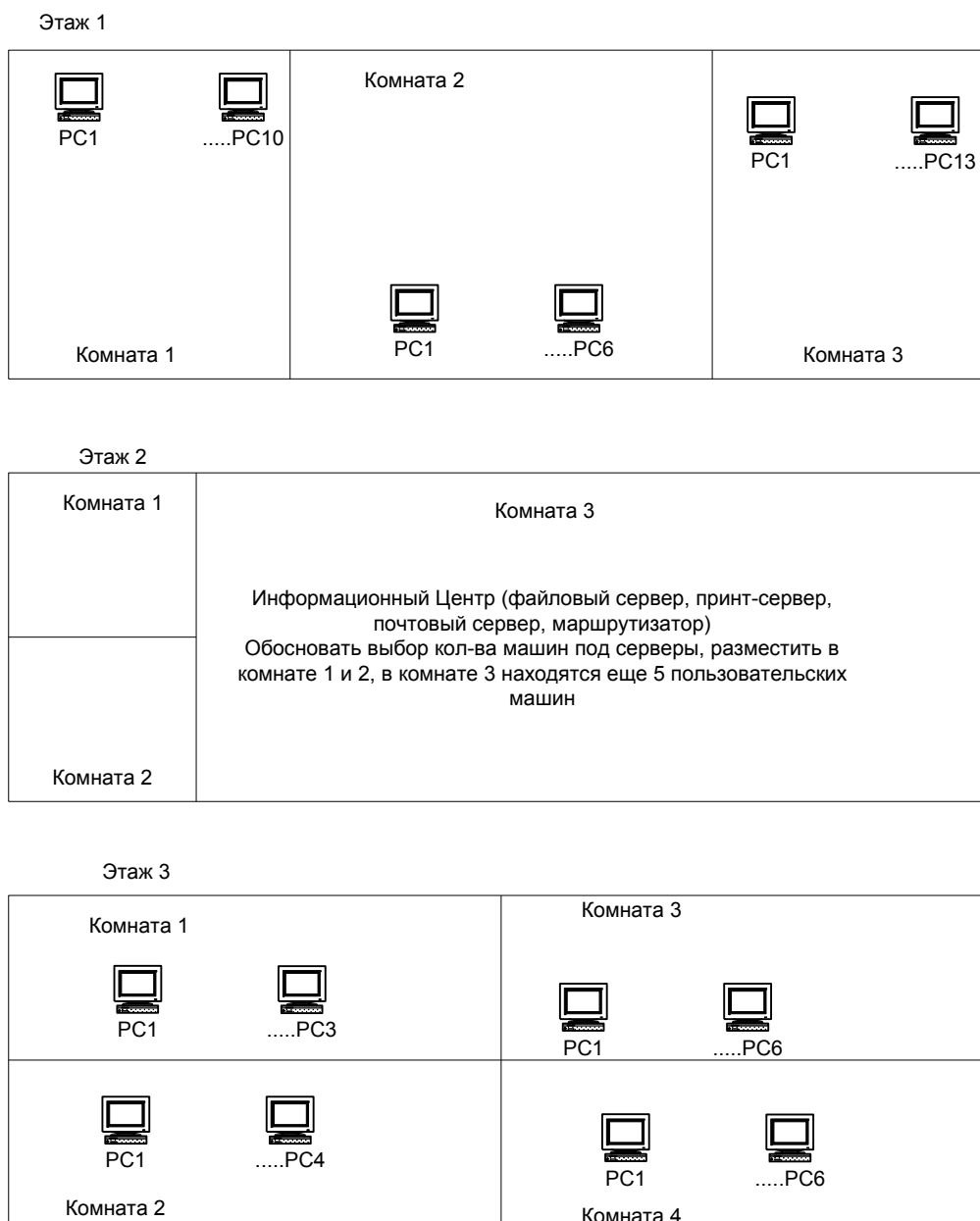


Рисунок 1– План размещения PC для проектирования ЛВС (задача 2 – вариант один для всех)

Требования к отчету по лабораторной работе:

Отчет по лабораторной работе должен содержать:

- схему размещения сотрудников фирмы по отделам (с отделами определиться самостоятельно);
- схему подключения узлов сети к коммутаторам, маршрутизатору (с учетом серверов и рабочих станций);
- описание выбранной Вами типовой конфигурации для серверов, рабочих станций, с указанием выбранной ОС и аппаратуры (тип процессора, память, жесткий диск – использовать готовую конфигурацию, предлагаемую фирмами);
 - перечень сетевого оборудования (коммутаторы, маршрутизаторы, кабель, пассивное сетевое оборудование), его кол-во, цена за единицу и общая стоимость (взять из прайса сетевого оборудования);
 - типы серверов (сервер приложений, файловый сервер, прин-сервер и т.д.);
 - действия сотрудников фирмы по настройке и поддержанию работоспособности локальной сети.
 - какие средства безопасности сети можно использовать?

Контрольные вопросы:

- 1) Что такое сеть на основе сервера?
- 2) Какие физические топологии Вы знаете?
- 3) Какие категории кабеля «витая пара» Вы знаете?
- 4) Какие еще типы кабеля Вы знаете?
- 5) Что такое 8P8C?
- 6) В чем отличие концентратора от коммутатора?
- 7) Для чего используется управляемый коммутатор?
- 8) В чем отличие маршрутизатора от коммутатора?
- 9) От чего зависит, на сколько портов выбрать коммутатор?
- 10) Для чего используются патч-панели?

11) Какие средства защиты сети Вы предложили бы для своего проекта?

Лабораторная работа №2

Тема: Установка и настройка сетевых протоколов. Изучение сетевых настроек ОС Windows

Цель работы: Освоить принципы настройки сетевых параметров ОС Windows.

Методические указания к выполнению работы

Для настройки сети машины, подключенной к локальной сети, необходимо обратиться к «Свойствам» «Сетевого окружения» (рисунок 1)

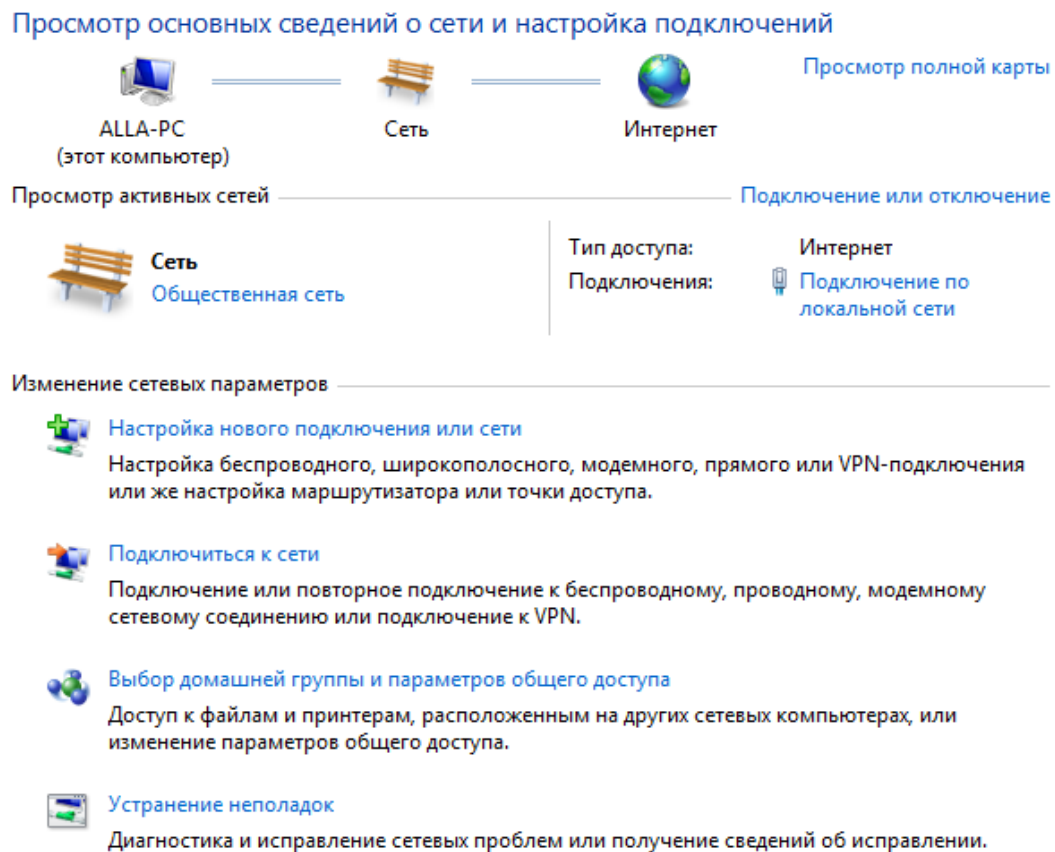


Рисунок 1 – «Свойства сетевого окружения»

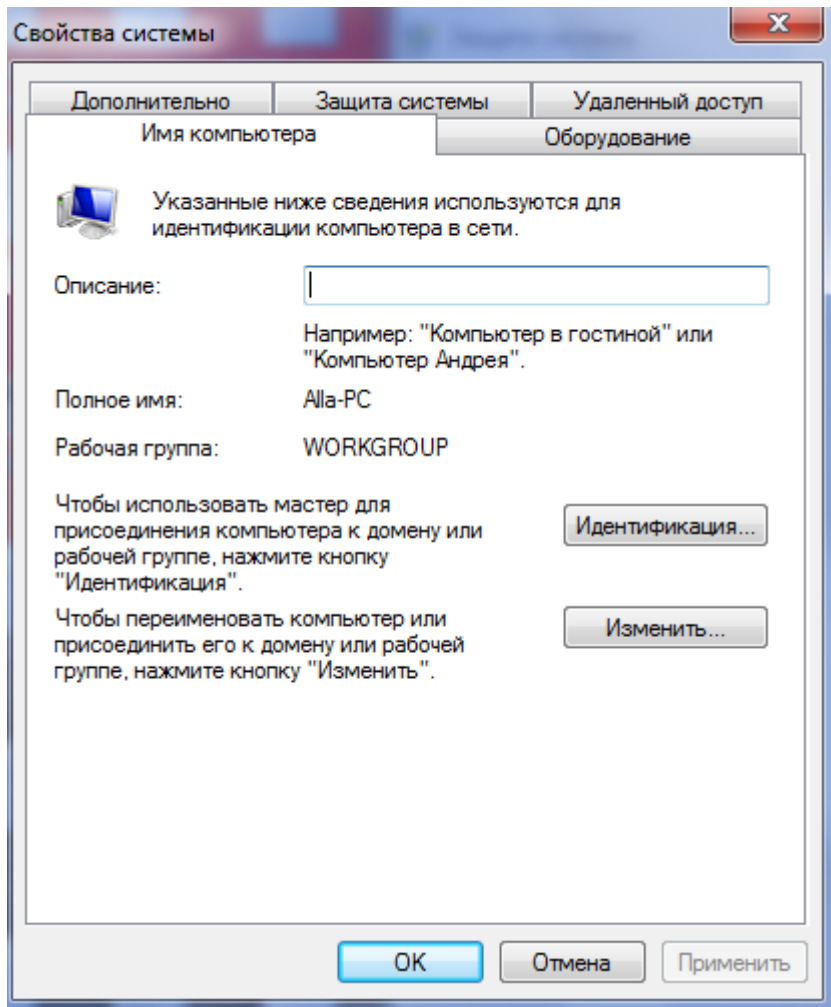


Рисунок 2- «Свойства «компьютер»»

Здесь необходимо указать имя компьютера в сети, к какой рабочей группе или домену принадлежит Ваш компьютер, и заполнить «Описание компьютера» (иногда совпадает с именем компьютера).

Теперь следует обратиться к вкладке «Конфигурации» (Рисунок 3)

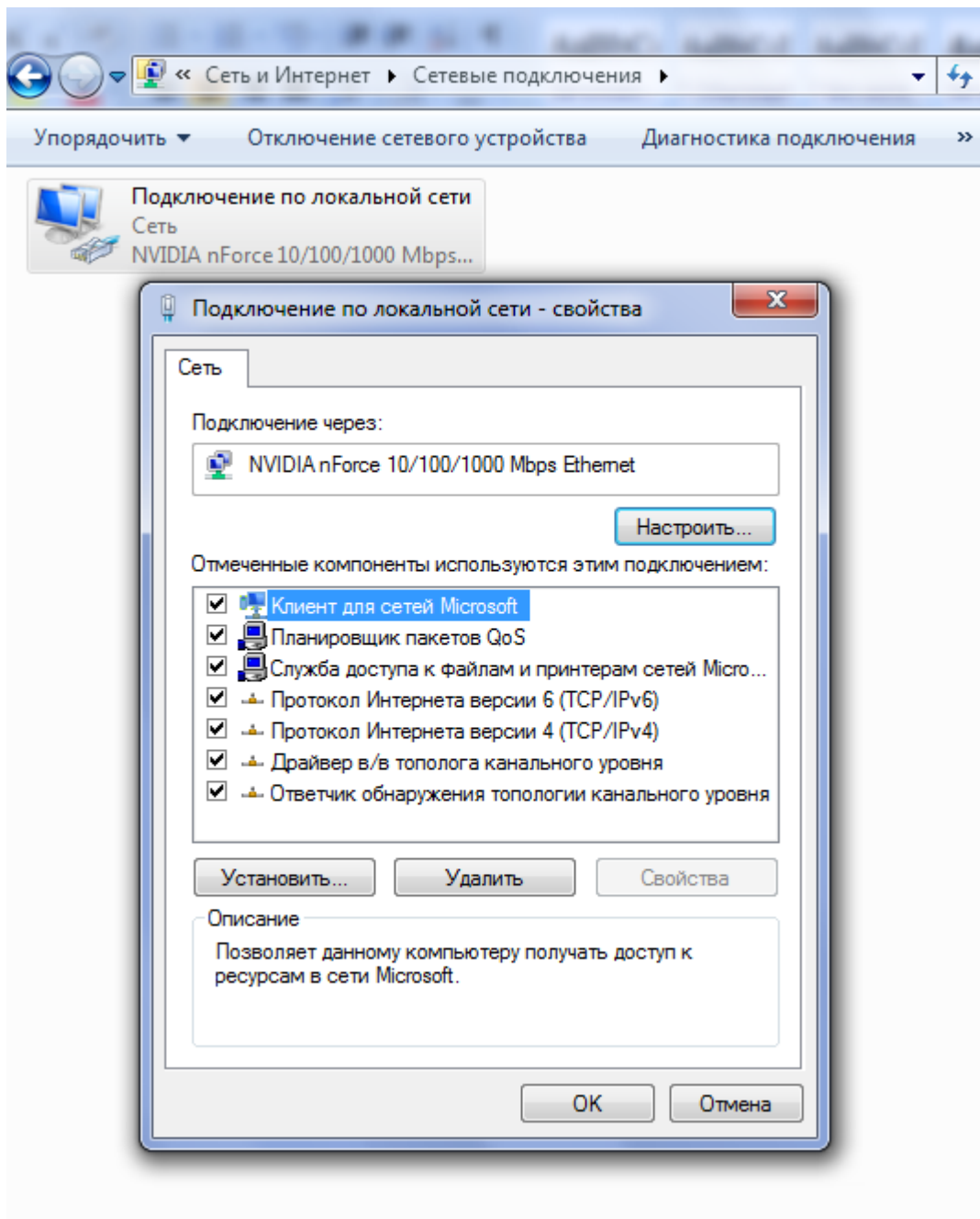
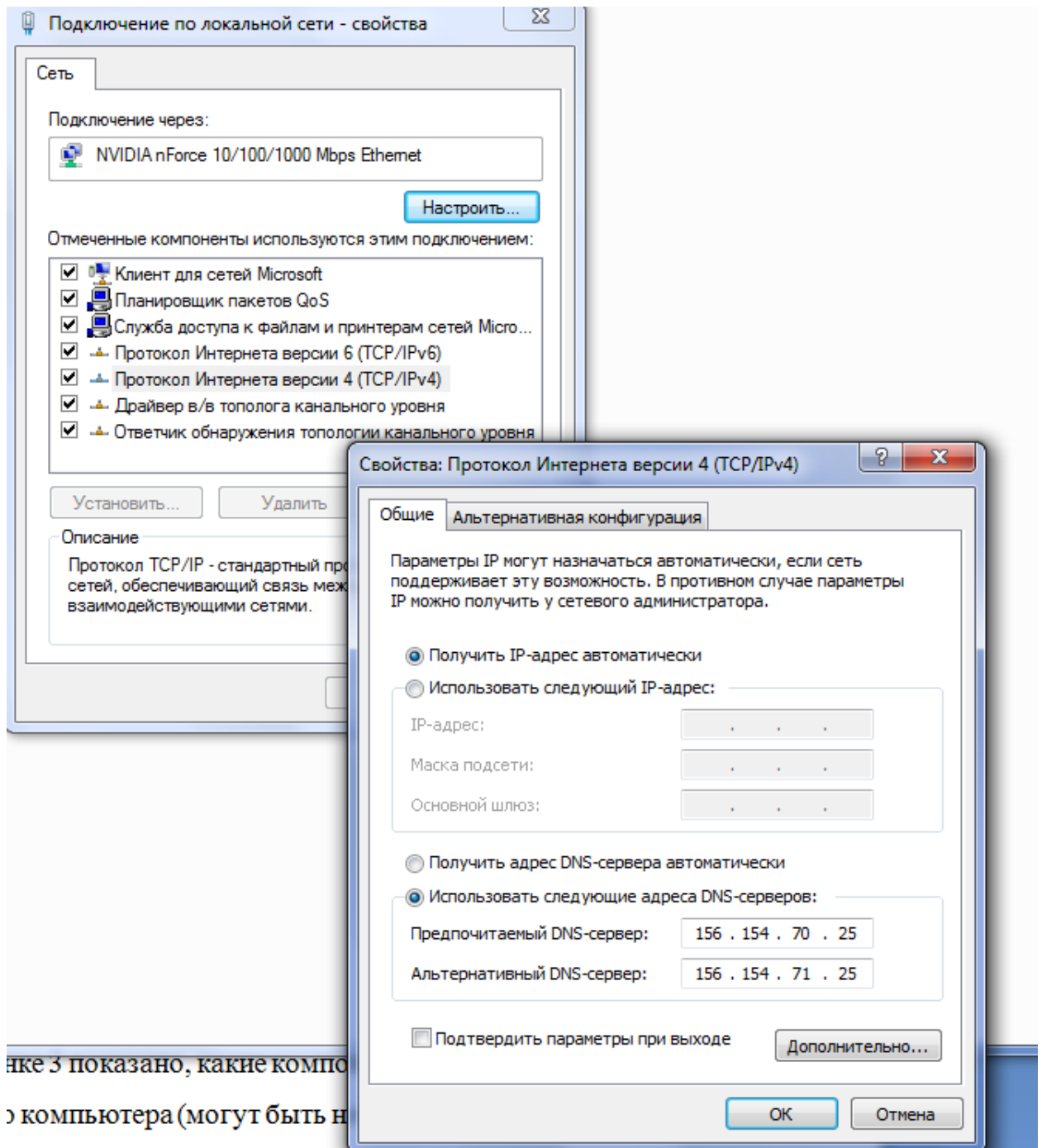


Рисунок 3.а – Просмотр установленных компонентов



Рисунке 3 показано, какие компоненты компьютера (могут быть н

Рисунок 3.б - Просмотр установленных компонентов

Для установления сети на локальном компьютере необходимо установить ряд протоколов и служб.

На рисунке 3 показано, какие компоненты могут быть установлены для определенного компьютера (могут быть некоторые изменения в зависимости от типа сети).

Например, служба доступа к файлам и принтерам устанавливается в том случае, если необходимо организовывать доступ к локальным ресурсам узла для других пользователей или иметь доступ к ресурсам, предоставляемым другими узлами сети.

Способ входа в сеть может быть или «Клиент для сетей Microsoft» или «Обычный вход в Windows». Выбор того или иного способа связан также с особенностями сети. Пользователи, объединенные в группы (например, РМІ) для входа в сеть обычно используют способ входа в сеть - «Клиент для сетей Microsoft». При таком входе при загрузке компьютера предлагается ввести логин и пароль, после чего будут доступны ресурсы сети, разрешенные для использования данной рабочей группы и, непосредственно, вошедшему под определенным логином и паролем пользователю.

«Клиент для сетей Microsoft» обеспечивает связь с другими компьютерами и серверами, работающими в среде Microsoft Windows, а также доступ к общим файлам и принтерам.

Далее следует установить протоколы, необходимые для осуществления доступа в сеть. Чтобы добавить новый протокол необходимо выполнить «Добавить...» и из предложенного списка выбрать протоколы.

С помощью «Добавить», можно также выбрать и другие типы устанавливаемых компонент (служба, клиент, сетевая плата) (рисунок 4)

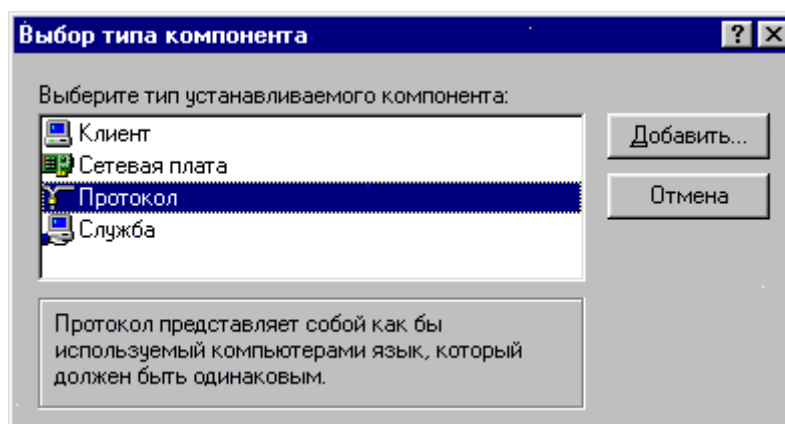


Рисунок 4 – Выбор типа устанавливаемого компонента

Следует особое внимание обратить на настройку «TCP/IP» - стек протоколов, используемый для подключения к Internet.

Настройка TCP/IP включает в себя набор вкладок. На каждой вкладке предложено ввести основные свойства TCP/IP. К таким свойствам относятся IP-адрес, маска подсети, сервер DNS, шлюз, привязка.

Установка IP-Address (рисунок 5). IP- Address конкретного узла

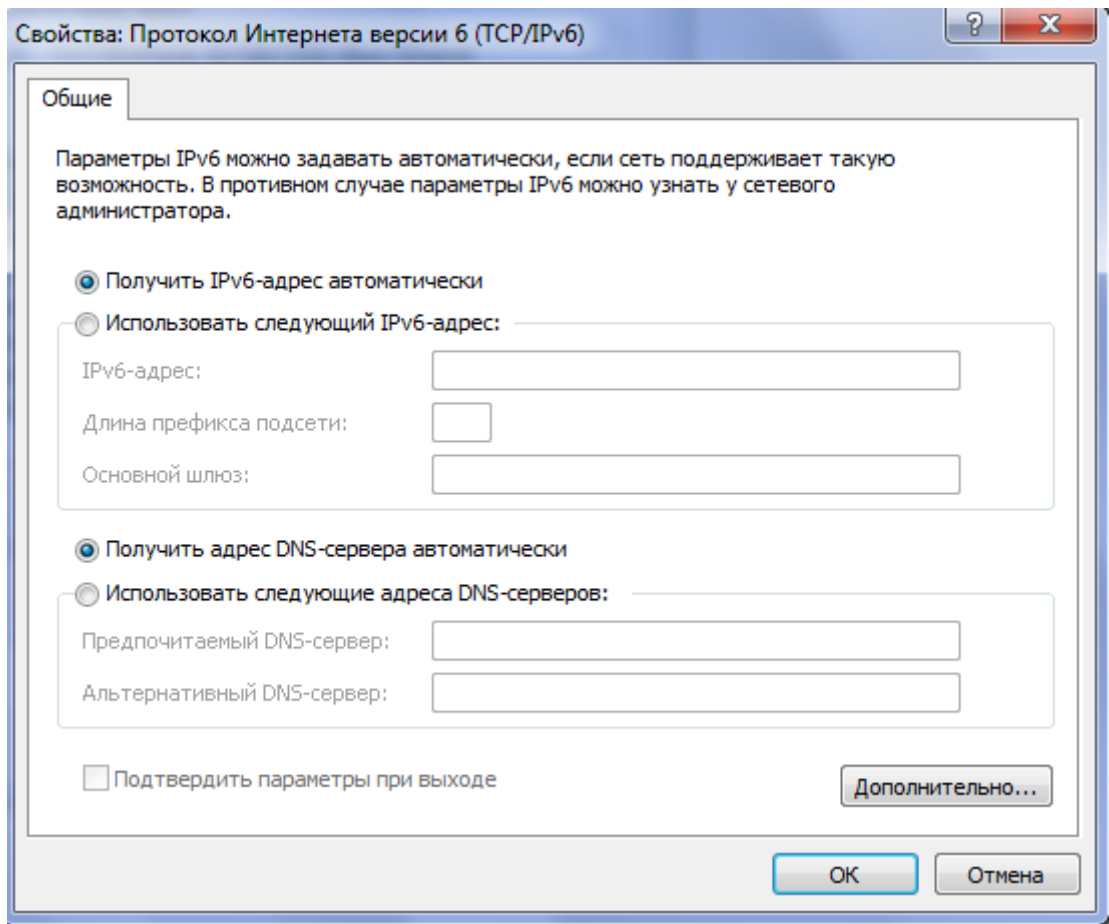


Рисунок 5 «Свойства» Настройка TCP/IP

можно узнать у администратора сети.

Маска подсети может быть различной, значение маски подсети связано с особенностями организации сегментов сети и назначается также администратором сети.

«Gateway» или шлюз – устройство, которое обеспечивает выход в другую сеть, назначается администратором сети.

Сервер DNS осуществляет соответствие между IP-адресами и именами узлов. В DNS прописывается адрес этого сервера.

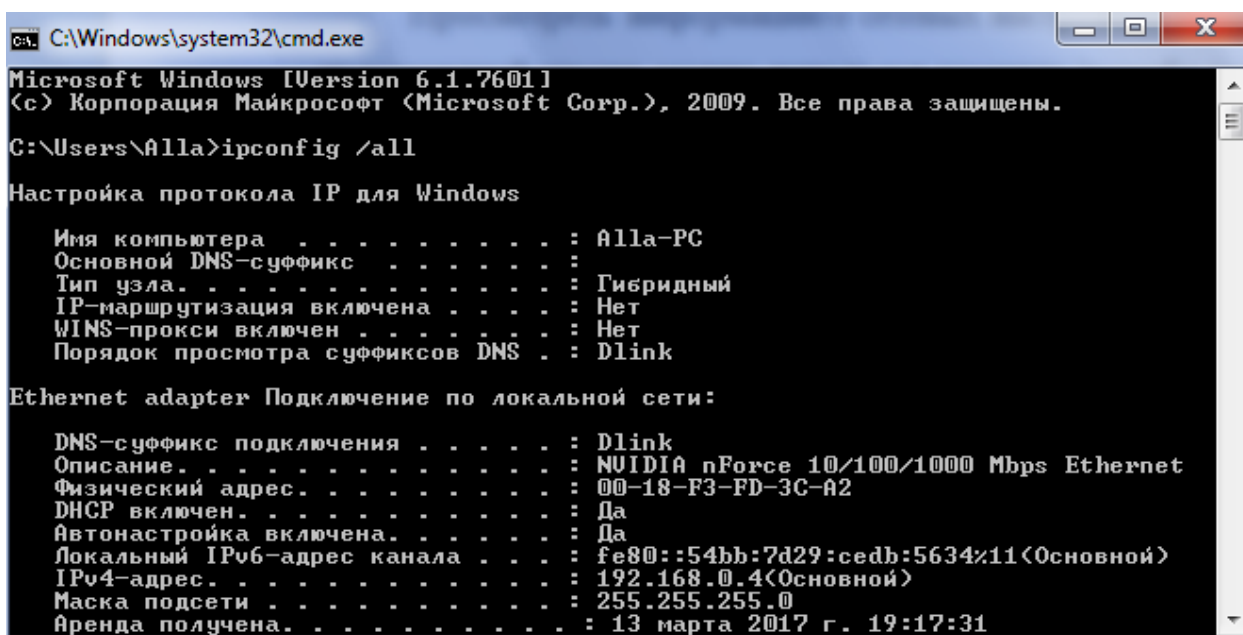
Для конкретной сети маска подсети, Gateway, DNS Server свои. При настройке сети на Вашем компьютере и незнанию вышеперечисленной информации, следует обратиться к системному администратору.

Следует помнить о том, что вся перечисленная выше информация, прописываемая в свойствах TCP/IP, может устанавливаться автоматически,

без непосредственного участия пользователя. Автоматическое назначение IP-адресов, DNS-сервера, шлюза, маски подсети выполняется с помощью DHCP-сервера. DHCP-сервер настраивается в сети, и как только производится включение компьютера, узел посылает DHCP-запрос на получение основных параметров конфигурации, а DHCP – сервер назначает все перечисленные свойства TCP/IP автоматически. При этом значительно упрощается процесс настройки сети на локальном узле.

Одной из особенностей работы DHCP-сервера является то, что IP-адрес узла может назначаться по-разному. Первый вариант, когда IP-адреса выделяются динамически из пула свободных адресов. Вторым вариантом, когда в целях безопасности и разграничения доступа к ресурсам по IP-адресам, IP-адреса назначаются статически, т.е. происходит привязка IP-адреса к MAC-адресу сетевой карты. Если в первом варианте у клиента, подключающегося к сети, каждый раз может быть разный IP-адрес из пула свободных, то во втором случае, каждому клиенту IP-адрес устанавливается жестко на все время.

Просмотреть информацию о сетевых настройках Вашего компьютера из командной строки, можно используя команду ipconfig (winipcfg в старых версиях ОС), см. рисунок 6.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\Alla>ipconfig /all

Настройка протокола IP для Windows

Имя компьютера . . . . . : Alla-PC
Основной DNS-суффикс . . . . . :
Тип узла. . . . . : Гибридный
IP-маршрутизация включена . . . . . : Нет
WINS-прокси включен . . . . . : Нет
Порядок просмотра суффиксов DNS . . . . . : Dlink

Ethernet adapter Подключение по локальной сети:

DNS-суффикс подключения . . . . . : Dlink
Описание. . . . . : NVIDIA nForce 10/100/1000 Mbps Ethernet
Физический адрес. . . . . : 00-18-F3-FD-3C-A2
DHCP включен. . . . . : Да
Автонастройка включена. . . . . : Да
Локальный IPv6-адрес канала . . . . . : fe80::54bb:7d29:cedb:5634%11<Основной>
IPv4-адрес. . . . . : 192.168.0.4<Основной>
Маска подсети . . . . . : 255.255.255.0
Аренда получена. . . . . : 13 марта 2017 г. 19:17:31
```

Рисунок 6 – Пример работы команды ipconfig /all

Следует помнить, что очень часто `ipconfig` используют не только для просмотра сетевых настроек (`ipconfig /all`), но и для обновления параметров сети (`ipconfig /renew`).

Приведенные в методических указаниях настройки (скриншоты) относятся к ОС Windows 7. Существенных отличий в настройках TCP/IP в ОС семейства Windows нет. Есть некоторое различие в визуальном отображении свойств сети, выполняя данную лабораторную работу под ОС более новых версий, пожалуйста, самостоятельно разберитесь с настройками сети и представьте в отчете скриншоты, соответствующие сетевым настройкам Вашего компьютера.

Для быстрого просмотра настроек сети Вашего компьютера в ОС Windows воспользуйтесь командой `ipconfig`, запущенной из командной строки. Вызов командной строки – команда `cmd`.

Информация по команде `ipconfig`:

`ipconfig /?`

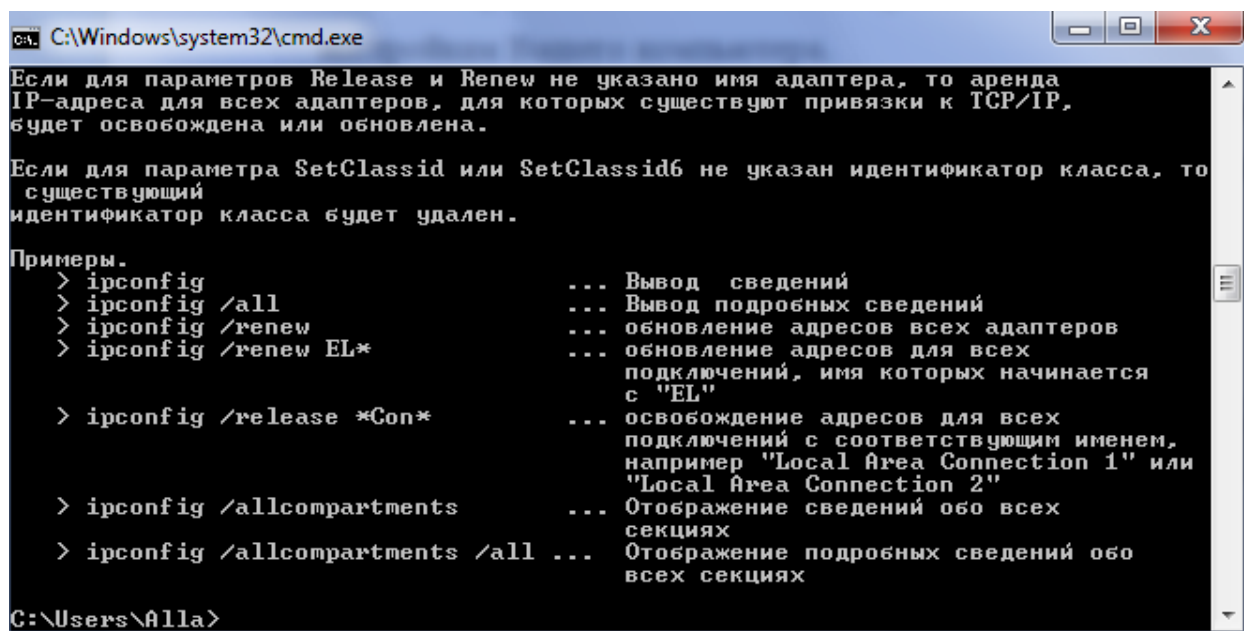


Рисунок 7 – помощь по команде «`ipconfig`»

- `/?` Отобразить это справочное сообщение.
- `/all` Отобразить полную информацию о настройке параметров.
- `/release` Освободить IP-адрес для указанного адаптера.

/renew Обновить IP-адрес для указанного адаптера.

/flushdns Очистить кэш разрешений DNS.

/registerdns Обновить все DHCP-аренды и перерегистрировать DNS-имена

/displaydns Отобразить содержимое кэша разрешений DNS.

/showclassid Отобразить все допустимые для этого адаптера коды (IDs) классов DHCP.

/setclassid Изменить код класса DHCP (ID).

По умолчанию отображается только IP-адрес, маска подсети и стандартный шлюз для каждого подключенного адаптера, для которого выполнена привязка с TCP/IP.

Для ключей /release и /renew, если не указано имя адаптера, то будет освобожден или обновлен IP-адрес, выданный для всех адаптеров, для которых существуют привязки с TCP/IP.

Для ключа SetClassID, если не указан код класса (ID), то существующий код класса будет удален.

Примеры:

- > ipconfig - Отображает краткую информацию.
- > ipconfig /all - Отображает полную информацию.
- > ipconfig /renew - Обновляет сведения для всех адаптеров.
- > ipconfig /renew EL* - Обновляет сведения для адаптеров, начинающихся с EL....

> ipconfig /release *ELINK?21* - Освобождает IP-адреса для всех адаптеров, удовлетворяющих запросу, например, ELINK-21, myELELINKi21adapter.

В разнородной сети (в сети, где используются различные операционные системы) бывает затруднительно настроить локальную сеть таким образом, чтобы ресурсы одного узла были доступны для других узлов. Чтобы избежать подобных проблем, и для быстрого поиска узла по его NetBIOS-имени, можно использовать дополнительные возможности

сетевых настроек, в частности использование файла `lmhosts.sam`. Этот файл содержит таблицу соответствия IP-адресов и обычных (NetBIOS) имен компьютеров. Каждый элемент должен располагаться в отдельной строке. IP-адрес должен начинаться с первой позиции строки, а за ним следует соответствующее имя компьютера. IP-адрес и имя компьютера должны быть отделены друг от друга хотя бы одним пробелом или символом табуляции. Знак "#" используется обычно для указания на начало комментария.

Для быстрого доступа к ресурсам узлов, находящихся в других подсетях, можно прописать соответствия IP-адресов и DNS- именами узлов.

Этот файл называется `hosts` и содержит сопоставления IP-адресов DNS - именам узлов. Каждый элемент должен располагаться в отдельной строке. IP-адрес должен находиться в первом столбце, за ним должно следовать соответствующее имя. IP-адрес и имя узла должны разделяться хотя бы одним пробелом. Кроме того, в некоторых строках могут быть вставлены комментарии, они должны следовать за именем узла и отделяться от него символом '#'.
Например:

```
127.0.0.1    localhost
```

Следует обратить внимание на то, что использование файлов `hosts` и `lmhosts.sam` целесообразно в том случае, если узлы, к которым Вы хотите получить более быстрый доступ, получают один и тот же IP-адрес (статический) при настроенном DHCP-сервере.

Установка дополнительных протоколов зависит от конфигурации сети, необходимость установки тех или иных протоколов можно узнать у сетевого администратора.

Задание к лабораторной работе

В соответствии с изложенным теоретическим материалом, выполнить ряд действий по установке сетевых компонентов. Посмотреть сетевые настройки на локальном компьютере, уметь объяснить использование соответствующих протоколов и их свойств, ответить на контрольные вопросы.

1. Определить количество сетевых подключений, используемых Вашим компьютером (скриншот).
2. Для каждого подключения дать его характеристику, подробно со скриншотами каждого окна и каждой вкладки с комментариями по каждому пункту настройки (назначение, что означает данное значение пункта и т.п.)

Для VPN-подключений:

- общие;
- параметры;
- безопасность;
- сеть;
- дополнительно.

Для подключений по локальной сети (для каждого сетевого адаптера – его тип и перечень свойств):

- общие (для каждого установленного компонента – его свойства подробно);
- дополнительно.

Для соединений удаленного доступа:

- общие;
- параметры;
- безопасность;
- сеть;
- дополнительно;

3. Продемонстрировать создание нового подключения удаленного доступа;
4. Продемонстрировать создание нового подключения к виртуальной частной сети.
5. Продемонстрировать добавление нового протокола, службы или клиента для любого сетевого подключения.
6. Показать, к какой рабочей группе принадлежит компьютер.
7. Продемонстрировать, как и где включается – выключается возможность шаринга ресурсов.
8. Показать, как разрешается доступ к общему ресурсу и как устанавливаются права доступа.

Контрольные вопросы:

1. Какие сетевые протоколы Вы знаете?
2. Какие транспортные протоколы Вы знаете?
3. Объяснить основные настройки TCP/IP.
4. Функции DHCP.
5. Что такое шлюз?
6. Назначение маски подсети?
7. Какие параметры сети могут назначаться сервером DHCP.
8. Назначение файлов hosts и lmhosts.sam.
9. Что такое MAC-адрес.
10. Что позволяет выполнять команда ipconfig?

Лабораторная работа №3

Тема: Изучение сетевых утилит для тестирования и настройки локальной сети в OS Windows и OS Linux.

Цель работы: Ознакомиться с основными командами для проверки наличия и настройки сети в OS Windows и OS Linux.

Методические указания к лабораторной работе:

После того, как выполнены все сетевые настройки, необходимо проверить, есть ли сеть. Это можно сделать с помощью следующих команд:

ping – проверяет соединение с удаленным хостом.

Использование:

```
ping [-t] [-a] [-n число] [-l размер] [-f] [-i TTL] [-v TOS]
      [-r число] [-s число] [[-j списокУзлов] | [-k списокУзлов]]
      [-w интервал] списокРассылки
```

Параметры:

| | |
|-----------|---|
| -t | Отправка пакетов на указанный узел до команды прерывания. |
| -a | Определение адресов по именам узлов. |
| -n число | Число отправляемых запросов. |
| -l размер | Размер буфера отправки. |
| -f | Установка флага, запрещающего фрагментацию пакета. |
| -i TTL | Задание времени жизни пакета (поле "Time To Live"). |
| -v TOS | Задание типа службы (поле "Type Of Service"). |
| -r число | Запись маршрута для указанного числа |

| | |
|----------------|---|
| | переходов. |
| -s число | Штамп времени для указанного числа переходов. |
| -j списокУзлов | Свободный выбор маршрута по списку узлов. |
| -k списокУзлов | Жесткий выбор маршрута по списку узлов. |
| -w интервал | Интервал ожидания каждого ответа в миллисекундах. |

Сервер ДонНТУ имеет IP-адрес 156.154.112.39. Если передачи данных нет, то возможны ошибки в сетевых настройках, либо сервер не отвечает по каким-то причинам. Для проверки команды ping можно в качестве тестируемого адреса выбрать адрес сервера кафедры ПИ – 10.80.128.1 (доступен только из сети университета).

Пример

```

C:\Windows\system32\cmd.exe

Обмен пакетами с donntu.ru [5.153.173.25] с 32 байтами данных:
Ответ от 5.153.173.25: число байт=32 время=2мс TTL=58
Ответ от 5.153.173.25: число байт=32 время=2мс TTL=58
Ответ от 5.153.173.25: число байт=32 время=1мс TTL=58
Ответ от 5.153.173.25: число байт=32 время=1мс TTL=58

Статистика Ping для 5.153.173.25:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (<0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 1мсек, Максимальное = 2 мсек, Среднее = 1 мсек

C:\Users\Alla>ping donntu.org

Обмен пакетами с donntu.org [156.154.112.39] с 32 байтами данных:
Ответ от 156.154.112.39: число байт=32 время=192мс TTL=55
Ответ от 156.154.112.39: число байт=32 время=171мс TTL=55
Ответ от 156.154.112.39: число байт=32 время=178мс TTL=55
Ответ от 156.154.112.39: число байт=32 время=174мс TTL=55

Статистика Ping для 156.154.112.39:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (<0% потерь)
Приблизительное время приема-передачи в мс:

```

Рисунок 1- Пример работы утилиты ping

Для определения участка сети, где прерывается передача данных можно использовать команду: tracert – определяет маршрут, фактически выбранный к узлу назначения.

Определение маршрута.

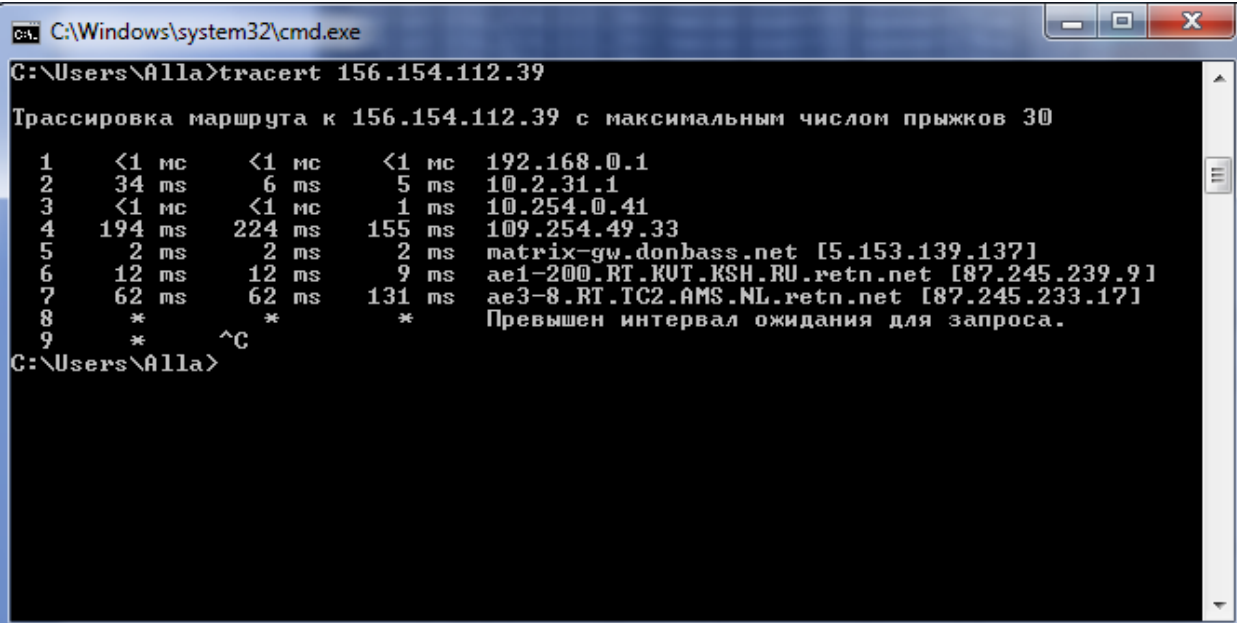
Использование:

tracert [-d] [-h максЧисло] [-j списокУзлов] [-w интервал] имя

Параметры:

| | |
|----------------|---|
| -d | Без определения адресов по именам узлов. |
| -h максЧисло | Максимальное число переходов при поиске узла. |
| -j списокУзлов | Свободный выбор маршрута по списку узлов. |
| -w интервал | Интервал ожидания каждого ответа в миллисекундах. |

Пример



```
cmd.exe C:\Windows\system32\cmd.exe
C:\Users\Alla>tracert 156.154.112.39

Трассировка маршрута к 156.154.112.39 с максимальным числом прыжков 30

 1  <1 мс    <1 мс    <1 мс    192.168.0.1
 2  34 ms    6 ms     5 ms     10.2.31.1
 3  <1 мс    <1 мс    1 ms     10.254.0.41
 4  194 ms   224 ms   155 ms   109.254.49.33
 5  2 ms     2 ms     2 ms     matrix-gw.donbass.net [5.153.139.137]
 6  12 ms    12 ms    9 ms     ae1-200.RT.KUT.KSH.RU.retn.net [87.245.239.9]
 7  62 ms    62 ms    131 ms   ae3-8.RT.TC2.AMS.NL.retn.net [87.245.233.17]
 8  *        *        *
 9  *        *        *
Превышен интервал ожидания для запроса.

C:\Users\Alla>^C
```

Рисунок 2- Пример работы утилиты tracert

Для определения сетевой конфигурации на узле можно использовать команду ipconfig.

```

C:\Windows\system32\cmd.exe

Настройка протокола IP для Windows

Имя компьютера . . . . . : Alla-PC
Основной DNS-суффикс . . . . . :
Тип узла . . . . . : Гибридный
IP-маршрутизация включена . . . . . : Нет
WINS-прокси включен . . . . . : Нет
Порядок просмотра суффиксов DNS . . . . . : Dlink

Ethernet adapter Подключение по локальной сети:

DNS-суффикс подключения . . . . . : Dlink
Описание . . . . . : NVIDIA nForce 10/100/1000 Mbps Ethernet
Физический адрес . . . . . : 00-18-F3-FD-3C-A2
DHCP включен . . . . . : Да
Автонастройка включена . . . . . : Да
Локальный IPv6-адрес канала . . . . . : fe80::54bb:7d29:cedb:5634%11<Основной>
IPv4-адрес . . . . . : 192.168.0.4<Основной>
Маска подсети . . . . . : 255.255.255.0
Аренда получена . . . . . : 13 марта 2017 г. 19:17:31
Срок аренды истекает . . . . . : 14 марта 2017 г. 19:17:31
Основной шлюз . . . . . : 192.168.0.1
DHCP-сервер . . . . . : 192.168.0.1
IAID DHCPv6 . . . . . : 234887411

```

Рисунок 3 - Определение конфигурации IP с помощью утилиты ipconfig

В Windows есть еще несколько полезных команд:

ARP

При передаче данных в локальных сетях (передача кадров канального уровня) необходимо знать MAC-адрес (физический адрес) сетевого устройства. Протокол, который устанавливает соответствие между IP-адресом и MAC-адресом, называется ARP.

Отображение и изменение используемой протоколом ARP таблицы соответствия адресов IP и физических адресов, выполняет команда ARP.

ARP -s inet_addr eth_addr [if_addr]

ARP -d inet_addr [if_addr]

ARP -a [inet_addr] [-N if_addr]

| | |
|----|--|
| -a | Вывод текущих записей таблицы ARP путем опроса текущих данных протокола. Если указан адрес inet_addr, то адреса IP и физические выводятся только для указанного компьютера. Если протокол ARP используется несколькими сетевыми интерфейсами, то выводятся записи из каждой таблицы ARP. |
| -g | Аналог -a. |

| | |
|------------|---|
| inet_addr | Задание адреса IP. |
| -N if_addr | Вывод текущих записей таблицы ARP для сетевого интерфейса, определяемого параметром if_addr. |
| -d | Удаление узла, определяемого параметром inet_addr. |
| -s | Добавление узла и связывание адреса IP inet_addr с физическим адресом eth_addr. Физический адрес задается с помощью 6 шестнадцатеричных чисел, разделяемых дефисами. Запись является постоянной. |
| eth_addr | Задание физического адреса. |
| if_addr | Необязательный параметр, указывающий адрес IP интерфейса, для которого следует изменить таблицу адресов. Если параметр не задан, используется первый доступный интерфейс. |

Пример работы команды `arp -a`.

```

C:\WINDOWS\system32\cmd.exe
where
  adapter          Connection name
                   (wildcard characters * and ? allowed, see examples)

Options:
  /?              Display this help message
  /all            Display full configuration information.
  /release        Release the IP address for the specified adapter.
  /renew          Renew the IP address for the specified adapter.
  /flushdns       Purges the DNS Resolver cache.
  /registerdns    Refreshes all DHCP leases and re-registers DNS names
  /displaydns     Display the contents of the DNS Resolver Cache.
  /showclassid   Displays all the dhcp class IDs allowed for adapter.
  /setclassid    Modifies the dhcp class id.

The default is to display only the IP address, subnet mask and
default gateway for each adapter bound to TCP/IP.

For Release and Renew, if no adapter name is specified, then the IP address
leases for all adapters bound to TCP/IP will be released or renewed.

For Setclassid, if no ClassId is specified, then the ClassId is removed.

Examples:
  > ipconfig          ... Show information.
  > ipconfig /all     ... Show detailed information
  > ipconfig /renew   ... renew all adapters
  > ipconfig /renew EL* ... renew any connection that has its
                        name starting with EL
  > ipconfig /release *Con* ... release all matching connections,
                        eg. "Local Area Connection 1" or
                        "Local Area Connection 2"

C:\Documents and Settings\Administrator.ALLA>arp -a

Interface: 192.168.0.4 --- 0x2
  Internet Address      Physical Address      Type
  192.168.0.1           cc-b2-55-5d-68-18    dynamic

C:\Documents and Settings\Administrator.ALLA>

```

Рисунок 4 - Просмотр таблицы arp

ROUTE - Обработка таблиц сетевых маршрутов.

ROUTE [-f] [команда [узел] [MASK маска] [шлюз] [METRIC метрика]]

| | |
|---------|---|
| -f | Очистка таблиц маршрутов от записей для всех шлюзов. При указании одной из команд, таблицы очищаются до выполнения команды. |
| команда | Одна из четырех команд PRINT Печать маршрута ADD Добавление маршрута DELETE Удаление маршрута CHANGE Изменение существующего маршрута |
| узел | Адресуемый узел. |

| | |
|--------|--|
| MASK | Если вводится ключевое слово MASK, то следующий параметр интерпретируется как параметр "маска". |
| маска | Значение маски подсети, связываемое с записью для данного маршрута. Если этот параметр не задан, по умолчанию подразумевается 255.255.255.255. |
| шлюз | Шлюз. |
| METRIC | Определение параметра метрика/цена для адресуемого узла. |

Поиск всех символических имен узлов проводится в файле сетевой базы данных NETWORKS. Поиск символических имен шлюза проводится в файле базы данных имен узлов HOSTS.

Для команд PRINT и DELETE можно указать узел и шлюз с помощью подстановочных знаков или опустить параметр "шлюз".

Сведения диагностики:

неправильное значение MASK приводит к ошибке,
(DEST & MASK) != DEST

Например

```
>route ADD 157.0.0.0 MASK 155.0.0.0 157.55.80.1
Сбой добавления маршрута: 87
```

Примеры.

```
>route PRINT
>route ADD 157.0.0.0 MASK 255.0.0.0 157.55.80.1 METRIC 3
           ^назначение   ^маска      ^шлюз      метрика^
>route PRINT
>route DELETE 157.0.0.0
>route PRINT
```

Пример работы команды route print

```

C:\Windows\system32\cmd.exe

C:\Users\Alla>route print
=====
Список интерфейсов
11...00 18 f3 fd 3c a2 .....NVIDIA nForce 10/100/1000 Mbps Ethernet
1 .....Software Loopback Interface 1
12...00 00 00 00 00 00 e0 .....Адаптер Microsoft ISATAP
13...00 00 00 00 00 00 e0 .....Teredo Tunneling Pseudo-Interface
=====

IPv4 таблица маршрута
=====
Активные маршруты:
Сетевой адрес          Маска сети            Адрес шлюза           Интерфейс             Метрика
0.0.0.0                0.0.0.0              192.168.0.1          192.168.0.4           20
127.0.0.0              255.0.0.0            On-link              127.0.0.1             306
127.0.0.1              255.255.255.255     On-link              127.0.0.1             306
127.255.255.255       255.255.255.255     On-link              127.0.0.1             306
192.168.0.0            255.255.255.0       On-link              192.168.0.4           276
192.168.0.4            255.255.255.255     On-link              192.168.0.4           276
192.168.0.255         255.255.255.255     On-link              192.168.0.4           276
224.0.0.0              240.0.0.0            On-link              127.0.0.1             306
224.0.0.0              240.0.0.0            On-link              192.168.0.4           276
255.255.255.255       255.255.255.255     On-link              127.0.0.1             306
255.255.255.255       255.255.255.255     On-link              192.168.0.4           276
=====

Постоянные маршруты:
Отсутствует

IPv6 таблица маршрута
=====
Активные маршруты:
Метрика   Сетевой адрес          Шлюз
13        ::/0                   On-link
1         306  ::1/128                On-link
13        58  2001::/32                On-link
13        306  2001:0:9d38:78cf:10c3:3ce6:9201:ce38/128
On-link
11        276  fe80::/64                On-link
13        306  fe80::/64                On-link
12        281  fe80::5efe:192.168.0.4/128
On-link
13        306  fe80::10c3:3ce6:9201:ce38/128
On-link
11        276  fe80::54bb:7d29:cedb:5634/128
On-link
1         306  ff00::/8                 On-link
13        306  ff00::/8                 On-link
11        276  ff00::/8                 On-link
=====

Постоянные маршруты:
Отсутствует

C:\Users\Alla>

```

Рисунок 4 - Пример работы утилиты route print

NETSTAT - Отображение статистики протокола и текущих сетевых подключений TCP/IP.

Использование:

netstat [-a] [-e] [-n] [-s] [-p имя] [-r] [интервал]

Параметры:

| | |
|----|--|
| -a | Отображение всех подключений и ожидающих портов. (Подключения со стороны сервера обычно не |
|----|--|

| | |
|----------|--|
| | отображаются). |
| -e | Отображение статистики Ethernet. Этот ключ может применяться совместно с ключом -s. |
| -n | Отображение адресов и номеров портов в числовом формате. |
| -р имя | Отображение подключений для протокола "имя": tcp или udp. Используется вместе с ключом -s для отображения статистики по протоколам. Допустимые значения "имя": tcp, udp или ip. |
| -r | Отображение содержимого таблицы маршрутов. |
| -s | Отображение статистики по протоколам. По умолчанию выводятся данные для TCP, UDP и IP. Ключ -р позволяет указать подмножество выводящихся данных. |
| интервал | Повторный вывод статистических данных через указанный интервал в секундах. Для прекращения вывода данных нажмите клавиши CTRL+C. Если параметр не задан, сведения о текущей конфигурации выводятся один раз. |

Пример использования:

1) netstat

Активные подключения

| Имя | Локальный адрес | Внешний адрес | Состояние |
|-----|-----------------|----------------------|-----------|
| TCP | MAG:1208 | 208.184.172.140:1975 | SYN_SENT |

2) netstat -a

Активные подключения

| Имя | Локальный адрес | Внешний адрес | Состояние |
|-----|-----------------|----------------------|-----------|
| TCP | MAG:1260 | MAG:0 | LISTENING |
| TCP | MAG:137 | MAG:0 | LISTENING |
| TCP | MAG:138 | MAG:0 | LISTENING |
| TCP | MAG:nbsession | MAG:0 | LISTENING |
| TCP | MAG:1260 | 208.184.172.140:1975 | SYN_SENT |
| UDP | MAG:nbname | *:* | |
| UDP | MAG:nbdatagram | *:* | |

3) netstat -e

Статистика интерфейса

| | Получено | Отправлено |
|----------------------|----------|------------|
| Байт | 595285 | 99343 |
| Одноадресные пакеты | 1322 | 1324 |
| Многоадресные пакеты | 1217 | 76 |
| Отброшено | 0 | 0 |
| Ошибки | 0 | 0 |
| Неизвестный протокол | 2132 | |

4) netstat -n

Активные подключения

| Имя | Локальный адрес | Внешний адрес | Состояние |
|-----|---------------------|----------------------|-----------|
| TCP | 192.168.33.160:1277 | 192.168.33.189:139 | TIME_WAIT |
| TCP | 192.168.33.160:1279 | 208.184.172.140:1975 | SYN_SENT |

5) netstat -r

Активные маршруты:

| Сетевой адрес | Маска | Адрес шлюза | Интерфейс | Метрика |
|-----------------|-----------------|----------------|----------------|---------|
| 0.0.0.0 | 0.0.0.0 | 192.168.33.62 | 192.168.33.160 | 1 |
| 127.0.0.0 | 255.0.0.0 | 127.0.0.1 | 127.0.0.1 | 1 |
| 192.168.33.128 | 255.255.255.192 | 192.168.33.160 | 192.168.33.160 | 1 |
| 192.168.33.160 | 255.255.255.255 | 127.0.0.1 | 127.0.0.1 | 1 |
| 192.168.33.255 | 255.255.255.255 | 192.168.33.160 | 192.168.33.160 | 1 |
| 224.0.0.0 | 224.0.0.0 | 192.168.33.160 | 192.168.33.160 | 1 |
| 255.255.255.255 | 255.255.255.255 | 192.168.33.160 | 192.168.33.160 | 1 |

Таблица маршрутов

Активные подключения

| Имя | Локальный адрес | Внешний адрес | Состояние |
|-----|-----------------|-------------------------|-----------|
| TCP | MAG:1282 | 208.184.172.140:1975 | SYN_SENT |
| TCP | MAG:1277 | PMI_NT_SERVER:nbsession | TIME_WAIT |

б) netstat -s

Статистика IP

| | |
|-----------------------------------|--------|
| Получено пакетов | = 2456 |
| Получено ошибок в заголовках | = 0 |
| Получено ошибок в адресах | = 18 |
| Направлено датаграмм | = 0 |
| Получено неизвестных протоколов | = 0 |
| Отброшено полученных пакетов | = 0 |
| Доставлено полученных пакетов | = 2456 |
| Запросов на вывод | = 1286 |
| Отброшено маршрутов | = 0 |
| Отброшено выходных пакетов | = 0 |
| Выходных пакетов без маршрута | = 0 |
| Требуется сборка | = 0 |
| Успешная сборка | = 0 |
| Сбоев при сборке | = 0 |
| Успешно фрагментировано датаграмм | = 0 |
| Сбоев при фрагментации датаграмм | = 0 |
| Создано фрагментов | = 0 |

Статистика ICMP

| | Получено | Оправлено |
|------------------------|----------|-----------|
| Сообщения | 930 | 9 |
| Ошибки | 0 | 0 |
| Не доставлено | 912 | 6 |
| Превышение времени | 0 | 0 |
| Неверные параметры | 0 | 0 |
| Отключение источника | 0 | 0 |
| Переадресовано | 0 | 0 |
| Проверочные пакеты | 0 | 0 |
| Ответные пакеты | 0 | 0 |
| Штампы времени | 0 | 0 |
| Штампы времени ответов | 0 | 0 |
| Маски адресов | 0 | 0 |
| Маски адресов ответов | 0 | 0 |

Статистика TCP

| | |
|-----------------------|-------|
| Активно открыто | = 246 |
| Пассивно открыто | = 2 |
| Сбоев при подключении | = 224 |
| Сброшено подключений | = 1 |
| Текущих подключений | = 0 |
| Получено сегментов | = 262 |

Отправлено сегментов = 500
Переотправлено сегментов = 675

Статистика UDP

Получено датаграмм = 916
Отсутствие портов = 1252
Ошибки при получении = 26
Отправлено датаграмм = 102

NET

Чтобы получить дополнительные сведения о конкретной команде Microsoft NET, поместите вслед за именем команды ключ /? (например NET VIEW /?).

| | |
|--------------|---|
| NET CONFIG | Вывод сведений о рабочей группе. |
| NET DIAG | Запуск программы Microsoft Network Diagnostics для получения данных о сети. |
| NET HELP | Вывод сведений о командах и сообщениях об ошибках. |
| NET INIT | Загрузка протокола и драйверов сетевой платы без привязки их к диспетчеру протоколов. |
| NET LOGOFF | Отключение всех используемых компьютером общих ресурсов. |
| NET LOGON | Идентификация пользователя как члена рабочей группы. |
| NET PASSWORD | Изменение пароля для входа в сеть. |
| NET PRINT | Вывод сведений об очередях печати и управление заданиями по выводу на печать. |
| NET START | Запуск служб. |
| NET STOP | Остановка работы служб. |
| NET TIME | Вывод времени с другого компьютера или синхронизация часов с часами на сервере времени Microsoft Windows для рабочих групп, Windows NT, Windows 95 или NetWare. |
| NET USE | Подключение и отключение сетевых ресурсов и вывод сведений о подключениях. |
| NET VER | Вывод типа и версии используемой системы переадресации. Вывод типа и версии используемой системы переадресации. |
| NET VIEW | Вывод списка компьютеров, обеспечивающих совместный доступ к ресурсам, или общих ресурсов конкретного компьютера. |

Вывод текущих параметров рабочей группы.
NET CONFIG [/YES]

| | |
|------|--|
| /YES | Выполнение команды NET CONFIG без предварительного запроса данных или подтверждения. |
|------|--|

Пример

net config

| | |
|----------------|------------|
| Компьютер | \\MAG |
| Пользователь | PREPOD |
| Рабочая группа | PMI |
| Корневая папка | C:\WINDOWS |

Версия программы 4.10.1998
Версия системы переадресации 4.00
Команда выполнена успешно.

Запуск программы Microsoft Network Diagnostics для проверки аппаратного соединения между компьютерами и вывода сведений о компьютере.

NET DIAGNOSTICS [/NAMES | /STATUS]

| | |
|---------|---|
| /NAMES | Имя сервера диагностики, необходимое для устранения конфликтов при использовании NET DIAG одновременно несколькими пользователями. Этот параметр применим лишь при использовании протокола NetBIOS. |
| /STATUS | Компьютер, о котором следует получить сведения. |

Вывод сведений о командах и сообщениях NET.

команда /?

NET HELP [суффикс]

NET HELP код_ошибки

| | |
|---------|--|
| команда | Определяет команду Microsoft NET, сведения о которой следует получить. |
|---------|--|

| | |
|------------|--|
| суффикс | Определяет второе слово интересующей команды. Например, суффикс команды NET VIEW - слово VIEW. |
| код_ошибки | Задаёт номер интересующего сообщения об ошибке. |

Чтобы получить краткое описание всех команд Microsoft NET, введите команду NET HELP без параметров.

Загрузка протокола и драйверов сетевой платы без их привязки к диспетчеру протоколов.

Эта команда может понадобиться при использовании драйверов сторонних поставщиков. Привязка драйверов в данном случае производится командой NET START NETBIND.

NET INITIALIZE [/DYNAMIC]

| | |
|----------|---|
| /DYNAMIC | Динамическая загрузка диспетчера протоколов. Она удобна при работе с сетями сторонних поставщиков, например, Banyan(R) VINES(R), и служит для устранения неполадок с памятью. |
|----------|---|

Разрыв связи между компьютером и общими ресурсами, к которым он подключен.

NET LOGOFF [/YES]

| | |
|------|--|
| /YES | Выполнение команды NET LOGOFF без предварительного запроса данных или подтверждения. |
|------|--|

Идентификация пользователя как члена рабочей группы.

NET LOGON [имя [пароль | ?]] [/DOMAIN:домен] [/YES]

[/SAVEPW:NO]

| | |
|---------|--|
| имя | Имя, идентифицирующее пользователя в рабочей группе. Оно может содержать не более 20 символов. |
| пароль | Уникальная строка символов, обеспечивающая доступ к файлу со списком паролей. Пароль может содержать не более 14 символов. |
| ? | Означает необходимость выдачи запроса на ввод пароля. |
| /DOMAIN | Определяет необходимость подключения к домену Microsoft Windows NT или LAN Manager. |
| домен | Имя домена Windows NT или LAN Manager. |

| | |
|------------|---|
| /YES | Выполнение команды NET LOGON без предварительного запроса данных или подтверждения. |
| /SAVEPW:NO | Выполнение команды NET LOGON без предварительного запроса на создание файла со списком паролей. |

Если предпочтительным является интерактивный ввод пароля и имени пользователя, воспользуйтесь командой NET LOGON без параметров.

Изменение пароля для входа в сеть.

NET PASSWORD [старый [новый]]

NET PASSWORD \\компьютер | /DOMAIN:домен
[имя [старый [новый]]]

| | |
|-----------|--|
| старый | Текущий пароль. |
| новый | Новый пароль. Его длина не должна превышать 14 символов. |
| компьютер | Имя сервера Windows NT или LAN Manager, на котором необходимо произвести смену пароля. |
| /DOMAIN | Означает, что смену пароля следует произвести в домене Windows NT или LAN Manager. |
| домен | Имя домена Windows NT или LAN Manager. |
| имя | Имя пользователя в сети Windows NT или LAN Manager. |

Первый вариант команды предназначен для смены пароля файла со списком паролей. Ее второй вариант обеспечивает смену пароля пользователя непосредственно на сервере или в домене Windows NT либо LAN Manager.

Вывод сведений об очереди печати на общем принтере и управление заданиями.

NET PRINT \\компьютер[\принтер] | порт [/YES]

NET PRINT \\компьютер | порт [задание# [/PAUSE | /RESUME |
/DELETE]] [/YES]

| | |
|-----------|--|
| компьютер | Имя компьютера, сведения об очереди печати которого следует получить. |
| принтер | Имя принтера, сведения о котором следует получить. |
| порт | Имя параллельного порта (LPT), назначенного интересующему принтеру. |
| задание# | Номер, присвоенный поставленному в очередь заданию по выводу на печать. Допустимые параметры: /PAUSE Приостановка задания. /RESUME Продолжение вывода остановленного задания. /DELETE Удаление задания. |
| /YES | Выполнение команды NET PRINT без предварительного запроса данных или подтверждения. |

Если в команде NET PRINT задано имя компьютера, на экран выводятся сведения обо всех очередях печати общих принтеров, присоединенных к этому компьютеру.

Запуск служб.

Службы нельзя запускать из сеанса MS-DOS в Windows.

NET START [BASIC | NWREDIR | WORKSTATION | NETBIND |
NETBEUI | NWLINK] [/LIST] [/YES] [/VERBOSE]

| | |
|-------------|---|
| BASIC | Запуск базовой системы переадресации. |
| NWREDIR | Запуск системы переадресации Microsoft, совместимой с Novell(R). |
| WORKSTATION | Запуск стандартной системы переадресации. |
| NETBIND | Привязка протоколов к драйверам сетевых плат. |
| NETBEUI | Запуск интерфейса NetBIOS. |
| NWLINK | Запуск IPX/SPX-совместимого интерфейса. |
| /LIST | Вывод списка запущенных служб. |
| /YES | Выполнение команды NET START без предварительного запроса данных или подтверждения. |
| /VERBOSE | Вывод сведений о драйверах устройств и службах по мере загрузки. |

Для запуска системы переадресации, выбранной при установке, используется команда NET START без параметров. В большинстве случаев использование параметров не потребуется.

Остановка работы служб.

Службы нельзя останавливать из сеанса MS-DOS в Windows.

NET STOP [BASIC | NWREDIR | WORKSTATION | NETBEUI |
NWLINK] [/YES]

| | |
|-------------|--|
| BASIC | Остановка базовой системы переадресации. |
| NWREDIR | Остановка системы переадресации Microsoft, совместимой с Novell(R). |
| WORKSTATION | Остановка стандартной системы переадресации. |
| NETBEUI | Остановка интерфейса NetBIOS. |
| NWLINK | Остановка IPX/SPX-совместимого интерфейса. |
| /YES | Выполнение команды NET STOP без предварительного запроса данных или подтверждения. |

Для остановки системы переадресации, выбранной при установке, используется команда NET STOP без параметров. При этом производится отключение всех общих ресурсов и удаление команд NET из памяти компьютера.

Вывод времени и синхронизация часов компьютера с общими часами на сервере времени Microsoft Windows для рабочих групп, Windows NT.

NET TIME [\\компьютер | /WORKGROUP:группа] [/SET] [/YES]

| | |
|------------|--|
| компьютер | Имя компьютера (сервера времени), предназначенного для вывода или синхронизации времени. |
| /WORKGROUP | Этот ключ указывает необходимость использования часов компьютера из другой рабочей группы. |
| группа | Имя рабочей группы, в которую входит нужный компьютер. При наличии в группе нескольких серверов времени команда NET TIME использует первый найденный из них. |
| /SET | Синхронизация часов компьютера с часами |

| | |
|------|--|
| | указанного компьютера или рабочей группы. |
| /YES | Выполнение команды NET TIME без предварительного запроса данных или подтверждения. |

Подключение и отключение общих ресурсов и вывод сведений о подключениях.

NET USE [диск: | *] [\\компьютер\папка [пароль | ?]]
[/SAVEPW:NO] [/YES] [/NO]

NET USE [порт:] [\\компьютер\принтер [пароль | ?]]
[/SAVEPW:NO] [/YES] [/NO]

NET USE диск: | \\компьютер\папка /DELETE [/YES]

NET USE порт: | \\компьютер\принтер /DELETE [/YES]

NET USE * /DELETE [/YES]

NET USE диск: | * /HOME

| | |
|------------|---|
| диск | Имя диска, назначаемое общей папке. |
| * | Эквивалент следующего свободного имени диска. При использовании совместно с ключом /DELETE производится отключение сразу всех ресурсов. |
| порт | Имя параллельного (LPT) порта, назначаемое общему принтеру. |
| компьютер | Имя компьютера, на котором расположен общий ресурс. |
| папка | Сетевое имя общей папки. |
| принтер | Сетевое имя общего принтера. |
| пароль | Пароль для доступа к общему ресурсу (если он имеется). |
| ? | Пароль для доступа к ресурсу запрашивается интерактивно. Этот режим может понадобиться лишь в том случае, когда ввод пароля необязателен. |
| /SAVEPW:NO | Использование этого ключа позволяет предотвратить запись пароля в файл со списком паролей. В этом случае при каждом подключении к ресурсу пароль надо будет вводить заново. |
| /YES | Выполнение команды NET USE без предварительного запроса данных или подтверждения. |
| /DELETE | Отключение общего ресурса. |
| /NO | Выполнение команды NET USE с автоматической выдачей отрицательных (NO) ответов на все запросы, |

| | |
|-------|--|
| | относящиеся к подтверждению действий. |
| /HOME | Подключение к основному каталогу, если сведения о нем имеются в учетной записи LAN Manager или Windows NT. |

Команда NET USE без параметров выводит список всех подключенных ресурсов.

Для просмотра выводимых сведений с паузами между отдельными экранами используются команды

NET USE /? | MORE
и
NET HELP USE | MORE

7.13 Вывод типа и версии используемой системы переадресации.

NET VER

Пример

```
net ver
```

Клиент Microsoft - Полная система переадресации. Версия 4.00

(c) Корпорация Microsoft, 1993-1995. Все права защищены.

Вывод списка компьютеров, входящих в рабочую группу, или списка общих ресурсов на заданном компьютере.

NET VIEW [\\компьютер] [/YES]
NET VIEW [/WORKGROUP:группа] [/YES]

| | |
|------------|---|
| компьютер | Имя компьютера, список общих ресурсов которого следует вывести. |
| /WORKGROUP | Этот ключ указывает необходимость вывода списка |

| | |
|--------|--|
| | компьютеров из другой рабочей группы, имеющих общие ресурсы. |
| группа | Имя рабочей группы, список компьютеров которой следует вывести. |
| /YES | Выполнение команды NET VIEW без предварительного запроса данных или подтверждения. |

Для вывода полного списка имеющих общие ресурсы компьютеров из рабочей группы, в которой зарегистрирован данный компьютер, используется команда NET VIEW без параметров.

Пример.

```
net view
```

Серверы, доступные рабочей группе PMI.

Сервер

Заметки

```

\\433_1           Computer 433_1
\\AG_STATION     AG Station
\\COMPUTER 429_2  Второй от двери компьютер
\\COMPUTER 429_3  Третий от двери компьютер
\\COMPUTER 429_41 Четвертый от двери компьютер
\\COMPUTER 429_5  Computer 429_5
\\COMPUTER 429_6  Шестой от двери компьютер
\\COMPUTER 429_8  Восьмой от двери компьютер
\\DMITRIEVA      Dmitrieva O. A.
\\MAG             MAG
\\PMI_NT_SERVER
\\SHOZDA         Computer 2000
\\TIGER          Tiger
\\C5_429_1       Первый от двери компьютер
Команда выполнена успешно.
```

Команды для работы с сетью в OS Linux

Команды, предназначенные для установления соединения с удаленной системой и проведения сеанса работы после установления соединения.

host – выводит IP-адрес указанной системы, используя службу DNS.

Можно указать IP-адрес, и он будет преобразован в имя системы.

Параметры: -d – отладочный режим;

- C – вывод списка всех систем в зоне;
- dd - аналогично d, но в более подробной форме.

hostname – выводит имя локальной системы.

- d – выводит имя DNS-сервера;
- f – вывод полного имени системы;
- s – кратко имени системы;

ping – отправляет пакеты на указанную систему для определения пропускной способности сети. Для получения более подробной информации см. man ping.

rlogin – позволяет провести сеанс работы на удаленной системе. Для получения более подробной информации см. man rlogin.

rwall <система> – отправляет сообщение всем пользователям, подключенным к указанной системе.

talk пользователь [терминал] – позволяет 2-м пользователям вести интерактивный разговор.

finger – выводит информацию об указанном пользователе.

Команды Linux: настройка сети:

traceroute[имя_машины] – маршрут передачи данных;

tracpath [имя_машины] - – маршрут передачи данных;

route -n - Выводит на экран таблицу маршрутизации;

netstat -rn - Выводит на экран таблицу маршрутизации;

[sudo] netstat -tup - Активные соединения с интернетом;

socklist - Показывает все открытые сокеты;

[sudo] netstat -anp --udp --tcp | grep LISTEN - Список приложений, которые открывают порты;

ifconfig - Показать параметры всех сетевых интерфейсов;

ifconfig eth0 - Показать параметры сетевого интерфейса eth0;

[sudo] ethtool eth0 - Показывает состояние сетевого интерфейса eth0 (для некоторых дистрибутивов требуется установка пакета ethtool). Команда

ethtool применяется только для проводных подключений, не работает с беспроводными интерфейсами;

[sudo] ethtool -s eth0 speed 100 duplex full autoneg off - Принудительно задать скорость сетевому интерфейсу 100Mbit и режим Full duplex и отключить автоматическое определение;

ifconfig eth0 192.168.50.254 netmask 255.255.255.0 - Задать основной IP адрес сетевому интерфейсу eth0;

ip addr add 192.168.50.254/24 dev eth0 - Задать основной IP адрес сетевому интерфейсу eth0;

ifconfig eth0:0 192.168.51.254 netmask 255.255.255.0 - Задать дополнительный IP адрес сетевому интерфейсу eth0;

ip addr add 192.168.51.254/24 dev eth0 label eth0:1 - Задать дополнительный IP адрес сетевому интерфейсу eth0;

[sudo] ifconfig eth0 up - Запустить сетевой интерфейс eth0;

[sudo] ifconfig eth0 down - Отключить сетевой интерфейс eth0;

ifconfig eth0 hw ether 00:01:02:03:04:05 - Смена MAC адреса;

[sudo] /etc/init.d/dhcpd restart - Перезагрузка DHCP клиента;

netstat -an | grep LISTEN - Показывает список всех открытых портов;

lsof -i - Показывает список всех открытых портов в сеть Internet;

[sudo] iptables -L -n -v - Показывает статус firewall (статус iptables);

[sudo] iptables -P INPUT ACCEPT - Открывает доступ ко всем портам;

[sudo] iptables -P FORWARD ACCEPT - Открывает доступ ко всем портам;

ifup - поднять сетевой интерфейс;

ifdown [имя_сетевого_интерфейса] – отключить сетевой интерфейс;

wget - [свободная](#) [неинтерактивная](#) [консольная](#) программа для загрузки файлов по сети. Поддерживает протоколы [HTTP](#), [FTP](#) и [HTTPS](#), а также поддерживает работу через HTTP [прокси-сервер](#). Программа включена почти во все дистрибутивы [GNU/Linux](#).

Задание к лабораторной работе

1. Определить IP-конфигурацию машины (WINIPCFG, IPCONFIG).
2. Получить статистику протоколов и текущих сетевых подключений (NETSTAT)
3. Определить видимость в сети компьютеров с заданными IP-адресами (PING).
4. Определить маршруты к узлам (TRACERT).
5. Получить список общих ресурсов текущей рабочей группы.
6. Продемонстрировать содержимое таблицы маршрутизации.
7. Продемонстрировать содержимое таблицы arp. Осуществить добавление статической записи в таблицу.
8. Запустить утилиту nslookup (для XP). Выполнить разрешение доменного имени в IP-адрес и наоборот.
9. Изучить основные команды для работы с сетью в ОС Linux, в отчете представить результаты работы некоторых команд (iptables используем только для просмотра, модификацию iptables проводить не рекомендуется в рамках лабораторной работы).
10. Уметь прокомментировать результаты работы сетевых утилит.

Результаты выполнения команд можно сохранить в текстовый файл.

Например:

```
tracert dgtu.donetsk.ua > report
```

Файл report будет содержать результат выполнения команды

```
tracert dgtu.donetsk.ua
```

Содержание отчета

1. Титульный лист
2. Результаты выполнения задания

- вызов утилит ipconfig, tracert, ping, arp, route, netstat, nslookup, net view
- результаты работы сетевых утилит под ОС Linux.

Контрольные вопросы:

- 1) Назначение протокола ARP и RARP.
- 2) Что такое метрика в таблицах маршрутизации?
- 3) Что такое TTL?
- 4) Назначение протокола ICMP?
- 5) Назначение протокола DHCP? Что такое срок аренды?
- 6) Выполнить сравнительный анализ работы 2-х протоколов транспортного уровня – TCP и UDP.
- 7) Что такое DNS.
- 8) Если на компьютере есть локальная сеть, но нет выхода в Internet, в чем может быть проблема? Объясните все возможные причины отсутствия выхода в Internet.
- 9) Что такое маска подсети?

Лабораторная работа №4

Тема: “Назначение IP-адресов. Маски подсети”

Цель: Изучение классификации IP-адресов. Назначение масок подсети.
Изучить механизм использования масок в IP-адресации.

Методические указания и задания к лабораторной работе

Одной из наиболее важных тем при обсуждении стека TCP/IP является IP-адресация. *IP-адрес* представляет собой числовой идентификатор, присваиваемый каждому компьютеру сети IP. Он отражает расположение устройства в сети. IP-адрес является программным, а не аппаратным адресом — последний “зашит” в компьютере или плате сетевого интерфейса. IP-адреса позволяют хостам одной сети взаимодействовать с хостами другой сети вне зависимости от типов этих локальных сетей.

Перед подробным изучением IP-адресации нужно усвоить несколько базовых понятий и терминов.

Термины IP-адресации

Byte (байт) 7 или 8 бит, в зависимости от использованной схемы проверки четности. В этой главе мы будем считать, что один байт всегда равен 8 бит.

Octet (октет) Всегда равен 8 бит (разрядам).

Network address (сетевой адрес) Точка назначения, используемая в маршрутизации пакетов к удаленной сети, например сетевые адреса 10.0.0.0, 172.16.0.0 и 192.168.10.0.

Broadcast address (адрес широковещательной рассылки) Используется приложениями и хостами для пересылки информации всем узлам сети. Примеры адресов широковещательной рассылки: 255.255.255.255 (всем узлам всех сетей), 172.16.255.255 (всем подсетям и хостам сети 17.16.0.0), 10.255.255.255 (широковещательная рассылка всем подсетям и хостам сети 10.0.0.0).

Иерархическая схема IP-адресации

IP-адрес содержит 32 бита информации, которые разделяются на четыре однобайтовые (восьмибитовые) секции, иначе называемые *октетами*. Существуют три способа представления IP-адресов:

- Представление десятичными числами, разделенными точками, например 172.16.30.56
- Двоичное представление, например
10101100.00010000.00011110.00111000
- Шестнадцатеричное представление, например AC 10 IE 38

Здесь показаны три формы представления одного и того же IP-адреса. Шестнадцатеричное представление используется реже, чем двоичное или десятичное, но все же применяется в некоторых программах, например, в реестре Windows IP-адреса компьютеров хранятся в шестнадцатеричном виде.

Для адресации выбрана иерархическая схема с тремя уровнями иерархии: сеть, подсеть и хост.

Для примера рассмотрим структуру телефонного номера. Первая его часть (код региона) описывает обширную географическую область. Вторая часть (префикс) сужает эту область до зоны действия локальной телефонной станции. Последний сегмент (собственно номер телефона) определяет конкретное соединение. При IP-адресации также используется схема с тремя уровнями. Вместо того чтобы рассматривать 32-разрядную комбинацию как единый идентификатор, в адресе выделяются части для адреса сети и для адреса узла.

| | | | | |
|----------------|-----------------------------------|------|------|------|
| Класс А | Сеть | Хост | Хост | Хост |
| Класс В | Сеть | Сеть | Хост | Хост |
| Класс С | Сеть | Сеть | Сеть | Хост |
| Класс D | Многоадресная рассылка | | | |
| Класс E | Класс для исследовательских работ | | | |

Рисунок 1 - Адресация сетей

Адрес сети однозначно определяет сеть. В IP-адресах всех машин, подключенных к одной сети, указывается один и тот же адрес сети. Например, в IP-адресе 172.16.30.56 адресом сети может быть 172.16.

Адрес узла присваивается каждой машине сети. В отличие от адреса сети, описывающего группу устройств, адрес узла уникален и однозначно определяет конкретную машину сети. Адрес узла называют также *адресом хоста*. В приведенном примере адрес узла имеет вид 30.56.

Диапазон сетевых адресов класса А

Создатели схемы IP-адресации установили, что первый бит первого байта сетевого адреса сети класса А всегда выключен (т.е. равен 0). Следовательно, адреса класса А находятся между 0 и 127.

Диапазон сетевых адресов класса В

В сетях класса В спецификация RFC предписывает, что всегда должен быть включен первый бит первого *байта*, однако второй бит должен быть выключен. Если выключить, а затем включить остальные шесть разрядов, то мы получим диапазон для сетей В:

10000000=128

10111111=191

Следовательно, сети класса В имеют в первом байте значения от 128

до 191.

Диапазон сетевых адресов класса С

В сетях класса С спецификация RFC предписывает, что всегда должны быть включены два первых бита первого октета. Найдем диапазон для сети класса С преобразованием из двоичного вида в десятичный:

11000000=192

11011111=223

Следовательно, если начало IP-адреса находится между 192 и 223, то это адрес сети класса С.

Диапазоны сетевых адресов классов D и E

Адреса в диапазоне между 224 и 255 зарезервированы для сетей классов D и E. Класс D используется для многоадресных рассылок, а класс E — для исследовательских разработок. Далее мы не будем возвращаться к этим классам адресов.

Диапазоны сетевых адресов для специального применения

Некоторые IP-адреса зарезервированы для специальных целей и сетевые администраторы не могут присвоить их узлам своих сетей.

Зарезервированные IP-адреса

| Адрес | Функция |
|---|--|
| Сетевой адрес из всех нулей | Означает "эта сеть или сегмент". |
| Сетевой адрес из всех единиц | Означает "все сети". |
| Сеть 127.0.0.1 | Зарезервирована для кольцевого тестирования. Предназначена для сетевого узла, который может послать пакет себе без генерации сетевого трафика. |
| Адрес узла из всех нулей | Означает "этот узел". |
| Адрес узла из всех единиц | Означает "все узлы" определенной сети, например 128.2.255.255 показывает "все узлы сети 128.2 (адреса класса B)". |
| Весь IP-адрес из нулей | Используется маршрутизаторами Cisco для указания пути по умолчанию. |
| Весь IP-адрес из единиц (255.255.255.255) | Широковещательная рассылка по всем узлам текущей сети, иногда называется "широковещательной рассылкой по всем единицам". |

Рисунок 2 - Зарезервированные IP-адреса

Адреса класса А

В IP-адресе сетей класса А первый байт занимает адрес сети, а в трех последующих байтах размещается адрес узла. Формат IP-адреса сети класса А:

Сеть.Узел.Узел.Узел

Например, в IP-адресе 49.22.102.70 адрес сети равен 49, а адрес узла — 22.102.70. Каждая машина этой сети должна иметь адрес сети, равный 49. Адрес сети класса А имеет длину 1 байт, причем его первый бит зарезервирован, но доступны оставшиеся семь разрядов. Это означает, что можно создать не более 128 сетей класса А. Почему? Потому что каждый из семи оставшихся битов может принимать значение 0 или 1, т.е. существует 2⁷ или 128 различных комбинаций.

Однако было решено, что нулевой адрес сети (0000 0000) резервируется для обозначения маршрута, выбранного по умолчанию. Однако из-за того, что нулевой адрес зарезервирован, диапазон становится уже: от 1 до 127. В результате реальное число сетей класса А равно 128-2, т.е. 126.

Под адрес узла в IP-адресе сетей класса А отведено 3 байта (24 разряда). В них можно разместить 2²⁴ различных двоичных комбинаций или адресов узлов. Поскольку адреса, состоящие только из нулей и только из единиц, зарезервированы, точное число узлов в сети класса А составляет 2²⁴ - 2 = 16777214.

Допустимые значения идентификаторов хостов в сети класса А

Рассмотрим пример определения допустимого идентификатора хоста для сетевого адреса класса А:

10.0.0.0 В сетевом адресе выключены все разряды, определяющие идентификатор хоста.

10.255.255.255 Все разряды для хостов в широковещательном адресе.

Допустимое количество хостов находится в диапазоне между сетевым адресом и адресом широковещательной рассылки: от 10.0.0.1 до 10.255.255.254. Заметим, что допустимы идентификаторы хостов из всех

нулей и 255. Для подсчета количества доступных адресов хостов нужно, помнить, что разряды хоста не могут быть все вместе включены или выключены.

Адреса класса В

В IP-адресе сетей класса В первые два байта занимает адрес сети, а в двух последующих байтах размещается адрес узла. Формат IP-адреса сети класса В:

Сеть. Сеть.Узел.Узел

Например, в IP-адресе 172.16.30.56 адрес сети равен 172.16, а адрес узла — 30.56.

Для адреса сети, состоящего из 16 разрядов, имеется 216 возможных комбинаций. Однако разработчики Интернета решили, что адрес сети класса В должен начинаться с комбинации 10. Поэтому свободными для формирования адреса остаются лишь 14 бит; это означает, что может существовать 214 или 16 384 сетей класса В.

Под адрес узла в IP-адресе сетей класса В отведено 2 байта. Поскольку адреса, состоящие только из нулей и только из единиц, зарезервированы, точное число узлов в сети класса В равно $2^{16} - 2 = 65\,534$.

Допустимые значения идентификаторов хостов в сети класса В

Рассмотрим пример определения допустимого идентификатора хоста для сетевого адреса класса В:

172.16.0.0 В сетевом адресе выключены все разряды, определяющие идентификатор хоста.

172.16.255.255 Все разряды для хостов в широковещательном адресе.

Допустимое количество хостов находится в диапазоне между сетевым адресом и адресом широковещательной рассылки: от 172.16.0.1 до 172.16.255.254.

Адреса класса С

Первые три *байта*, в IP-адресе сетей класса C занимает адрес сети, и всего один байт остается для адреса узла. Формат IP-адреса сети класса C:

Сеть.Сеть.Сеть.Узел

Например, в IP-адресе 192.168.100.102 адрес сети равен 192.168.100, а адрес узла —102.

Первые три разряда адреса сети класса C занимает комбинация 110. Поэтому для формирования адреса остается лишь $24 - 3 = 21$ разряд. Таким образом, может существовать 2^{21} или 2 097 152 сетей класса C.

Под адрес узла в IP-адресе сетей класса C отведен 1 байт. Следовательно, в каждой сети класса C может быть $2^8 - 2 = 254$ узла.

Допустимые значения идентификаторов хостов в сети класса C

Рассмотрим пример определения допустимого идентификатора хоста для сетевого адреса класса C:

192.168.100.0 В сетевом адресе выключены все разряды, определяющие идентификатор хоста.

192.168.100.255 Все разряды для хостов в широковещательном адресу.

Допустимое количество хостов находится в диапазоне между сетевым адресом и адресом широковещательной рассылки: от 192.168.100.1 до 192.168.100.254.

Маска подсети

При применении схемы адресации с подсетями каждая машина сети должна знать, какая часть адреса хоста занята адресом подсети. Для этого на каждом компьютере создается *маска подсети*. Это 32-разрядное число, которое позволяет получателю пакета IP отделить идентификатор сети в IP-адресе от идентификатора хоста.

Администратор сети создает 32-разрядную маску подсети, состоящую из 0 и 1. Единицы в маске подсети помечают позиции, относящиеся к адресам сети и подсети.

Не во всех сетях нужны подсети, т.е. иногда используются маски подсети по умолчанию (иными словами, в такой сети нет адресов подсетей).

Маски подсетей по умолчанию

| Класс | Формат | Маска по умолчанию |
|-------|---------------------|--------------------|
| A | Узел.Узел.Узел.Узел | 255.0.0.0 |
| B | Сеть.Сеть.Узел.Узел | 255.255.0.0 |
| C | Сеть.Сеть.Сеть.Узел | 255.255.255.0 |

Рисунок 3 - Маски подсетей по умолчанию

Выделение подсетей в классе C

Существуют разные способы выделения подсетей, среди которых можно выбрать наиболее подходящий для себя. Сначала мы обсудим двоичный метод, а затем познакомимся с другим способом выделения подсетей.

В адресном пространстве класса C для определения хостов доступны только 8 разрядов. Биты подсети отсчитываются слева направо без пропусков разрядов. Масками подсетей могут быть:

10000000=128

11000000=192

11100000=224

11110000=240

11111000=248

11111100=252

11111110=254

Спецификация RFC не разрешает использовать для подсетей только один разряд, поскольку он всегда будет либо включен, либо выключен, а это недопустимо. Следовательно, первой правильной маской подсети будет 192, а последней — 252, поскольку нужно не менее двух разрядов для указания хостов.

Двоичный метод: Выделение подсетей в классе C

Рассмотрим выделение подсетей в адресном пространстве класса C с помощью двоичного метода. Сначала следует выявить первую доступную

маску подсети, которая заимствует два разряда. Например, можно использовать 255.255.255.192.

$$192=11000000$$

Два разряда применяются для выделения подсетей, 6 разрядов определяют хосты в каждой подсети. Какими будут подсети? Поскольку разряды подсети не могут быть одновременно включены или выключены, допустимы только две подсети:

$$01000000=64 \text{ (все разряды хостов выключены) или}$$

$$10000000=128 \text{ (все разряды хостов выключены)}$$

Корректные адреса хостов находятся между подсетями, за исключением вариантов, когда одновременно включены или выключены все разряды хостов.

Для выявления адресов хостов нужно сначала выключить все разряды хостов в адресе, а затем включить их, чтобы найти широковещательный адрес подсети. Допустимые адреса хостов располагаются между двумя полученными адресами.

В таблице ниже показана подсеть 64, диапазон хостов и адрес широковещательной рассылки.

Подсеть 64

| Подсеть | Хост | Описание |
|---------|------------|---|
| 01 | 000000=64 | Сеть (первая операция) |
| 01 | 000001=65 | Первый допустимый хост |
| 01 | 111110=126 | Последний допустимый хост |
| 01 | 111111=127 | Широковещательный адрес (вторая операция) |

Рисунок 4 - Подсеть 64

В таблице ниже показана подсеть 128, диапазон хостов и адрес широковещательной рассылки.

Подсеть 128

| Подсеть | Хост | Описание |
|---------|------------|---------------------------|
| 10 | 000000=128 | Адрес подсети |
| 10 | 000001=129 | Первый допустимый хост |
| 10 | 111110=190 | Последний допустимый хост |
| 10 | 111111=191 | Широковещательный адрес |

Рисунок 5 - Подсеть 128

Операция проста, но в наших примерах рассмотрен только случай с двумя разрядами для подсети. Что делать, когда нужно 9, 10 или даже 20 разрядов? Рассмотрим альтернативный метод, пригодный для выделения большого количества подсетей.

Альтернативный метод:

Выделение подсетей в классе C

Установив маску подсети, следует определить количество подсетей, хостов и широковещательные адреса. Для этого нужно ответить на несколько простых вопросов:

1. Сколько подсетей формирует данная маска?
2. Сколько хостов будет в каждой подсети?
3. Каковы правильные подсети?
4. Каковы правильные хосты в каждой подсети?
5. Какие широковещательные адреса в подсетях?

Приведем примеры ответов на поставленные вопросы:

1. Сколько подсетей? $2^x - 2 =$ количество подсетей, где X равно количеству маскируемых разрядов (т.е. единиц). Например, для 11000000 мы имеем $2^2 - 2$, т.е. 2 подсети.
2. Сколько хостов в подсетях? $2^x - 2 =$ количество хостов в подсети, где X равно количеству немаскируемых разрядов (т.е. нулей). Например, для 11000000 мы имеем $2^6 - 2$, т.е. 62 хоста в подсети.

3. Каковы корректные подсети? $256\text{-маска_подсети} = \text{базовое_количество}$.
Например, $256 - 192 = 64$.

4. Каковы корректные хосты? Количество хостов равно разности между подсетями, минус "все нули" и "все единицы".

5. Каков широковещательный адрес в каждой подсети? Адрес широковещательной рассылки получается после включения всех разрядов хостов, поэтому легко вычисляется для любой подсети.

Примеры выделения подсетей в классе C

Рассмотрим несколько примеров выделения подсетей в классе C с помощью рассмотренных выше методов.

Пример 1: 255.255.255.192

Начнем с адреса подсети в классе C, который использовался в предыдущем примере (255.255.255.192), чтобы показать преимущество альтернативного метода над двоичным. В этом примере мы используем сетевой адрес 192.168.10.0 и маску подсети 255.255.255.192.

192.168.10.0=Сетевой адрес

255.255.255.192=Маска подсети

Не трудно получить ответы на пять основных вопросов:

1. Сколько подсетей? В 192 включены два разряда (11000000), поэтому $2^2 - 2 = 2$. (вычитание 2 связано с некорректными по определению адресами, в которых включены или выключены все разряды подсети).

2. Сколько хостов в подсети? Выключено 6 разрядов хоста (11000000), следовательно, $2^6 - 2 = 62$ хоста.

3. Какова правильная подсеть? $256 - 192 = 64$ и мы получаем первую подсеть, а также базовое количество (переменную). Далее следует складывать эту переменную до тех пор, пока не будет достигнута маска подсети. $64 + 64 = 128$. $128 + 64 = 192$, но это уже некорректная маска, поскольку в ней включены все разряды подсети. Итак, получаем две подсети: 64 и 128.

4. Каковы правильные хосты? Они находятся между подсетями. Проще всего выявить их адреса, записав адреса подсетей и адреса широковещательных рассылок.

5. Какие широковещательные адреса в подсетях? Это число находится перед следующей подсетью и имеет включенными все биты хостов.

В таблице ниже показаны подсети 64 и 128, диапазон хостов в каждой из них и широковещательные адреса в каждой подсети.

Диапазоны подсетей 64 и 128

| Первая подсеть | Вторая подсеть | Описание |
|----------------|----------------|---|
| 64 | 128 | Подсеть (первая операция) |
| 65 | 129 | Первый хост (адреса хостов вычисляются позже) |
| 126 | 190 | Последний хост |
| 127 | 191 | Широковещательный адрес (вторая операция) |

Рисунок 6 - Диапазоны подсетей 64 и 128

Мы получили те же ответы, что и в двоичном методе, но нам уже не пришлось прибегать к преобразованию числа из двоичного вида в десятичный. Однако этот метод не всегда будет проще двоичного. Для первой подсети, где только два разряда подсети, двоичный метод будет удобнее. Возможно, следует хорошо изучить оба метода, поскольку часто приходится выполнять вычисления о подсетях в уме.

Остальные примеры вычисления масок можно найти в литературе:

1) CCNA Cisco Certified Network Associate Учебное руководство, Экзамен 640-507, Тодд Леммл, Издательство "Лори", 2002 г.

Задание к лабораторной работе:

1 Классификация IP-адресов.

- 1.1 Перевести число из двоичной системы в десятичную.
- 1.2 Перевести число из десятичной системы в двоичную.
- 1.3 Представить IP-адреса в двоичном формате и определить класс сети.

2 Разбиение сети на подсети

Дана сеть класса В. Необходимо ее разбить на 8 подсетей.

2.1 Определить маску каждой из подсетей

2.2 Определить номера подсетей

2.3 Определить число хостов в каждой из подсетей. Привести примеры IP-адресов хостов во всех подсетях и привести диапазон IP-адресов хостов.

3 Дана сеть класса С. Определить префикс сети, который позволит создать N хостов в каждой подсети.

3.1 Какое число компьютеров можно подключить к каждой подсети?

3.2 Какое максимальное число подсетей может быть определено?

3.3 Привести номера подсетей в двоичном формате и точечной нотации.

3.4 Привести пример IP-адресов хостов в подсети номер M. Привести диапазон IP-адресов в этой подсети.

3.5 Для подсети M определить широковещательный адрес. Привести его в десятичном и двоичном формате.

Варианты заданий см. в таблице ниже.

Таблица 1 – Варианты к заданиям

| Вар | Пункт 1.1 | Пункт 1.2 | Пункт 1.3 | Задание 2 | Задание 3 | | |
|-----|---|-------------------|---|------------|--------------|----|---|
| | | | | | IP | | |
| 1 | 01100110, 10111001, 11100111, 00111011 | 165, 254, 23, 56 | 127.0.1.2, 198.45.238.38, 45.218.75.1 | 136.56.0.0 | 196.56.4.0 | 17 | 2 |
| 2 | 01011101, 11110010, 00110110, 10011101 | 24, 156, 89, 246 | 156.23.65.2, 24.67.149.16, 62.48.179.23 | 145.78.0.0 | 210.234.6.0 | 20 | 6 |
| 3 | 10011010, 00110110, 10011011, 01111000 | 254, 125, 23, 156 | 13.15.56.16, 165.48.14.98, 78.245.11,23 | 186.5.0.0 | 208.25.198.0 | 9 | 8 |
| 4 | 01101111, 01110100, 00110011, 01101111 | 248, 26, 89, 183 | 202.11.23.7, 49.10.22.98, 109.252.26.23 | 173.98.0.0 | 194.168.23.0 | 23 | 7 |

| | | | | | | | |
|----|---|-------------------|---|-------------|--------------|----|---|
| 5 | 10111011, 11101101, 01101111, 01011011 | 35, 81, 193, 46 | 187.23.65.1, 26.23.26.4, 69.136.32.14 | 129.37.0.0 | 199.242.3.0 | 31 | 4 |
| 6 | 01101000, 10011011, 01110011, 00111011 | 149, 167, 23, 49 | 54.23.65.4, 195.26.156.5, 127.0.0.1 | 181.64.0.0 | 193.25.165.0 | 12 | 2 |
| 7 | 01101111, 01011101, 01111111, 11111011 | 45, 64, 121, 221 | 200.25.121.1, 126.2.23.1, 36.1.46.5 | 156.23.0.0 | 205.32.57.0 | 6 | 4 |
| 8 | 10111011, 01110111, 01011101, 10010011 | 158, 172, 45, 250 | 46.56.66.76, 189.12.136.1, 56.11.46.14, | 162.28.0.0 | 201.34.26.0 | 18 | 1 |
| 9 | 01110111, 10011101, 11100110, 10111011 | 188, 165, 149, 13 | 38.46.16.16, 159.16.0.4, 168.197.12.3 | 176.2.0.0 | 200.234.59.0 | 28 | 5 |
| 10 | 10001011, 01101110, 01111111, 01001101 | 154, 198, 67, 59 | 86.16.4.3, 74.23.49.1, 136.15.48.1 | 189.37.0.0 | 195.65.23.0 | 22 | 6 |
| 11 | 01101011, 01011011, 01110010, 10011101 | 56, 165, 89, 143 | 194.168.1.3, 65.111.166.1, 24.1.49.1 | 164.168.0.0 | 197.148.6.0 | 14 | 7 |
| 12 | 10100101, 01011001, 10011011, 10111101 | 226, 167, 165, 8 | 33.48.19.16, 126.16.19.4, 176.16.48.3 | 138.195.0.0 | 198.29.163.0 | 7 | 5 |

Контрольные вопросы к лабораторной работе:

- 1) Что такое IP-адрес?
- 2) Какие классы IP-адресов Вы знаете?
- 3) Что такое широковещательный адрес?
- 4) Для чего используются маски подсети?

Лабораторная работа № 5

Тема: Исследование сетевых протоколов

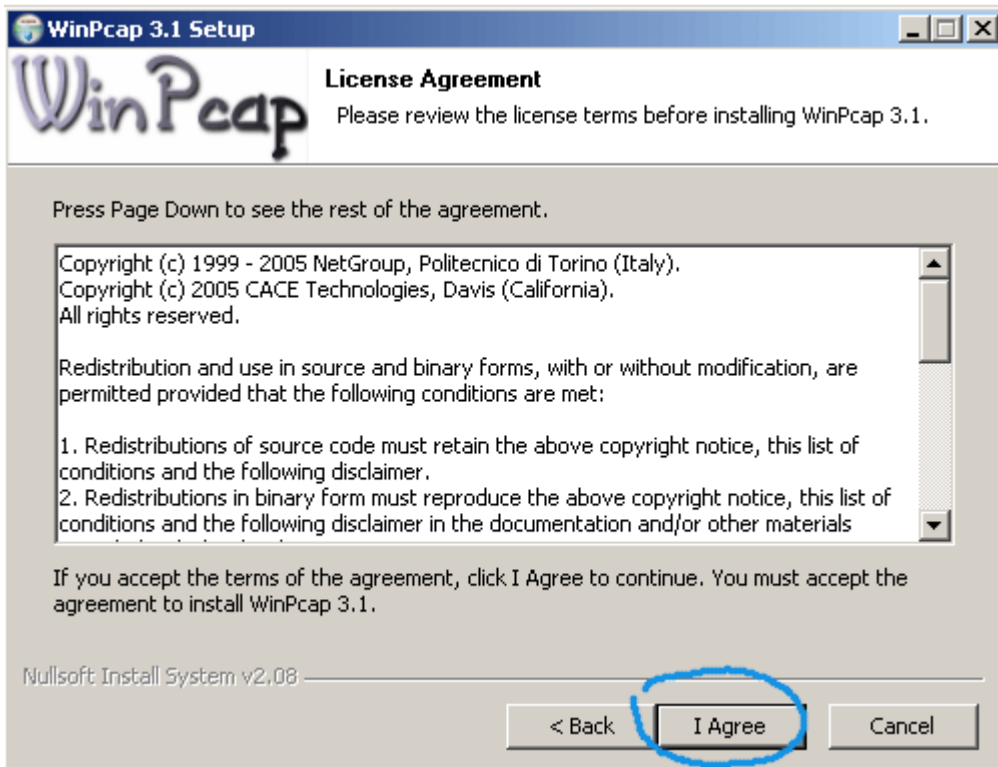
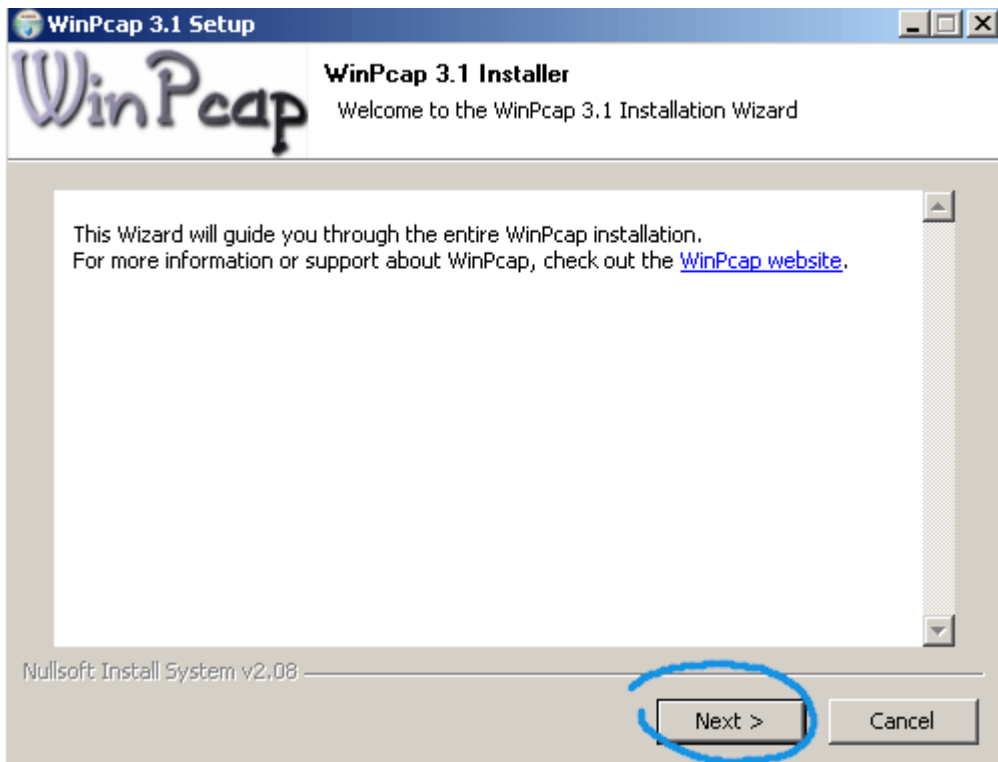
Цель: Приобретение практических навыков в анализе пакетов, передаваемых по сети, с использованием программы-сниффера.

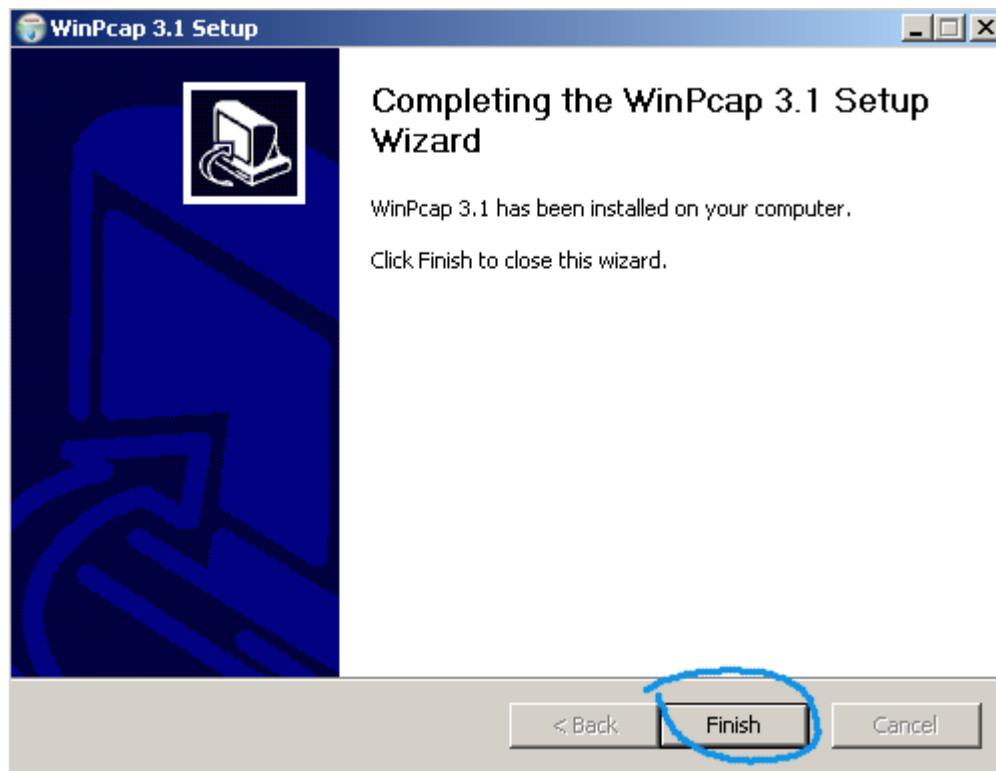
Методические указания к выполнению лабораторной работы

Для выполнения лабораторной работы понадобится специальная программа - сниффер. Сниффер - это программа, которая позволяет фиксировать все пакеты, которые приходят на сетевой интерфейс компьютера, накапливать их, сохранять и анализировать содержимое. Подобные программы могут работать в двух режимах: выборочном и неразборчивом. В выборочном режиме фиксируются только те пакеты, которые предназначены данному интерфейсу, в неразборчивом (promiscuous) фиксируются любые пакеты, полученные интерфейсом.

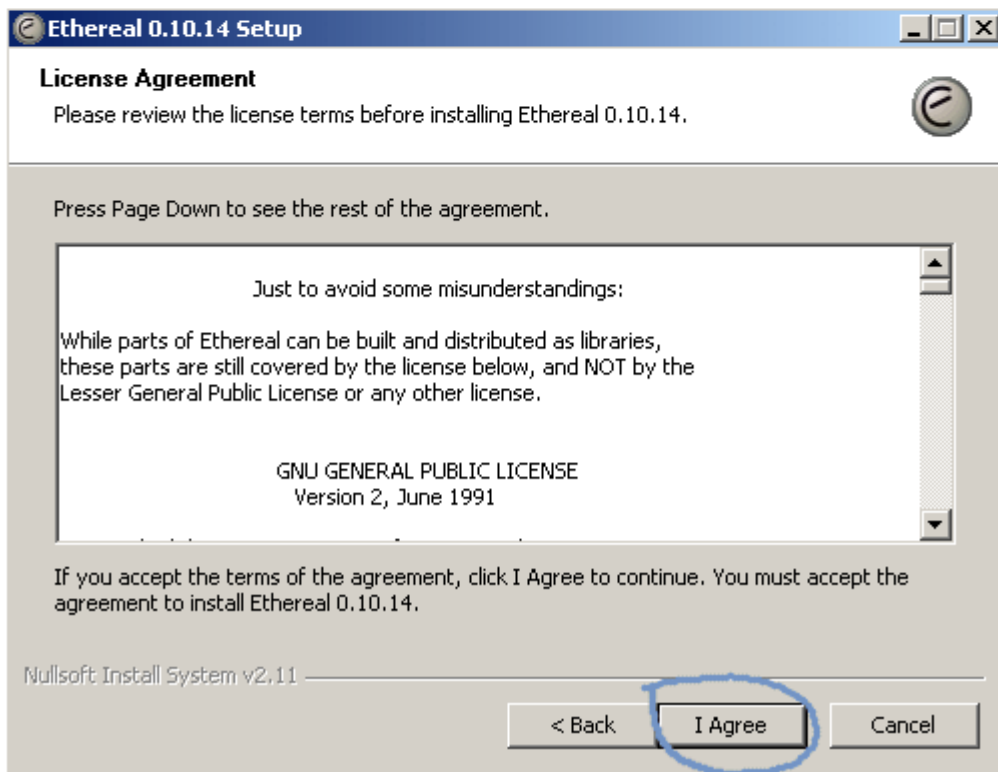
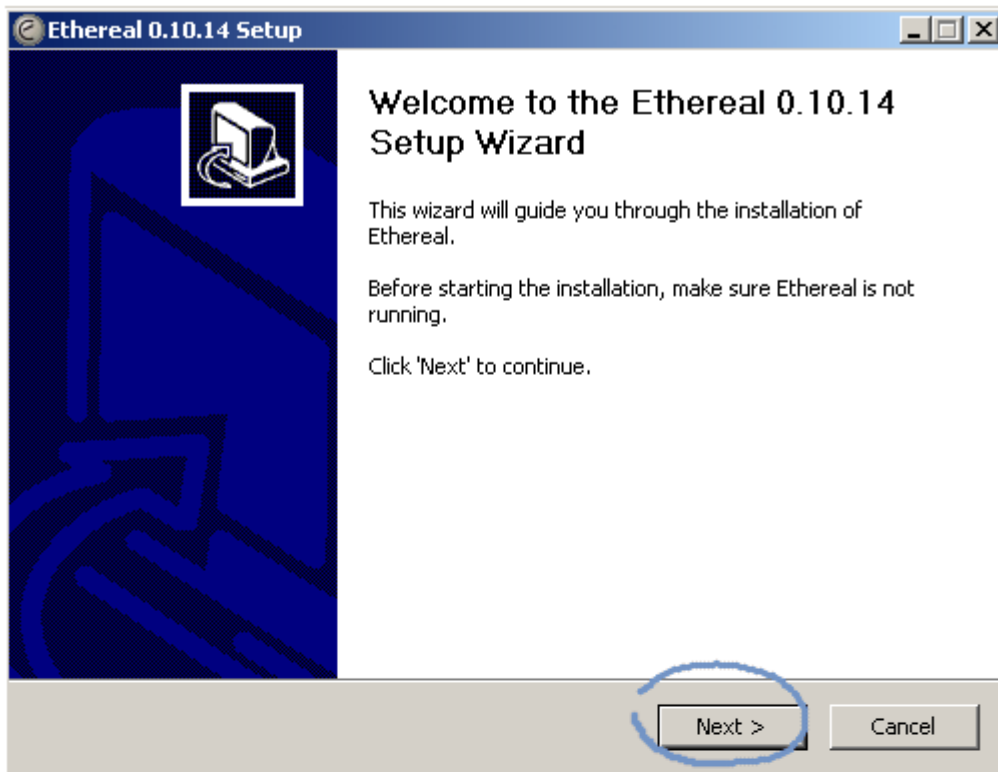
Для выполнения работы предлагается использовать сниффер Ethereal www.ethereal.com. Данная программа имеет интуитивно понятный, удобный графический интерфейс, обладает широкими возможностями по фильтрации пакетов и анализу их содержимого для более чем 400 протоколов. Для работы программы под управлением ОС Windows требуется предварительная установка библиотеки WinPCap www.winpcap.org (последняя версия 3.1)

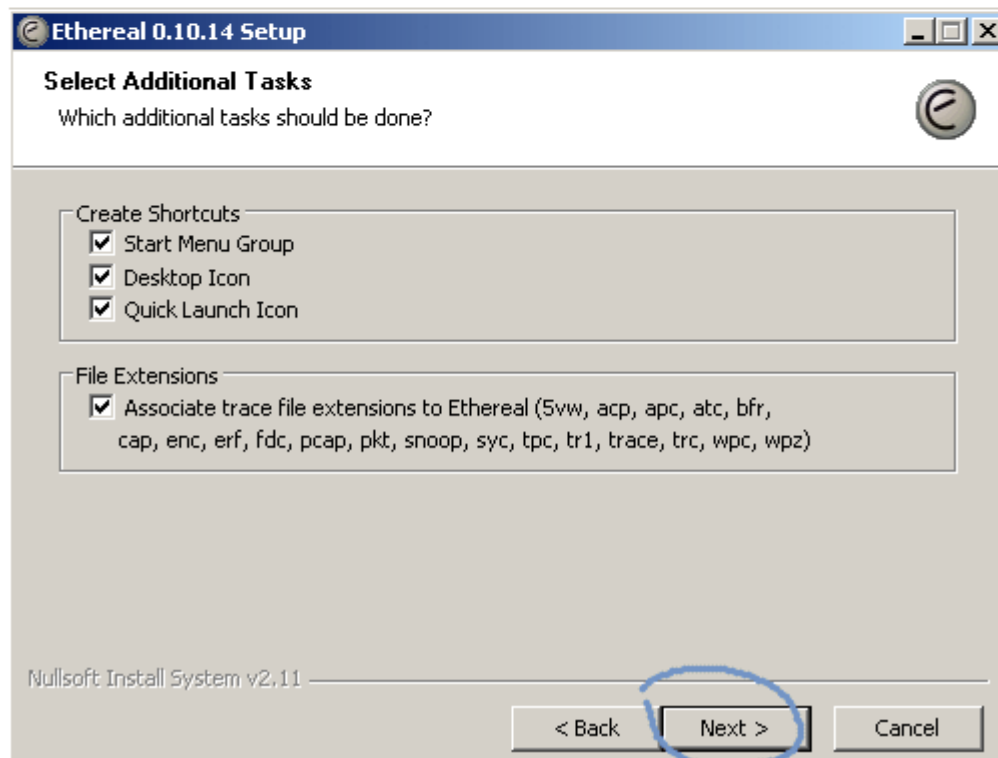
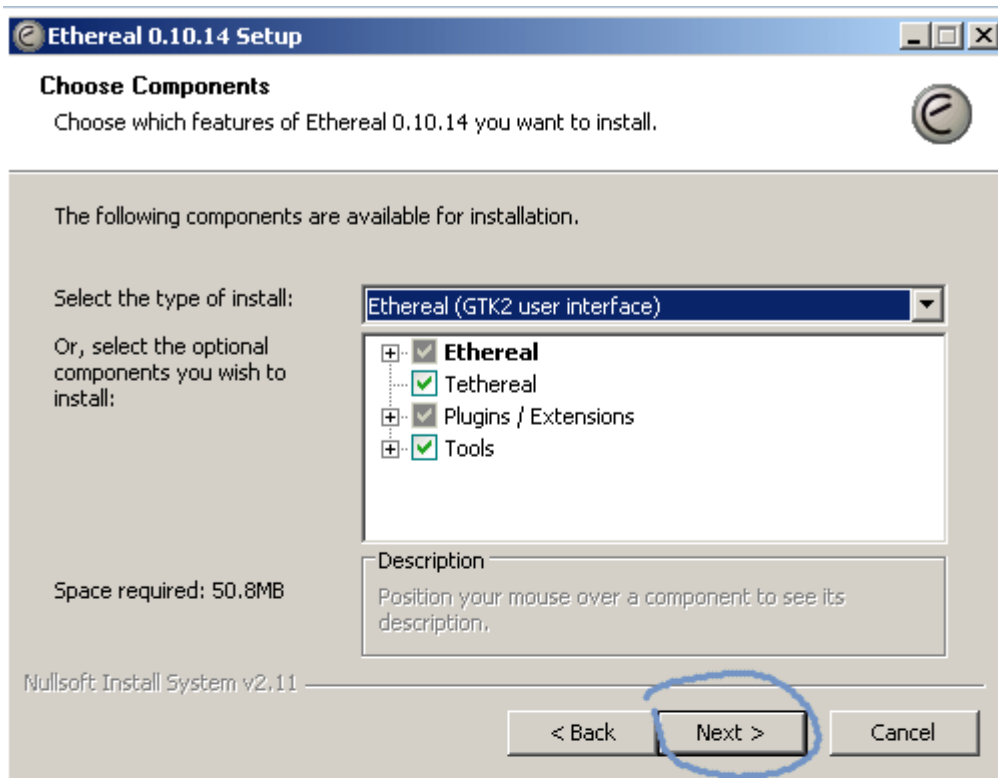
Для установки библиотеки WinPCap следуйте рекомендациям на рисунках:

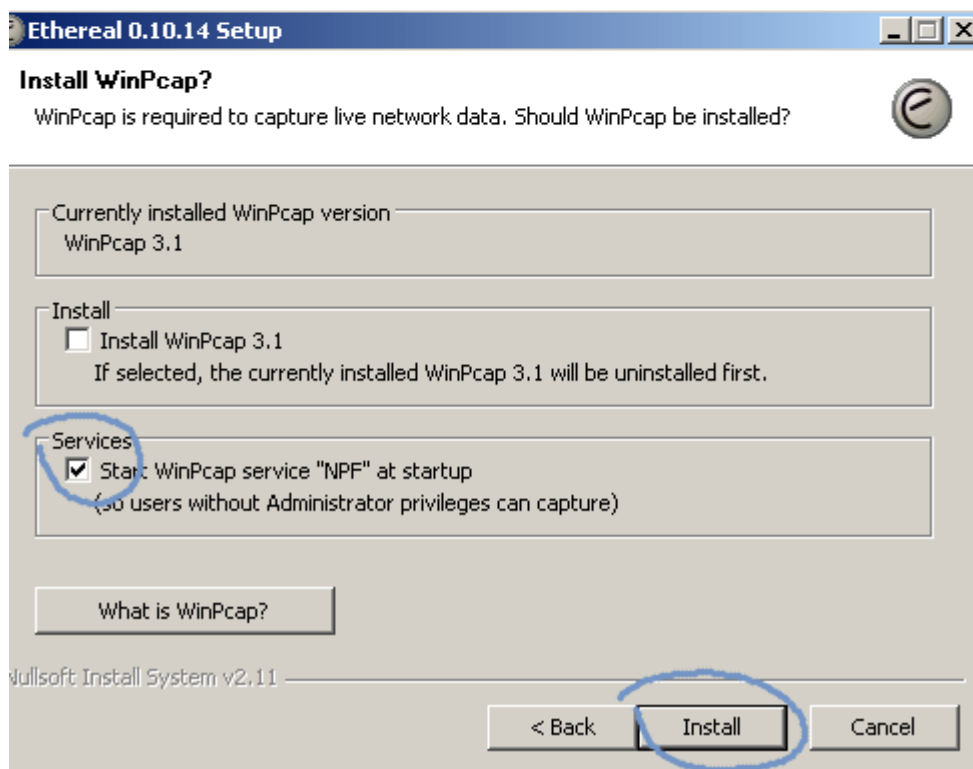
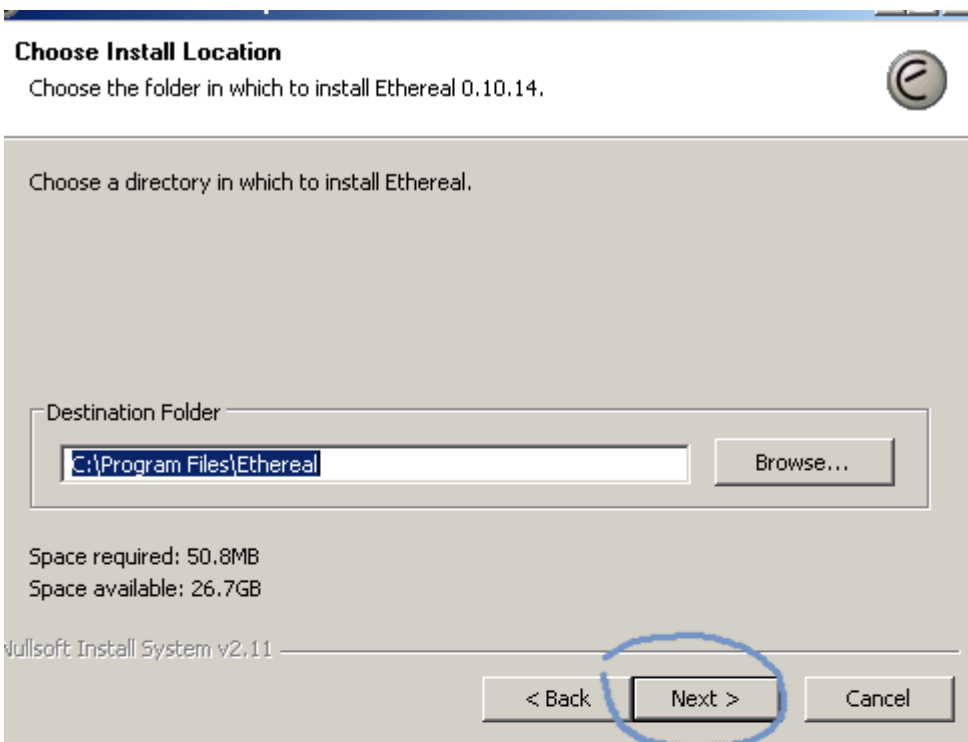


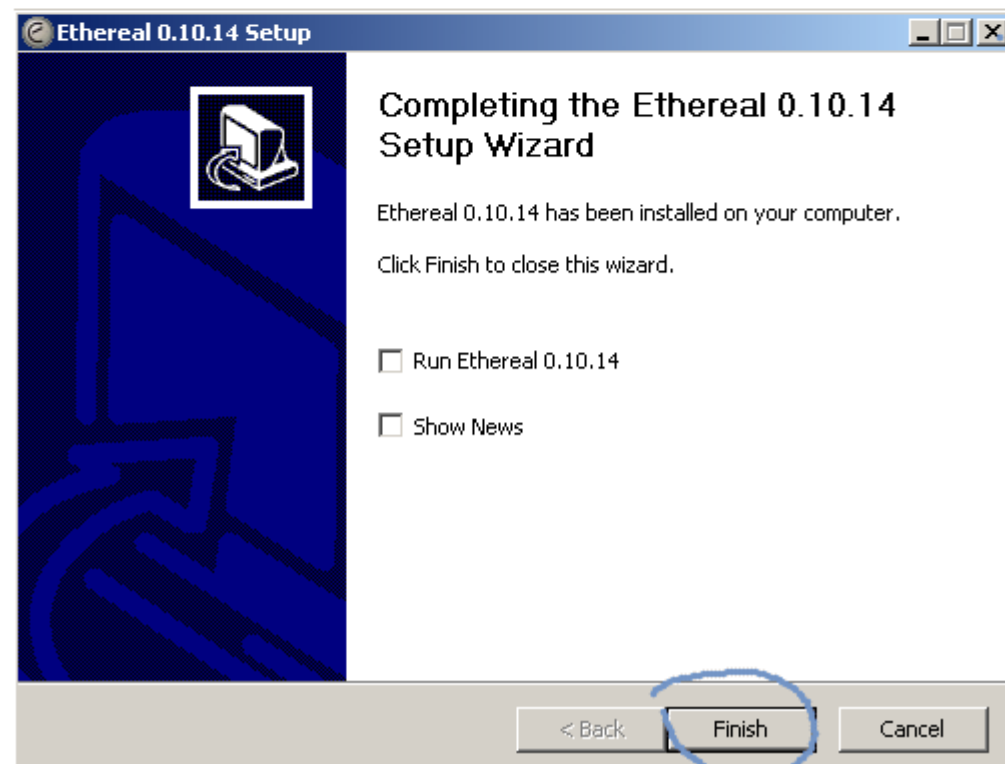
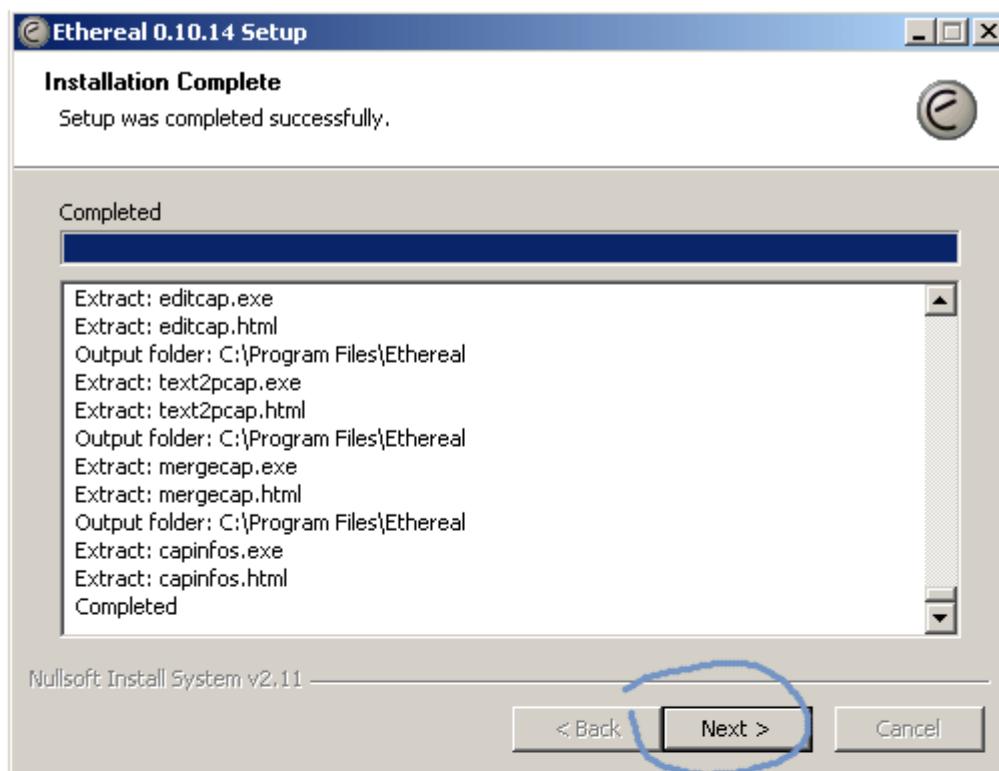


Для установки программы Ethereal следуйте рекомендациям на рисунках:

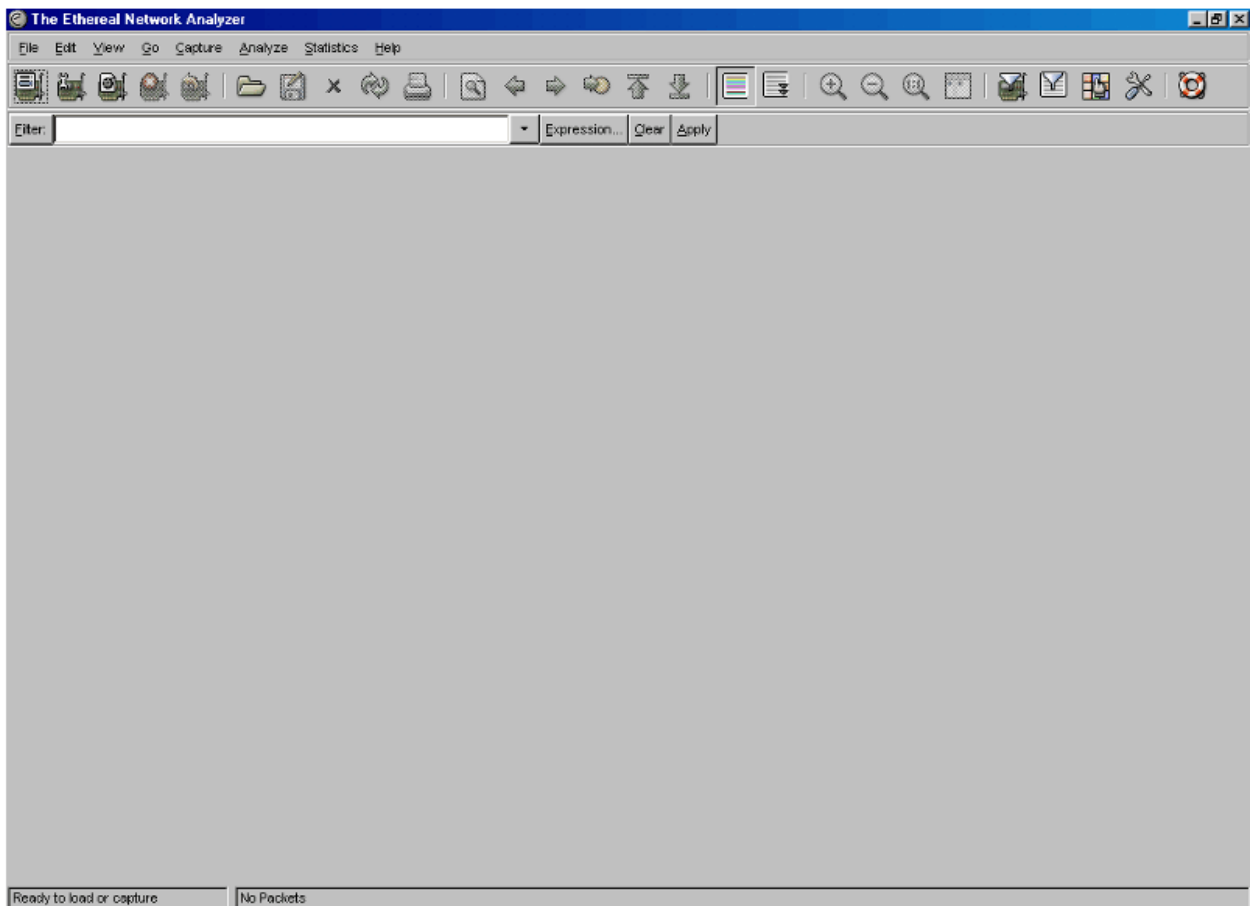




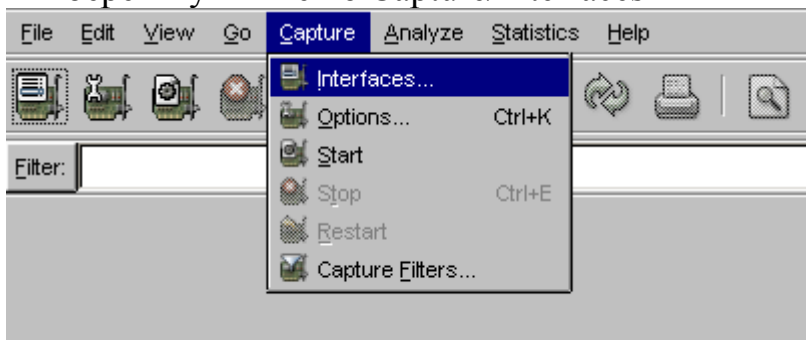




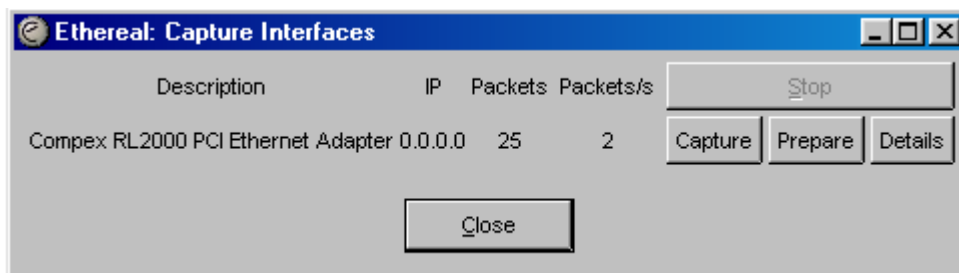
Запуск сниффера для перехвата пакетов (см. рисунки ниже)



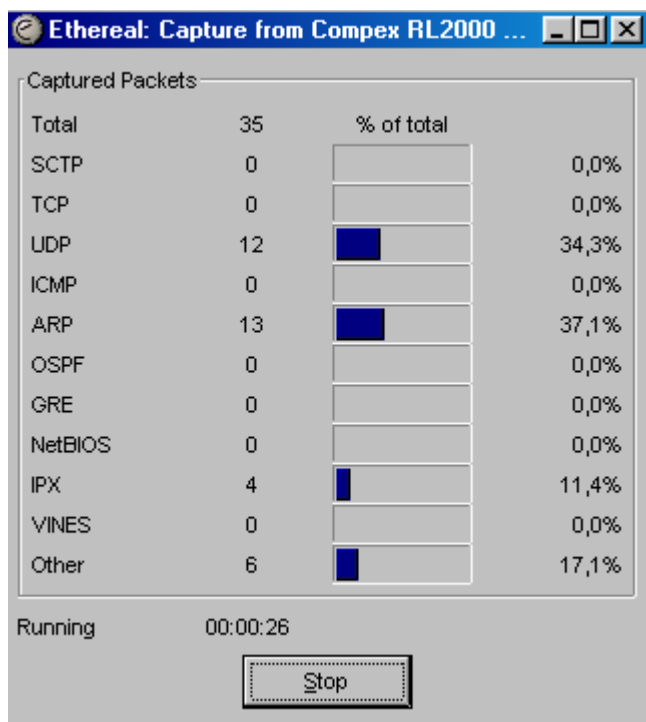
Выберем пункт меню Capture/Interfaces



В появившемся окне выберем интерфейс, на котором будем перехватывать пакеты. Если на компьютере в данный момент только один активный интерфейс, то он будет единственным отображаться в окне.



Нажмем кнопку Capture

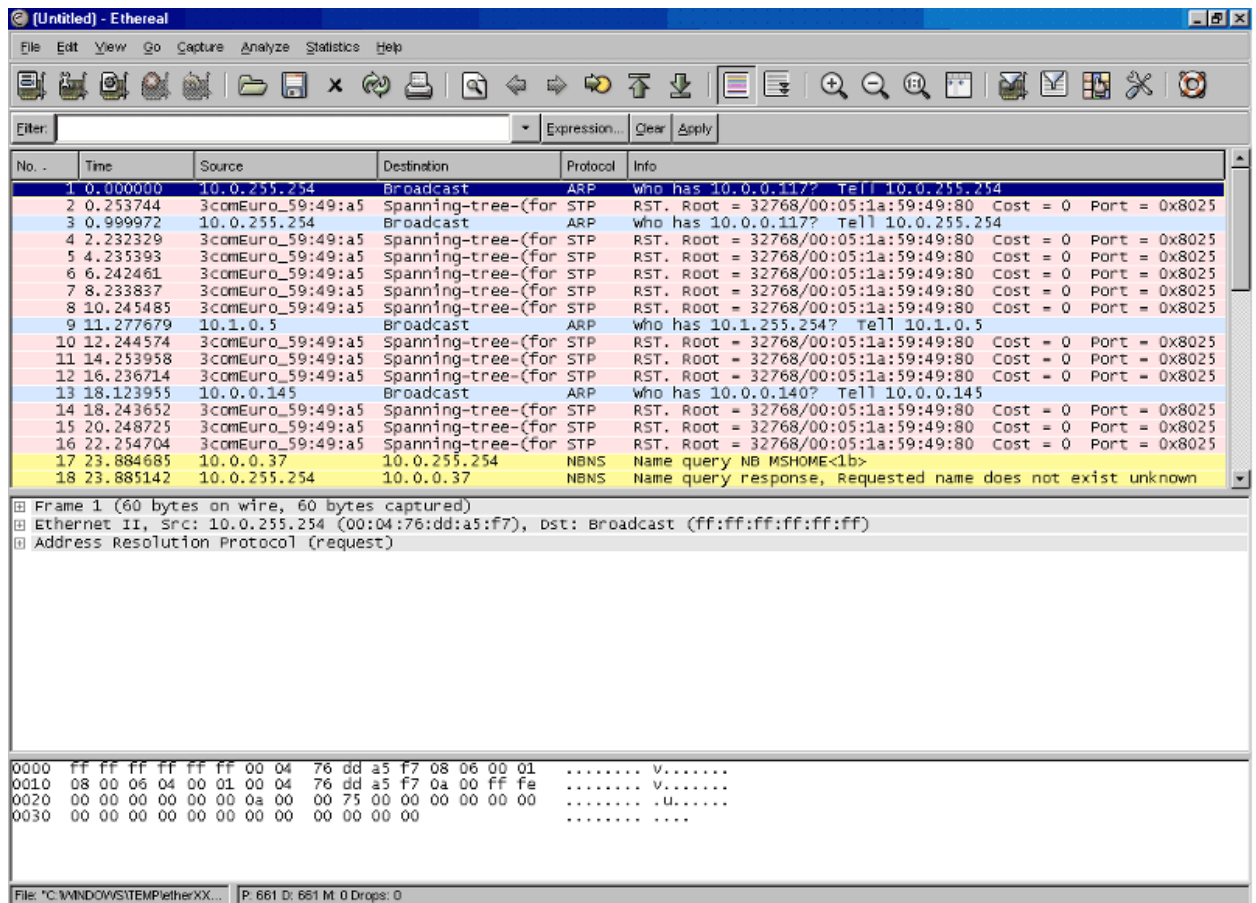


В появившемся окне будет отображаться ход перехвата пакетов. Для наиболее известных протоколов будет отображаться статистика перехвата.

Далее для выполнения лабораторной работы необходимо смоделировать ситуации, которые привели бы к появлению пакетов требуемых нам протоколов: TCP, UDP, HTTP, FTP.

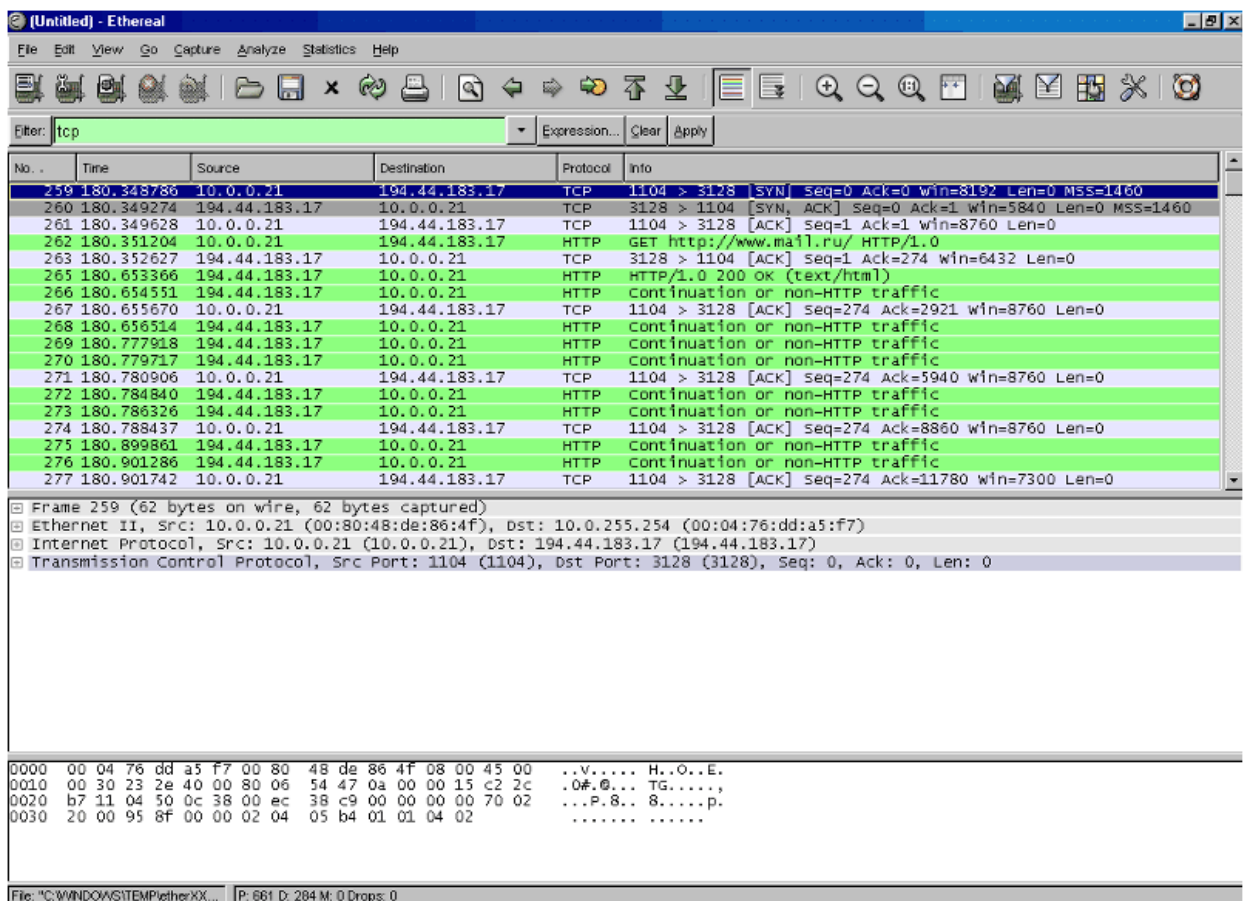
Анализ протоколов TCP и HTTP

Для получения последовательности TCP-пакетов рекомендуется при запущенном сниффере открыть в браузере какую-либо страницу, например www.mail.ru. В результате в окне статистики перехвата должен появиться определенный процент TCP-пакетов. Дождитесь окончания загрузки страницы. Далее в окне статистики перехвата нажмите Stop. После обработки пакетов, которая может занять некоторое время, на экране появится следующая информация:

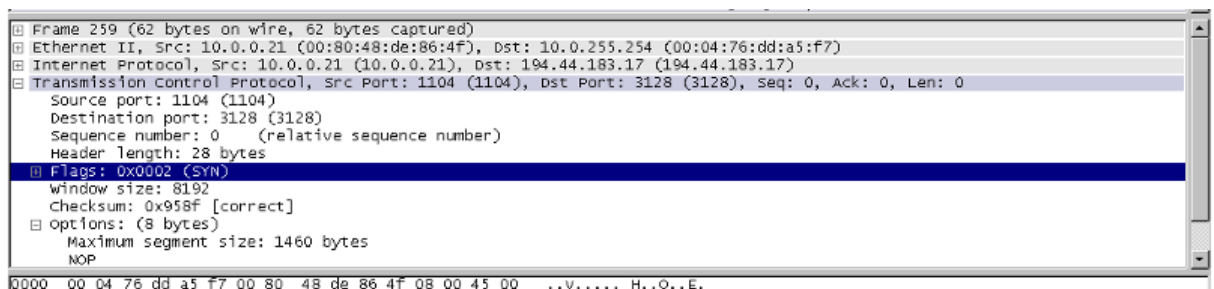


Окно программы разбивается на три части. Первая содержит список всех перехваченных пакетов. Вторая - содержимое текущего выделенного пакета. Третья - шестнадцатиричный дамп памяти, соответствующий выбранному пакету.

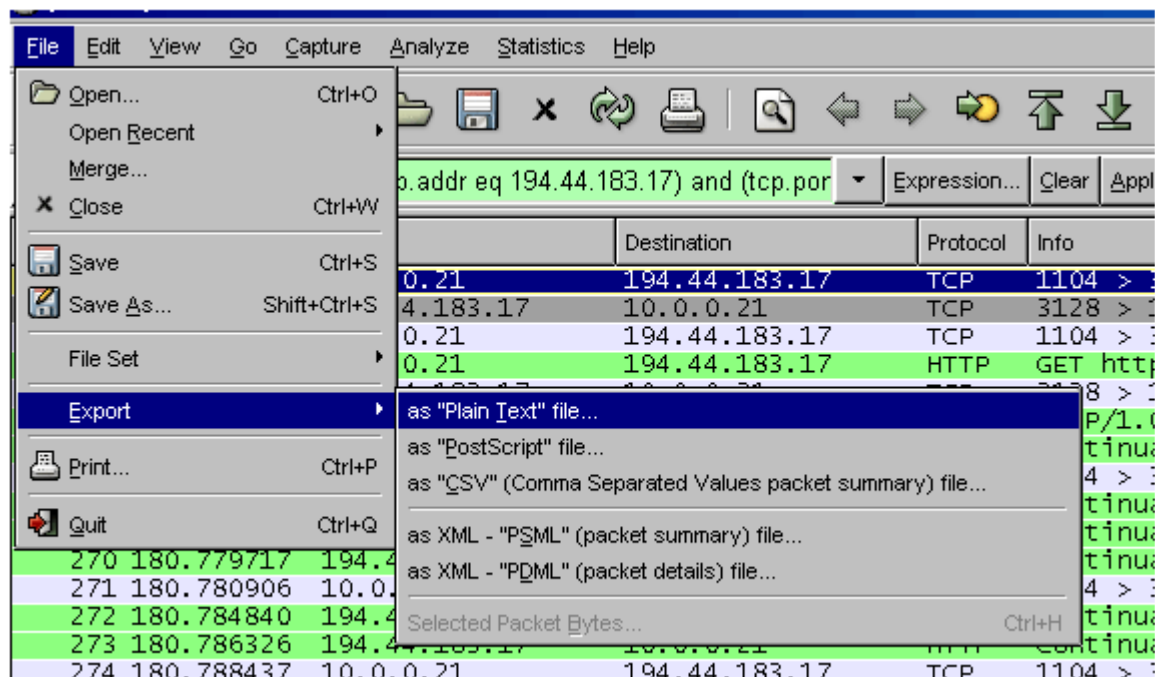
Для отбора интересующих нас пакетов применим фильтр. Поле фильтра находится под панелью инструментов в верхней части окна. Введем в окно фильтра выражение tcp и нажмем ввод. В результате фильтрации в верхнем окне останутся только TCP-пакеты. Наличие HTTP-пакетов объясняется тем, что на транспортном уровне протокол HTTP использует именно TCP.



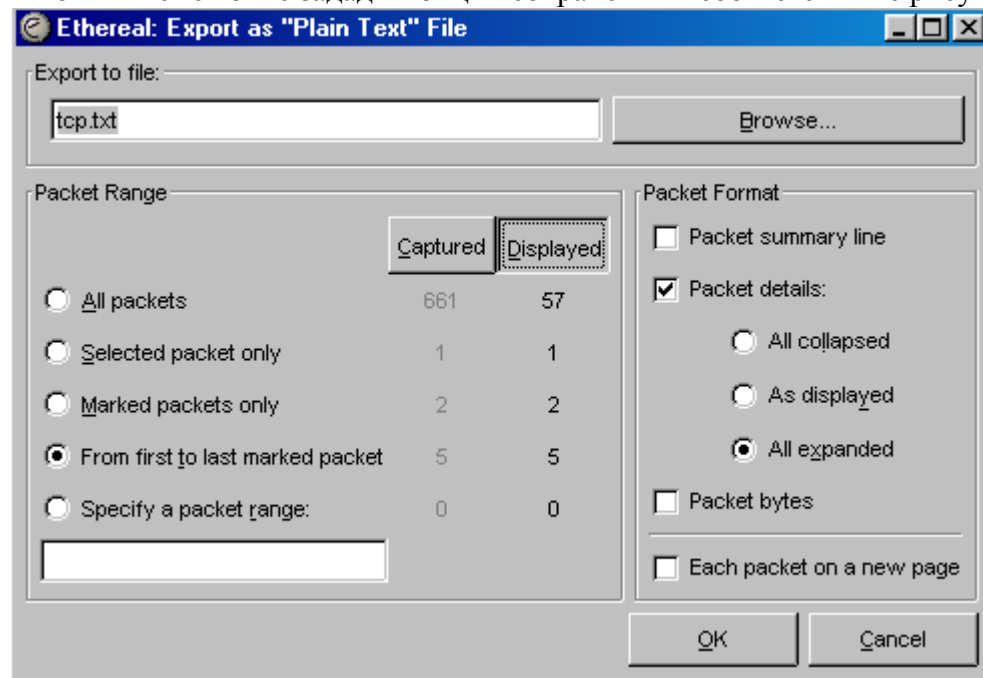
Для подробного изучения содержимого пакетов можно раскрывать протоколы всех уровней, как показано на рисунке:



Для того, чтобы гарантировать выбор TCP-пакетов только для конкретного соединения, выделим TCP-сессию. Для этого выделим первый пакет в списке и, нажав правую клавишу мыши, выберем пункт Follow TCP Stream.



В появившемся окне зададим опции сохранения в соответствии с рисунком.



Выберем путь и имя файла "tcp.txt". Укажем, что сохранять необходимо только отображаемые "Displayed" пакеты и только с первого по последний помеченный "From first to last marked frames". Для пакетов укажем их формат: без суммарной информации о пакете, раскрывать все уровни "All expanded", не сохранять байты пакета. После чего нажмем Ок.

Задание к лабораторной работе:

Изучить с помощью сниферов структуры пакетов протоколов tcp (3-4 пакета одной сессии), udp, http, icmp, arp, ip, tcr. Сравнить с описанием основных полей заголовков протоколов, представленных в RFC.

Требования к отчету:

В отчете по лабораторной работе привести назначение каждого из вышеперечисленных протоколов, которые требуется изучить. Привести анализ своего трафика для каждого из изучаемых протоколов, проанализировать структуры заголовков пакетов, сравнив с описанием в RFC. Скриншоты работы снифера, демонстрирующие передачу данных по выше указанным протоколам привести в отчете.

Контрольные вопросы:

- 1) Назначение протокола ip.
- 2) Назначение протокола udp.
- 3) Принципы передачи данных протокола tcp.
- 4) Структура arp-запроса и arp-ответа.
- 5) Назначение протокола icmp.
- 6) Структура заголовка протокола http.

Лабораторная работа №6

Тема: Моделирование протокола TCP

Цель работы: Изучить принцип работы протокола TCP с помощью программы моделирования JASPER.

Методические указания к выполнению лабораторной работы

JASPER- гибкая и в то же время мощная система для моделирования протоколов, которая дает возможность моделировать с протоколами и получать результаты в графическом виде. Jasper имеет модульную структуру, что позволяет легко добавлять новые протоколы.

Структура системы позволяет при реализации новых протоколов абстрагироваться от не значительных деталей в моделях протоколов. Например, использовать абстрактный формат сообщений, не уделять внимание эффективности модели протокола, с точки зрения ее программной реализации (интересует не скорость выполнения определенных операций, а их последовательность и взаимосвязь). Все выше перечисленное позволяет реализовывать модели достаточно сложных протоколов, например, таких как TCP, IP и т. д. В качестве языка программирования был выбран Java. При программировании на Java один и тот же код может выполняться как отдельное приложение, так и как апплет. Java предоставляет достаточно удобные средства для реализации графики. Объектно-ориентированная природа Java сделала более удобным и легким создание моделирующей системы с поддержкой .plug-in. протоколов. Протокол может расширять другой протокол, построенный на базовом классе протоколов.

Пользователь может контролировать моделирование протоколов с помощью пунктов меню, например либо отправить сообщение, либо закрыть соединение. Проблемы при передаче данных часто осложняют работу протокола, но пользователь также может контролировать работу протокола в такой момент, именно пользователь решает, должно ли сообщение быть потеряно или фрагментировано в течение передачи.

Отметим следующие преимущества моделирующей системы:

- в виду того, что код, написанный на Java, имеет высокую переносимость, Jasper может без проблем работать на разных платформах;
- моделирование происходит в интерактивном режиме и под контролем пользователя;
- Jasper является хорошим примером объектно-ориентированного программирования на Java.

Большинство существующих программ поддерживают моделирование протоколов и анализ производительности работы, но их основной целью является именно изучение и оценка производительности, а не детальное рассмотрение работы протокола. Один из основных примеров это программа *ns/nam*[2] (Network Simulator/Network Animator), которая поддерживает анализ моделей сетей. Существует достаточное количество готовых моделей Интернета и протоколов беспроводных сетей. Другой пример - программа *cnet*[3] (computer network), которая используется для моделирования сетей и определения производительности сети как единого целого.

Большое количество программ моделирующих работу сетей разработано в рамках проекта VINT (Virtual Internet). Как и в предыдущих случаях, они отличаются от Jasper тем, что фокусируют свое внимание на производительности, а не на функциональности и свойствах моделей. Другой класс моделирующих систем связан с распределенными алгоритмами, но они отличаются представлением результатов и целями от программ, моделирующих протоколы. Учитывая цели и возможности моделирующей системы, можно сказать, что Jasper является моделирующей системой уникальной в своем роде.

Общая структура

Структура файловой системы программного комплекса представлена на рисунке 1.



Рисунок 1 - Структура файловой системы программы

Программа разработана в соответствии с широко известной парадигмой MVC:

Model:

Модель протокола и возможные операции определены как набор Java классов. Хотя любой стиль программирования может быть адаптирован для программирования протоколов, но наиболее подходящим является объектно-ориентированный стиль, акцентирующий внимание на состоянии протокола.

Такой стиль используется в стандартах различных протоколов и реализуется на специфических языках ESTELLE или SDL. Машина состояний проще реализуется на обычных языках программирования.

Основные состояния . Это нумерованный тип (список переменных типа `final int` в Java).

Второстепенные состояния . Это простые переменные. Машина состояний кодируется как таблица или как набор операторов `if/switch` (так сделано в JASPER). Преимущество данного подхода заключается в том, что упрощается процесс разработки протокола на основе базового класса. Несколько моделей в JASPER написаны с использованием ESTELLE или SDL.

View:

Изображение протокола . это графическое представление его свойств, его работы. Несколько видов графического представления работы были разработаны для Jasper, но на данный момент поддерживается только одно представление - TSD (Time Sequence Diagram) или временная диаграмма.

Такие диаграммы часто используются для представления работы протоколов, хотя детали представления отличаются среди авторов программ.

Вариант поддерживаемый Jasper похож на Service Conventions standard используемый для OSI (Open Systems Interconnection). Временные диаграммы состоят из столбцов, которые соответствуют участникам коммуникации, таким как двум протоколам и коммуникационной среде. Более сложные модели могут содержать несколько протоколов и также пользователей протоколов. С течением времени диаграмма движется вниз, показывая новые сообщения, характеризующие происходящие события.

Controller:

Основной класс программы играет роль глобального механизма управления. Он координирует действия участников коммуникации (пользователей, протоколов, коммуникационной среды). Правила протоколов определяют, какие действия в данный момент разрешены пользователю.

Выбор пользователя определяет следующий шаг в моделировании.

Пользователь также может отменить предыдущий шаг или вернуть отмененный шаг в последовательности событий.

Интерфейсы

Для реализации моделирования протоколов необходимы классы, с помощью которых будет организовано взаимодействие с подключаемыми протоколами.

Абстрактный класс *Protocol* обеспечивает базовую функциональность для управления классами протоколов, и должен быть наследован для каждого протокола. Каждый объект протокола моделируется классами, которые реализуют интерфейс *ProtocolEntity*. Пользователи и коммуникационная среда также является реализацией интерфейса *ProtocolEntity*. Это позволяет однообразно получать доступ ко всем участникам коммуникации.

Интерфейс *ProtocolEntity* требует реализации следующих методов:

initialise он вызывается, когда начинается моделирование или происходит рестарт; его основная задача . это инициализация переменных состояний;

getName возвращает имя объекта для вывода в заголовок столбца временной диаграммы;

getServices возвращает перечень действий, которые объект может совершить в данный момент; они выводятся в меню выбора для пользователя программы;

performService оповещает объект о выборе пользователя; объект выполняет действие и переходит к следующему состоянию;

receivePDU оповещает объект, о том что он получил PDU (Protocol Data Unit);

setPeer устанавливает объекту его собеседника., т.е. другой объект, с которым он обменивается сообщениями;

transmitPDU вызывается объектом для отправки PDU другому объекту.

Timeouts - это интерфейс для управления таймерами сообщений.

Пользователь может контролировать таймауты, используя меню. При выборе таймаута для сообщения пользователь сможет исследовать реакцию протокола на данную ситуацию. Зачастую такое исследование раскрывает наиболее интересные аспекты реализации протокола. В исходном коде протокола определено, где должны быть использованы таймауты. Как правило, это низкоуровневые, а не высокоуровневые протоколы. Протокол с таймаутами устанавливает и сбрасывает таймеры в определённых точках своего поведения. Так как моделирование происходит не в реальном времени, действительные значения таймаутов не моделируются. Объект протокола, который реализует интерфейс таймера, должен реализовать следующие методы:

hasTimer вызывается с параметром, значение которого тип PDU, для того чтобы определить поддерживает ли протокол таймауты для данного вида PDU

setTimer устанавливает объекту таймера статус для определенного PDU;

Объект ожидает запись этого значения и использует его для принятия решения о том, передать заново данные при таймауте или отменить подтверждение.

Базовый класс среды *Medium* описывает главные свойства коммуникационной среды. Хотя его функциональности достаточно для большинства протоколов, он может быть при необходимости наследован и расширен. Например, переупорядочивание сообщений в среде необходимо добавить для протоколов IP (Internet Protocol) и TCP (Transmission Control Protocol). Вообще, наследование разрешено для всей иерархии классов среды.

Объекты протокола и среды согласуются между собой с помощью обмена событиями *ProtocolEvent*:

send сервис пользователя посылает сообщение объекту протокола;

deliver объект протокола доставляет сообщение сервису пользователя;

transmit объект протокола посылает сообщение коммуникационной среде;

receive объект протокола получает сообщение от коммуникационной среды;

timeout заставляет объект протокола заново передать неподтвержденное сообщение;

lose теряет сообщение внутри коммуникационной среды;

fragment фрагментирует сообщение коммуникационной среды.

Вдобавок существует специальное событие *comment*. Оно используется, для того чтобы объяснить, почему некоторое действие произошло. Например, объект может указать, что его окно сейчас разблокировано или он заново передает сообщение после таймаута. Эти комментарии не несут полезной информации для пользователя, но они полезны для понимания того, что делает протокол. Комментарии генерируются кодом, определенным в описании модели протокола.

Базовые классы

Базовый класс временной диаграммы *SequenceDiagram* описывает основные графические представления протоколов. В данной реализации Jasper существует только подкласс *TimeSequenceDiagram*. Однако структура моделирующей системы позволяет использовать другие графические представления. Они могут быть использованы как вместе, так и порознь.

Класс *Simulator* полностью контролирует моделирование протоколов. Так как все протоколы реализуют один и тот же интерфейс, то нет необходимости знать подробности реализации конкретного протокола или коммуникационной среды.

Базовый класс *PDU* описывает сообщения протоколов. Формат *PDU* важен на практике, но для моделирования может быть абстрагирован до поля класса. Базовый класс описывает тип сообщения, объект отправитель, объект получатель, номер в последовательности и SDU (Service Data Unit или сообщение пользователя). Так как реальные данные не важны для понимания работы протокола, то они игнорируются при моделировании. Для протоколов, которые обрабатывают не упорядоченность и фрагментацию сообщений, необходимо работать с данными на абстрактном уровне.

Содержимое данных может быть идентифицировано метками, например, D0 или D1, или по позиции данных в потоке. Протоколы со специализированными PDUs (например, TCP) наследуются от базового класса *PDU*.

Пример моделирования

В виде примера рассмотрим рисунок 2, на котором изображены основные классы, используемые для моделирования протокола TCP. Закрашенные классы . это классы моделирующей системы, остальные принадлежат реализации модели протокола TCP. На верхнем уровне экземпляр объекта *Protocol*, например, *TCP* играет роль менеджера протокола.

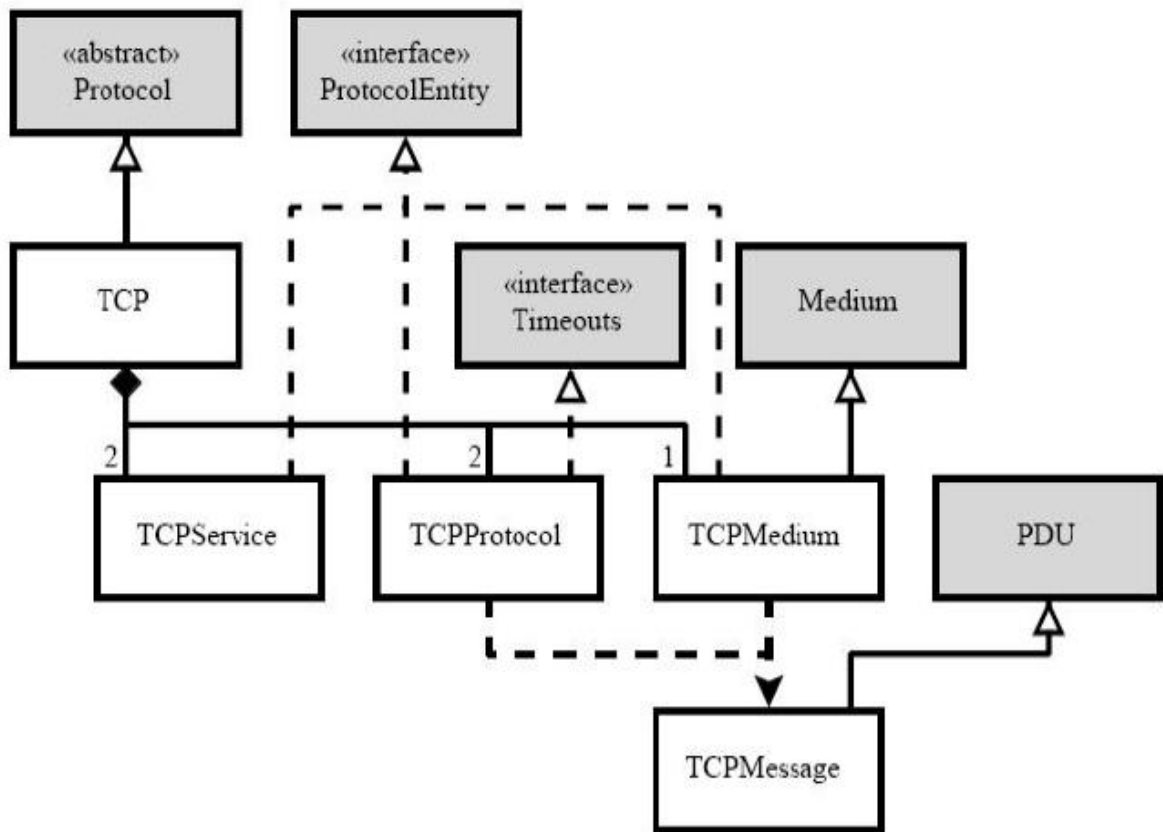


Рисунок 2 - Диаграмма классов, использующихся для моделирования TCP

Он отвечает за создание и согласование наследованных от него объектов протокола. Объекты поддерживают соответствие интерфейсу *ProtocolEntity*. Для протокола *TCP* существуют пользователи (сервис пользователей *TCPService*), объект протокола (*TCPProtocol*), и среда (*TCPMedium*).

Объект также может реализовывать интерфейс таймаутов *Timeouts*, если ему необходимо, чтобы таймеры работали на сообщениях протокола. *TCPMedium* расширяет основной класс среды *Medium*. Оба класса *TCPProtocol* и *TCPMedium* используют класс *TCPMessage* - подкласс базового класса *PDU*.

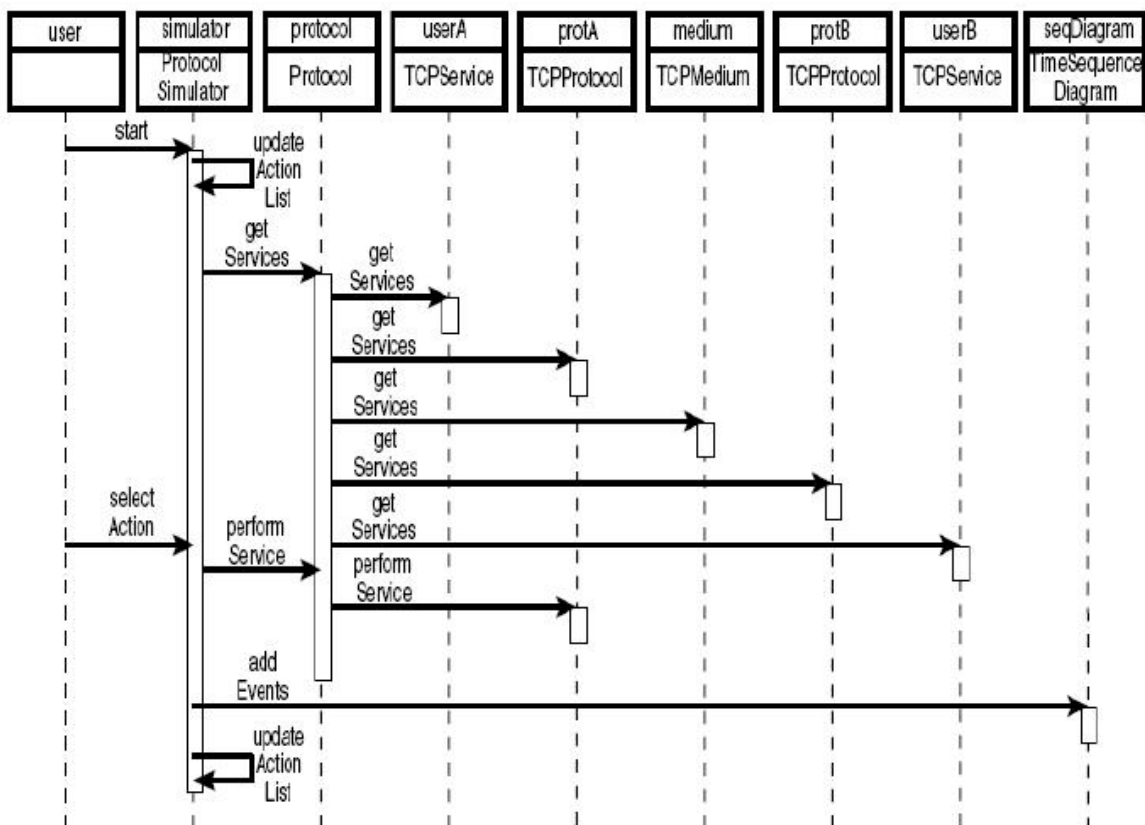


Рисунок 3 - Взаимодействие моделируемых объектов

Взаимодействие моделируемых объектов показано на диаграмме на рисунке 3. В момент окончания каждого соединения существуют экземпляры объектов *TCPService* (представляет интерфейс пользователя) and *TCPProtocol* (представляет протокол). *TCPMedium* представляет коммуникационную среду. Объекты классов *ProtocolSimulator* и *Protocol* используют метод *getServices*, для того чтобы спросить у объектов какие действия (выбор пользователя) возможны в данный момент. При выборе пользователем одного из возможных действий, происходит вызов метода *performService* для реализации выбора с соответствующим объектом. В течение выполнения протокол может вызвать события, например, фрагментацию или доставку сообщения. Все важные действия отсылаются методом *addEvents* экземпляру объекта *TimeSequenceDiagram* для отображения на экране. На рисунке 4 изображено моделирование протокола TCP.

Параметры моделирования, такие как размер пользовательского сообщения и размер получаемого протоколом окна устанавливаются в верхней части окна. Когда пользователь нажимает на кнопку *Change Values*, параметры моделирования считываются и проверяются с помощью кода на JavaScript на самой веб-странице. После этого параметры передаются апплету, который выполняется в этом же окне. В апплете в левом нижнем углу расположены кнопки, с помощью которых пользователь может контролировать моделирование. Кнопка *Undo* отменяет последний шаг, что в свою очередь может быть отменено кнопкой *Redo* (она не активна, пока пользователь ничего не отменял). Кнопка *Run* запускает моделирование в случайном режиме. Кнопка *Clear* заново запускает моделирование. Если Jasper запущен как приложение, то у него появляются дополнительные возможности, такие как сохранение, загрузка и печать сценария моделирования. Кнопка *Load* используется для загрузки сценария моделирования, созданного до этого с помощью кнопки *Save*. Кнопка *Print* используется для печати текущего процесса моделирования. С права от кнопок находится меню. Например, пользователь может кликнуть на элементе управления *.Client: Send 100 octets (Push)*. для выполнения данного действия. Временная диаграмма после этого обновиться и при необходимости перейдет к последнему событию. На рисунке 3 изображен процесс моделирования протокола TCP, на котором хорошо видны все элементы управления (приложение запущено как апплет).

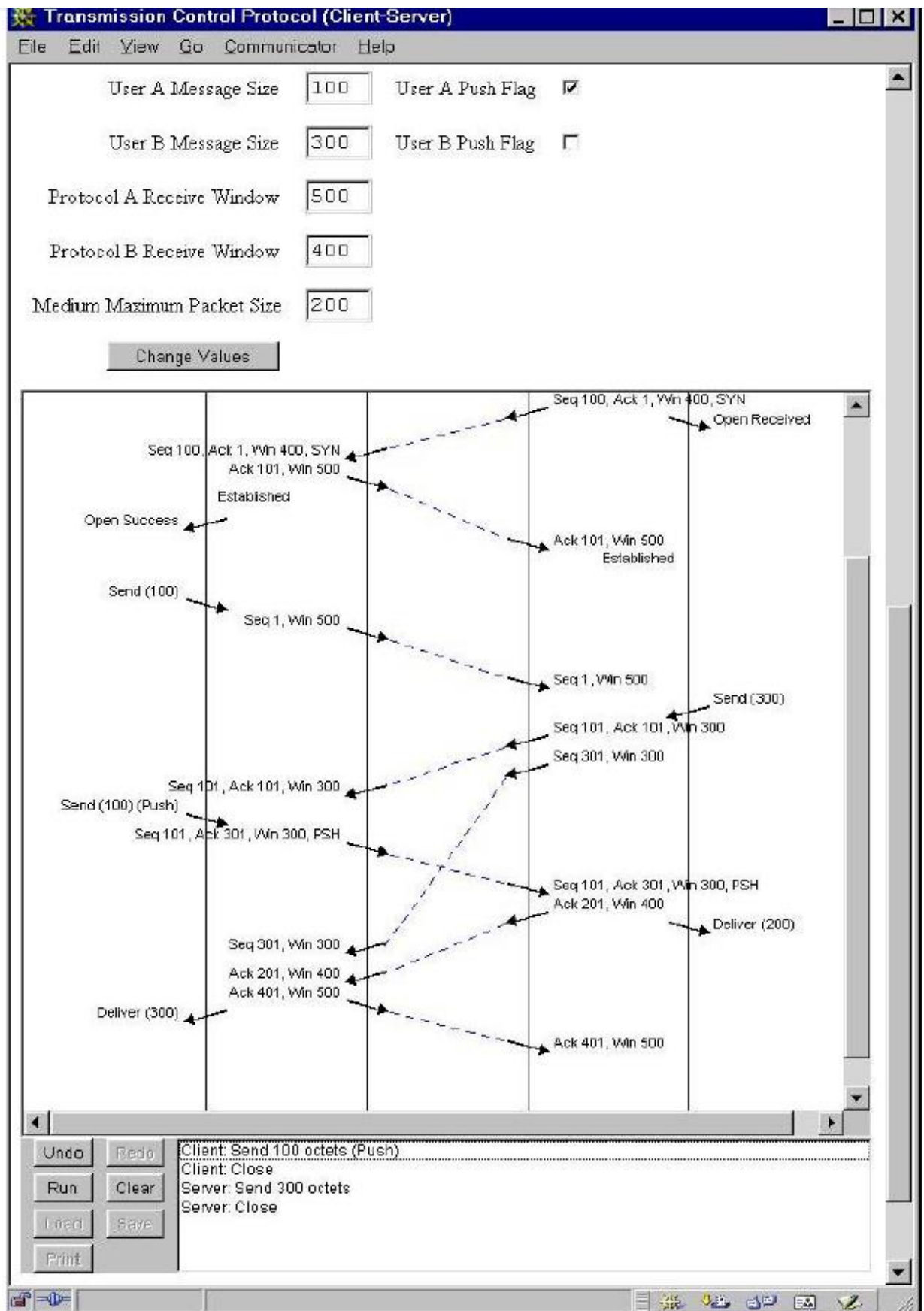


Рисунок 4 - Моделирование протокола TCP

Рассмотрим структуру пакета TCP

Таблица 1.1 Значения бит поля *флаги*

| Обозначение битов (слева на право) поля <i>флаги</i> | Значение бита, если он равен 1 |
|--|--|
| Urg | Флаг важной информации, поле <i>Указатель важной информации</i> имеет смысл, если urg=1. |
| Ack | Номер октета, который должен прийти следующим, правилен. |
| Psh | Этот сегмент требует выполнения операции push. Получатель должен передать эти данные прикладной программе как можно быстрее. |
| rst | Прерывание связи. |
| syn | Флаг для синхронизации номеров сегментов, используется при установлении связи. |
| fin | Отправитель закончил посылку байтов. |

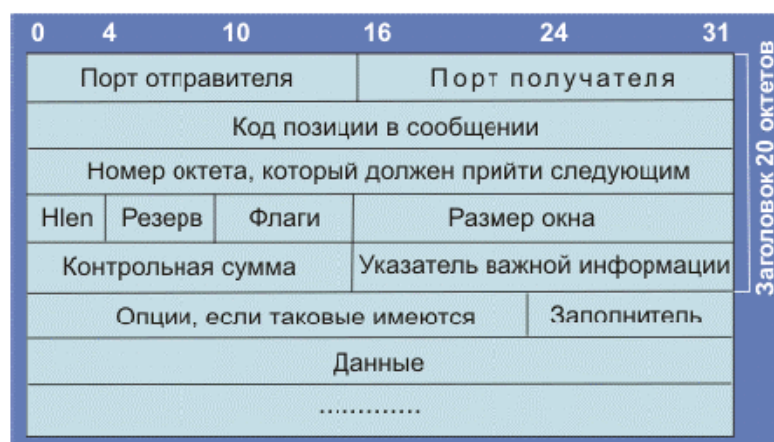


Рисунок 5 - Формат TCP-сегмента

Поле *опции* зарезервировано на будущее и в заголовке может отсутствовать, его размер переменен и дополняется до кратного 32-бит с помощью поля *заполнитель*.

Поле *данные* в TCP-сегменте может и отсутствовать, характер и формат передаваемой информации задается исключительно прикладной программой, максимальный размер этого поля составляет в отсутствии опций 65495 байт.

TCP является протоколом, который ориентируется на согласованную работу ЭВМ и программного обеспечения партнеров, участвующих в обмене информацией. Установление связи клиент-сервер осуществляется в три этапа:

1. Клиент посылает SYN-сегмент с указанием номера порта сервера, который предлагается использовать для организации канала связи (active open).

2. Сервер откликается, посылая свой SYN-сегмент, содержащий идентификатор (ISN - initial sequence number). Начальное значение ISN не равно нулю. Процедура называется *passive open*.

3. Клиент отправляет подтверждение получения syn-сегмента от сервера с идентификатором равным $ISN(\text{сервера})+1$.

Стандартная процедура установления связи представлена на рисунке 6 (под словом стандартная. подразумевается отсутствие каких-либо отклонений от штатного режима, например, одновременного открывание соединения со стороны сервера и клиента). Если же соединение одновременно иницируется клиентом и сервером, в конечном итоге будет создан только один виртуальный канал.

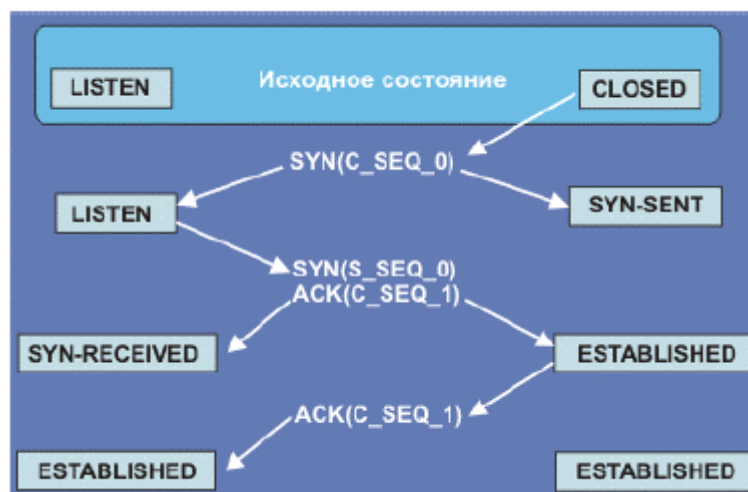


Рисунок 6 - Алгоритм установления связи

Префикс **S** на рисунке указывает на сервер, а **C** . на клиента.

Параметры в скобках обозначают относительные значения ISN. После установления соединения $ISN(S) = s_seq_1$, а $ISN(C) = c_seq_1$.

Каждое соединение должно иметь свой неповторимый код ISN. Для реализации режима соединения прикладная программа на одном конце канала устанавливается в режим пассивного доступа ("passive open"), а операционная система на другом конце ставится в режим активного доступа ("active open"). Протокол TCP предполагает реализацию 11 состояний (established, closed, listen, syn_sent, syn_received и т.д.; см. также RFC-793), переход между которыми строго регламентирован. Машина состояний для протокола TCP может быть описана диаграммой, представленной на рис. 1.3. Здесь состояние closed является начальной и конечной точкой последовательности переходов. Каждое соединение стартует из состояния closed. Из диаграммы машины состояний видно, что ни одному из состояний не поставлен в соответствие какой-либо таймер. Это означает, что машина состояний TCP может оставаться в любом из состояний сколь угодно долго. Исключение составляет keep-alive таймер, но его работа является опциональной, а время по умолчанию составляет 2 часа. Это означает, что машина состояния может оставаться 2 часа без движения. В случае, когда два компьютера (С и S) попытаются установить связь друг с другом одновременно, реализуется режим simultaneous connection (RFC-793). Оба компьютера посылают друг другу сигналы SYN. При получении этого сигнала партнеры посылают отклики SYN+ACK. Оба компьютера должны обнаружить, что SYN и SYN+ACK относятся к одному и тому же соединению. Когда С и S обнаружат, что SYN+ACK соответствует посланному ранее SYN, они выключат таймер установления соединения и перейдут непосредственно в состояние syn_rcvd. В состоянии established пакет будет принят сервером, если его ISN лежит в пределах s_ack, s_ack+s_wind (s_wind - ширина окна для сервера). Аналогичный диапазон ISN для клиента выглядит как: c_ack, c_ack+c_wind (c_wind - ширина окна для клиента). c_wind и s_wind могут быть не равны. *Пакеты, для которых эти условия не выполняются, будут отброшены.* ISN - идентификатор пакета, посылаемого клиентом (С) или сервером (S). Клиент, послав syn

серверу S, переходит в состояние `syn_sent`. При этом запускается таймер установления соединения. Как при установлении соединения, так и при его разрыве приходится сталкиваться с *проблемой двух армий*. Представим себе, что имеется две армии А и Б, причем Б больше по численности чем А. Армия Б разделена на две части, размещенные по разные стороны от армии А. Если две части армии Б одновременно нападут на армию А, победа гарантирована. В то же время нападение на А одной из частей армии Б обрекает ее на поражение. Но как обеспечить одновременность? Здесь предполагается, что радио еще не изобретено и передача сообщений осуществляется вестовыми, которые в нашем случае могут быть перехвачены врагом. Как убедиться, что вестовой дошел? Первое, что приходит в голову, это послать другого вестового с подтверждением. Но он также с некоторой вероятностью может быть перехвачен. А отправитель не будет знать, дошел ли он. Ведь если сообщение перехвачено, отправитель первичного запроса не выдаст команды на начало, так как не уверен, дошло ли его первое сообщение. Возникает вопрос, существует ли алгоритм, который бы гарантировал надежность синхронизации решений путем обмена сообщениями при ненадежной доставке? Повысит ли достоверность увеличение числа обменов между партнерами? Ответом на этот вопрос будет - нет, не существует. В этом читатель, порассуждав логически, может убедиться самостоятельно. Не трудно видеть, что схожие проблемы возникают в любом протоколе, работающем через установление соединения. Чаще всего эта проблема решается путем таймаутов и повторных попыток. Сервер, получив **SYN**, откликается посылкой другого SYN. Когда С получает SYN от S (но не получает ACK, например, из-за его потери или злого умысла), он предполагает, что имеет место случай одновременного открытия соединения. В результате он посылает **syn_ack**, отключает таймер установления соединения и переходит в состояние `syn_received`. Сервер получает `syn_ack` от С, но не посылает отклика. Тогда С ожидает получения `syn_ack` в состоянии `syn_received`.

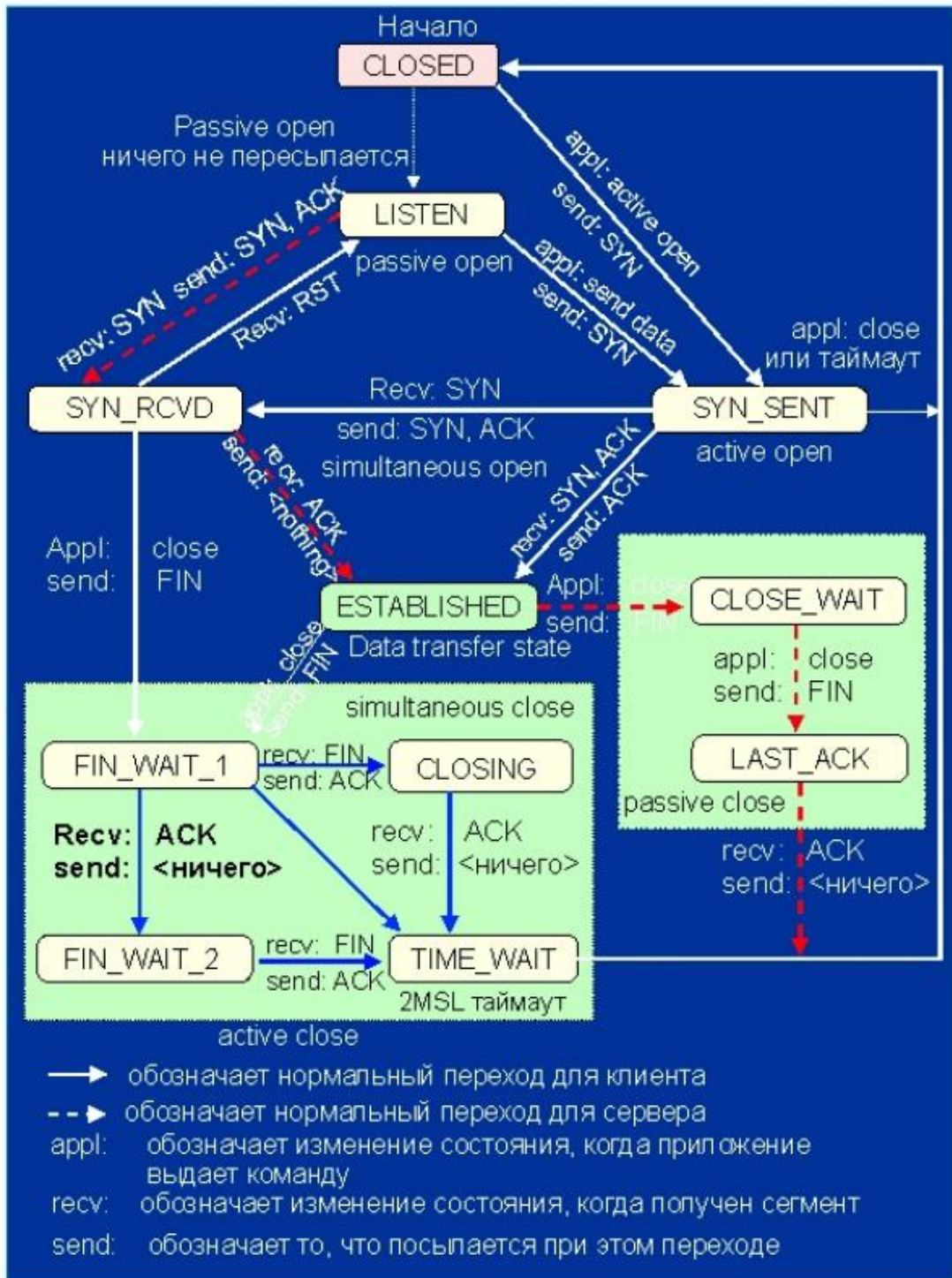


Рисунок 7 - Машина состояний для протокола TCP (W.R. Stevens, TCP/IP Illustrated, V1. Addison-Wesley publishing company. 1993.)

Так как время пребывания в этом состоянии не контролируется таймером, С может остаться в состоянии `syn_received` вечно. Из-за того, что переходы из состояния в состояние не всегда четко определены, протокол TCP допускает и другие виды атак. Хотя TCP-соединение является полнодуплексным, при рассмотрении процесса разрыва связи проще его

рассматривать как два полудуплексных канала, каждый из которых ликвидируется независимо. Сначала инициатор разрыва посылает сегмент с флагом **FIN**, сообщая этим партнеру, что не намерен более что-либо передавать. Когда получение этого сегмента будет подтверждено (ACK), данное направление передачи считается ликвидированным. При этом передача информации в противоположном направлении может беспрепятственно продолжаться. Когда партнер закончит посылку данных, он также пошлет сегмент с флагом **FIN**. По получении отклика ACK виртуальный канал считается окончательно ликвидированным. Таким образом, для установления связи требуется обмен тремя сегментами, а для разрыва - четырьмя. Но протокол допускает совмещение первого ACK и второго **FIN** в одном сегменте, сокращая полное число закрывающих сегментов с четырех до трех.

Задание к лабораторной работе

Промоделировать протокол TCP с заданными параметрами с помощью системы JASPER, рассмотрев не менее 5 ветвей моделирования, проанализировать результаты. Параметры протокола приведены ниже в таблице 1.2.

Таблица 1.2 – Варианты заданий

| № | Размер сообщения пользователя А | Размер окна пользователя А | Размер сообщения пользователя В | Размер окна пользователя В | Максимальный размер сообщения в среде | Вид моделирования |
|----|---------------------------------|----------------------------|---------------------------------|----------------------------|---------------------------------------|-------------------|
| 1 | 50 | 250 | 25 | 200 | 200 | Client-Server |
| 2 | 100 | 300 | 50 | 300 | 200 | Client-Server |
| 3 | 150 | 350 | 75 | 400 | 200 | Client-Server |
| 4 | 200 | 400 | 100 | 500 | 200 | Client-Server |
| 5 | 250 | 450 | 125 | 600 | 200 | Client-Server |
| 6 | 50 | 250 | 25 | 300 | 300 | Peer-to-Peer |
| 7 | 100 | 300 | 50 | 400 | 300 | Peer-to-Peer |
| 8 | 150 | 350 | 75 | 500 | 300 | Peer-to-Peer |
| 9 | 200 | 400 | 100 | 600 | 300 | Peer-to-Peer |
| 10 | 250 | 450 | 125 | 700 | 300 | Peer-to-Peer |
| 11 | 150 | 250 | 50 | 400 | 300 | Client-Server |
| 12 | 200 | 300 | 75 | 500 | 300 | Client-Server |
| 13 | 150 | 350 | 100 | 600 | 300 | Client-Server |
| 14 | 250 | 400 | 125 | 700 | 300 | Client-Server |
| 15 | 300 | 450 | 150 | 800 | 300 | Client-Server |
| 16 | 150 | 250 | 25 | 500 | 300 | Peer-to-Peer |
| 17 | 200 | 300 | 50 | 600 | 300 | Peer-to-Peer |
| 18 | 250 | 350 | 75 | 700 | 300 | Peer-to-Peer |
| 19 | 300 | 400 | 100 | 800 | 300 | Peer-to-Peer |
| 20 | 350 | 450 | 125 | 900 | 300 | Peer-to-Peer |
| 21 | 50 | 250 | 50 | 400 | 400 | Client-Server |
| 22 | 100 | 300 | 75 | 500 | 400 | Client-Server |
| 23 | 150 | 350 | 100 | 600 | 400 | Client-Server |
| 24 | 200 | 400 | 125 | 700 | 400 | Client-Server |
| 25 | 250 | 450 | 150 | 800 | 400 | Client-Server |

Порядок выполнения работы

1. Внимательно ознакомьтесь с методическим материалом
2. Промоделируйте протокол с заданными параметрами не менее 5 раз по различным ветвям событий
3. Проведите сравнительный анализ полученных результатов

Содержание отчета

1. Название и цель лабораторной работы.
2. Постановка задачи.

3. Описание модели протокола
4. Графические результаты моделирования.
5. Выводы.

Контрольные вопросы

1. Когда применяют протокол TCP?
2. Какие флаги могут использоваться в заголовке TCP-сегмента и для чего?
3. Перечислите и поясните этапы установления соединения между клиентом и сервером.
4. Сколько и каких состояний существует в протоколе TCP?
5. Какая проблема возникает как при установлении соединения, так и при его разрыве?

Лабораторная работа №7

Тема: Настройка контроллера домена и установка дополнительных ролей.

Цель: Ознакомиться с принципами установки и настройки серверных операционных систем семейства Windows, изучить роли сервера.

Методические указания к выполнению работы:

Windows Server является наиболее безопасной, надежной, отказоустойчивой и удобной в управлении ОС.

Установка и настройка Windows Server и Active Directory.

При настройке сервера различают:

1) Типовая настройка для первого сервера (Typical Configuration For A First Server), мастер сделает сервер контроллером нового домена, установит службы Active Directory и при необходимости службы DNS (Domain Name Service), DHCP (Dynamic Host Configuration Protocol) и RRAS (Routing And Remote Access).

2) Особая конфигурация (Custom Configuration)

Возможно с помощью мастера настроить следующие роли:

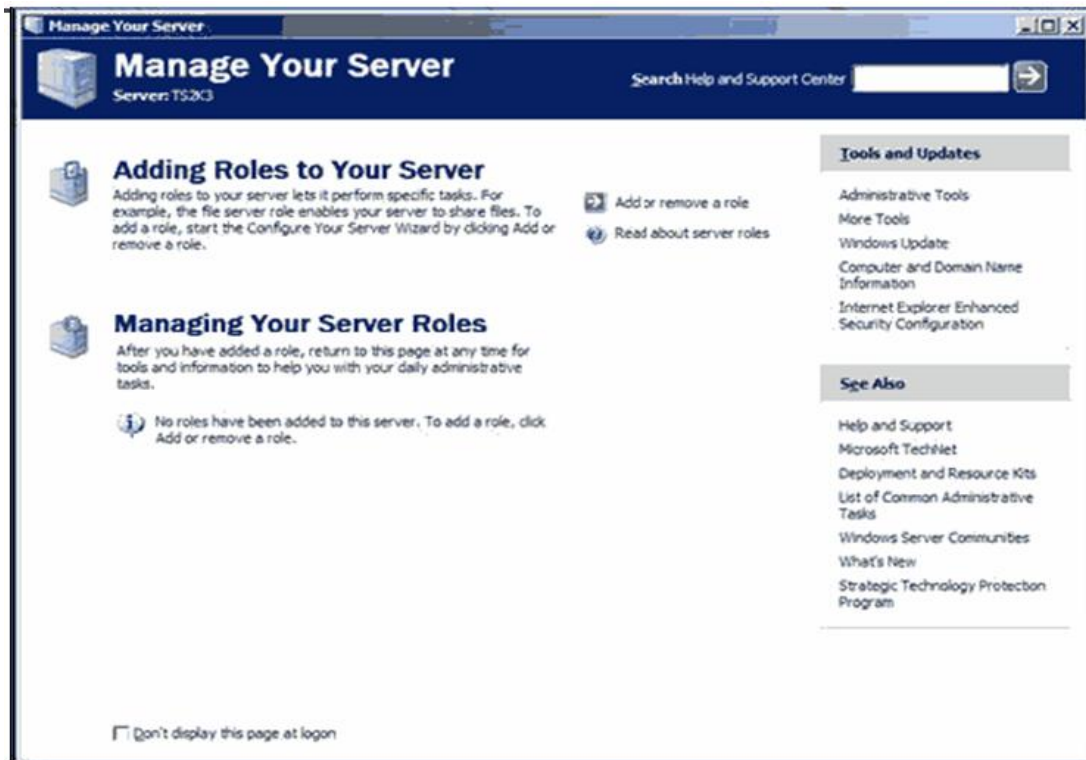


Рисунок 1 - Установка ролей

Роль - это функция сервера (например, почтовый сервер, контроллер домена).

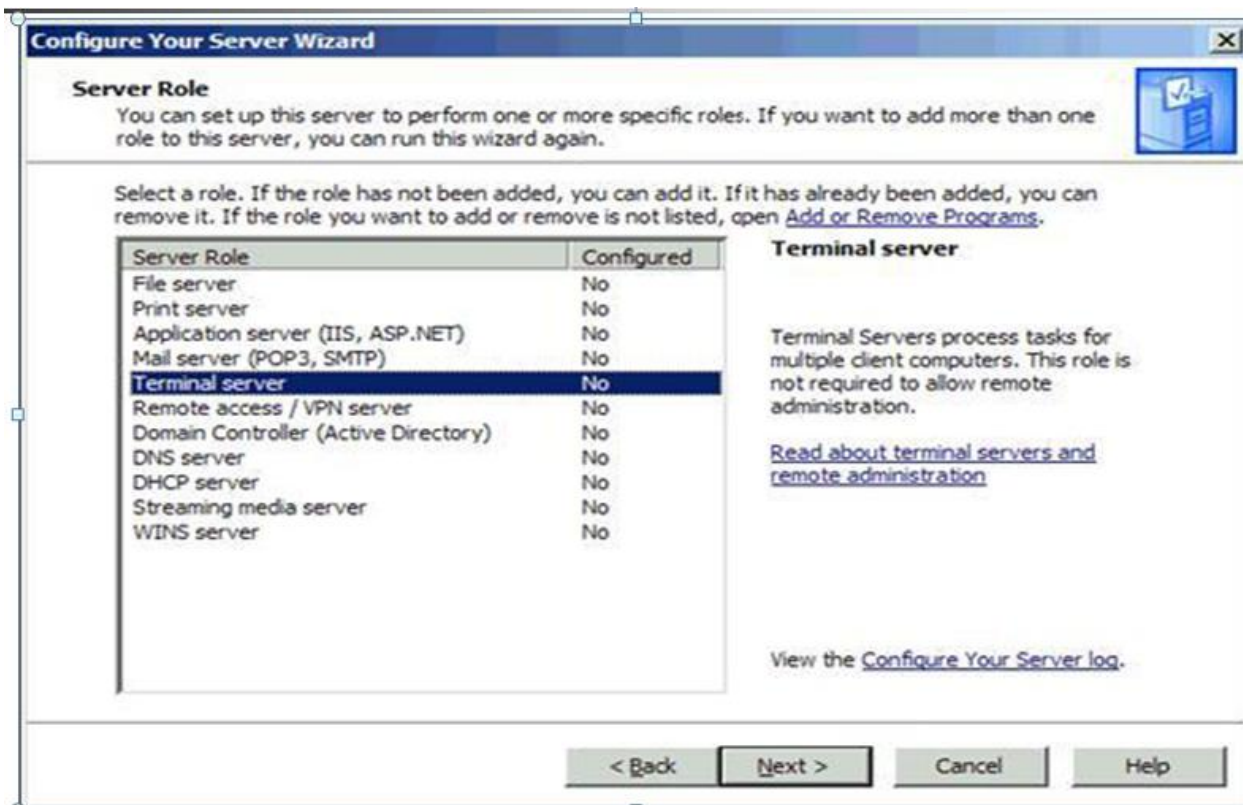


Рисунок 2 - Выбор роли сервера

А) Файловый сервер (File Server). Обеспечивает централизованный доступ к файлам и каталогам для пользователей, отделов и организации в целом.

б) Сервер печати (Print Server). Обеспечивает централизованное управление печатающими устройствами, предоставляя клиентским компьютерам доступ к общим принтерам и их драйверам.

в) Application Server IIS, ASP.NET (Сервер приложений IIS, ASP.NET).

Серверы приложений часто конфигурируют включая следующее:

- Слияние ресурсов (Resource pooling);
- Управление распределенными транзакциями;
- Встроенная защита;
- Отказоустойчивость.

г) Mail Server POP3, SMTP (Почтовый сервер POP3, SMTP). Устанавливает POP3 и SMTP, чтобы сервер мог выступать в роли почтового сервера для клиентов POP3.

д) Сервер терминалов (Terminal Server). Позволяет множеству пользователей с помощью клиентского ПО Службы терминалов (Terminal Services) или Дистанционное управление рабочим столом (Remote Desktop) подключаться к приложениям и ресурсам сервера.

е) Сервер удаленного доступа или VPN-сервер (Remote Access/VPN Server). Обеспечивает маршрутизацию по нескольким протоколам и службы удаленного доступа для коммутируемых, локальных (LAN) и глобальных (WAN) вычислительных сетей.

Виртуальная частная сеть (virtual private network, VPN) обеспечивает безопасное соединение пользователя с удаленными узлами через стандартные интернет-соединения.

ж) Контроллер домена Active Directory (Domain Controller Active Directory). Предоставляет:

- службы каталогов клиентам сети. Этот вариант позволяет создать контроллер нового или существующего домена и установить DNS.

з) DNS Server (DNS сервер). Обеспечивает разрешение имен узлов: DNS-имена преобразуются в IP-адреса (прямой поиск) и обратно (обратный поиск).

и) DHCP-сервер (DHCP Server). Предоставляет службы автоматического выделения IP адресов клиентам, настроенным на динамическое получение IP-адресов.

к) Сервер потоков мультимедиа (Streaming Media Server). Предоставляет службы WMS

(Windows Media Services), которые позволяют серверу передавать потоки мультимедийных данных в интрасети или через Интернет.

Служба каталогов Active Directory

Модель домена характеризуется единым каталогом ресурсов предприятия — Active Directory, которому доверяют все системы безопасности, принадлежащие домену.

Служба Active Directory, играет роль идентификационного хранилища и сообщает «кто есть кто» в этом домене.

Active Directory — коллекция файлов, включая журналы транзакций и системный том (Sysvol), содержащий сценарии входа в систему и сведения о групповой политике.

Это службы, поддерживающие и использующие БД, включая:

- протокол LDAP (Lightweight Directory Access Protocol),
- протокол безопасности Kerberos,
- процессы репликации,
- и службу FRS (File Replication Service).

Контроллер домена назначается Мастером установки Active Directory.

После того как сервер становится контроллером домена, на нем хранится копия (реплика) Active Directory, и изменения БД на любом контроллере реплицируются на все остальные контроллеры домена.

Назначение службы каталогов Active Directory

Каталог (справочник) может хранить различную информацию, относящуюся к пользователям, группам, компьютерам, сетевым принтерам, общим файловым ресурсам.

Служба каталогов Active Directory предоставляя следующие возможности:

- Единая регистрация в сети;
- Безопасность информации.
- Централизованное управление.
- Администрирование с использованием групповых политик.
- Интеграция с DNS.
- Расширяемость каталога.
- Масштабируемость.
- Репликация информации.
- Гибкость запросов к каталогу.
- Стандартные интерфейсы программирования.

Учетная запись пользователя является примером объекта.

Active Directory не может существовать без домена и наоборот.

Домен — это основная административная единица службы каталогов.

База данных домена содержит:

- учетные записи пользователей;
- учетные записи групп;
- учетные записи компьютеров.

Контроллеры домена

Контроллеры домена — специальные серверы, которые хранят соответствующую данному домену часть базы данных Active Directory.

Основные функции контроллеров домена:

- 1) хранение БД Active Directory;
- 2) синхронизация изменений в AD;

3) аутентификация пользователей.

Рекомендуется в каждом домене устанавливать не менее двух контроллеров домена

Если несколько моделей доменов совместно используют непрерывное пространство имен DNS, они образуют логические структуры, называемые деревьями (tree).

Дочерний домен автоматически устанавливает двухсторонние транзитивные доверительные отношения с родительским доменом (Ресурсы одного из доменов могут быть доступны пользователям других доменов.)

Корпорация Microsoft рекомендует строить Active Directory в виде одного домена.

Домены Active Directory с разными корневыми доменами образуют несколько деревьев. Они объединяются в самую большую структуру Active Directory — лес (forest).

Первый домен, создаваемый в лесе, считается его корневым доменом, в корневом домене хранится схема AD.

При управлении деревьями и лесом нужно помнить два очень важных момента:

1) первое созданное в лесе доменов дерево является корневым деревом, первый созданный в дереве домен называется корневым доменом дерева (tree root domain);

2) первый домен, созданный в лесе доменов, называется корневым доменом леса (forest root domain), данный домен не может быть удален (он хранит информацию о конфигурации леса и деревьях доменов, его образующих).

Организационные подразделения (ОП)

Организационные подразделения (Organizational Units, OU) — контейнеры внутри AD, которые создаются для объединения объектов в целях делегирования административных прав и применения групповых политик в домене.

ОП существуют только внутри доменов и могут объединять только объекты из своего домена.

Глобальный каталог

Если доменов несколько, приобретает важность компонент Active Directory, называемый глобальным каталогом (global catalog): он предоставляет информацию об объектах, расположенных в других доменах леса.

Глобальный каталог является перечнем всех объектов, которые существуют в лесе Active Directory.

Физическая структура Active Directory служит для связи между логической структурой AD и топологией корпоративной сети.

Основные элементы физической структуры Active Directory — **контроллеры домена и сайты**.

Сайт — группа IP-сетей, соединенных быстрыми и надежными коммуникациями.

Назначение сайтов — управление процессом репликации между контроллерами доменов и процессом аутентификации пользователей.

Структура сайтов никак не зависит от структуры доменов.

Один домен может быть размещен в нескольких сайтах, и в одном сайте могут находиться несколько доменов

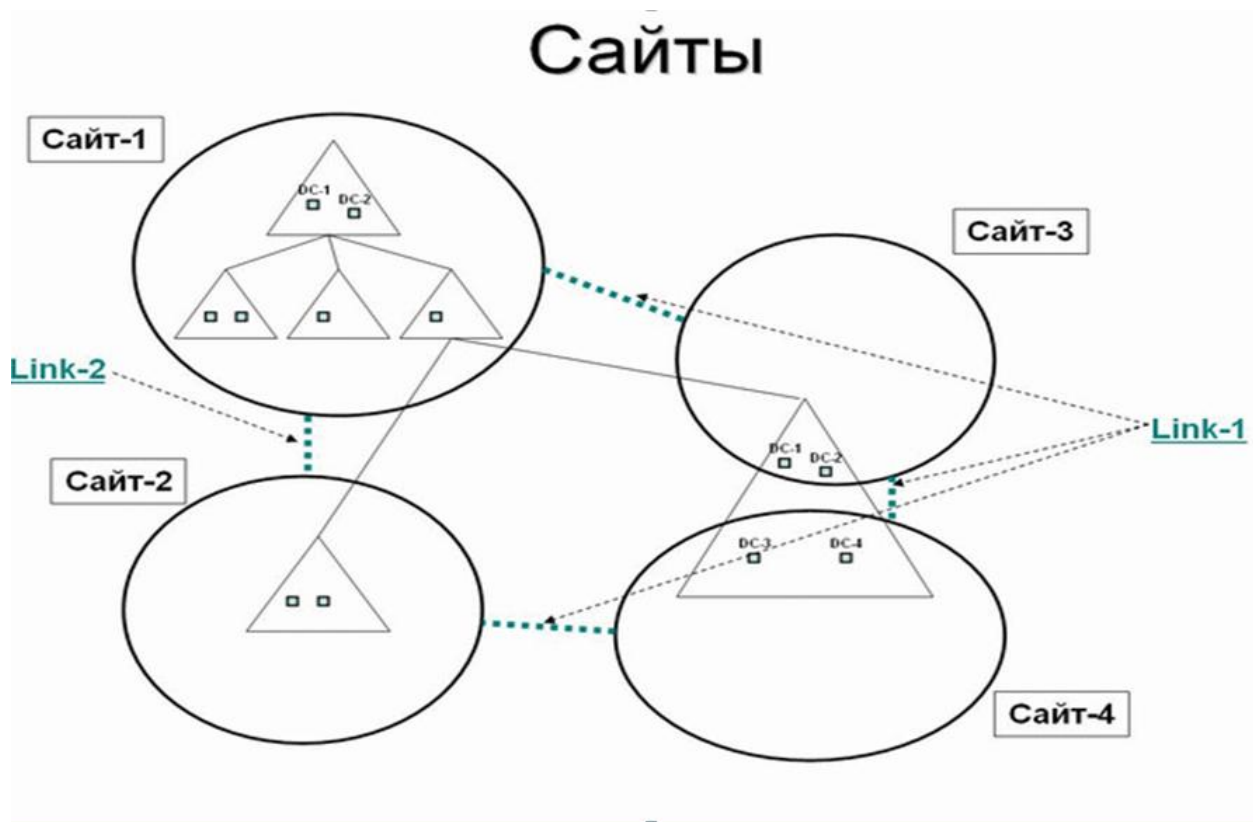


Рисунок 3 - Сайты и домены

Механизмы репликации изменений в AD внутри сайта и между сайтами различные.

Внутри сайта контроллеры домена соединены линиями с высокой пропускной способностью. Поэтому репликация между контроллерами производится каждые 5 минут, данные при передаче не сжимаются, для взаимодействия между серверами используется технология вызова удаленных процедур (RPC).

Рекомендуется в каждом сайте установить как минимум один контроллер домена.

В каждом сайте необходимо также размещать как минимум один сервер глобального каталога.

Пользователи сети (в том числе компьютеры и сетевые службы) используют серверы глобального каталога для поиска объектов.

Репликацию выполняет компонента служб каталогов, называемая Knowledge Consistency Checker, или КСС, вариант перевода данного термина — "наблюдатель показаний целостности").

топологию репликации можно с помощью административной консоли "Active Directory - сайты и службы".

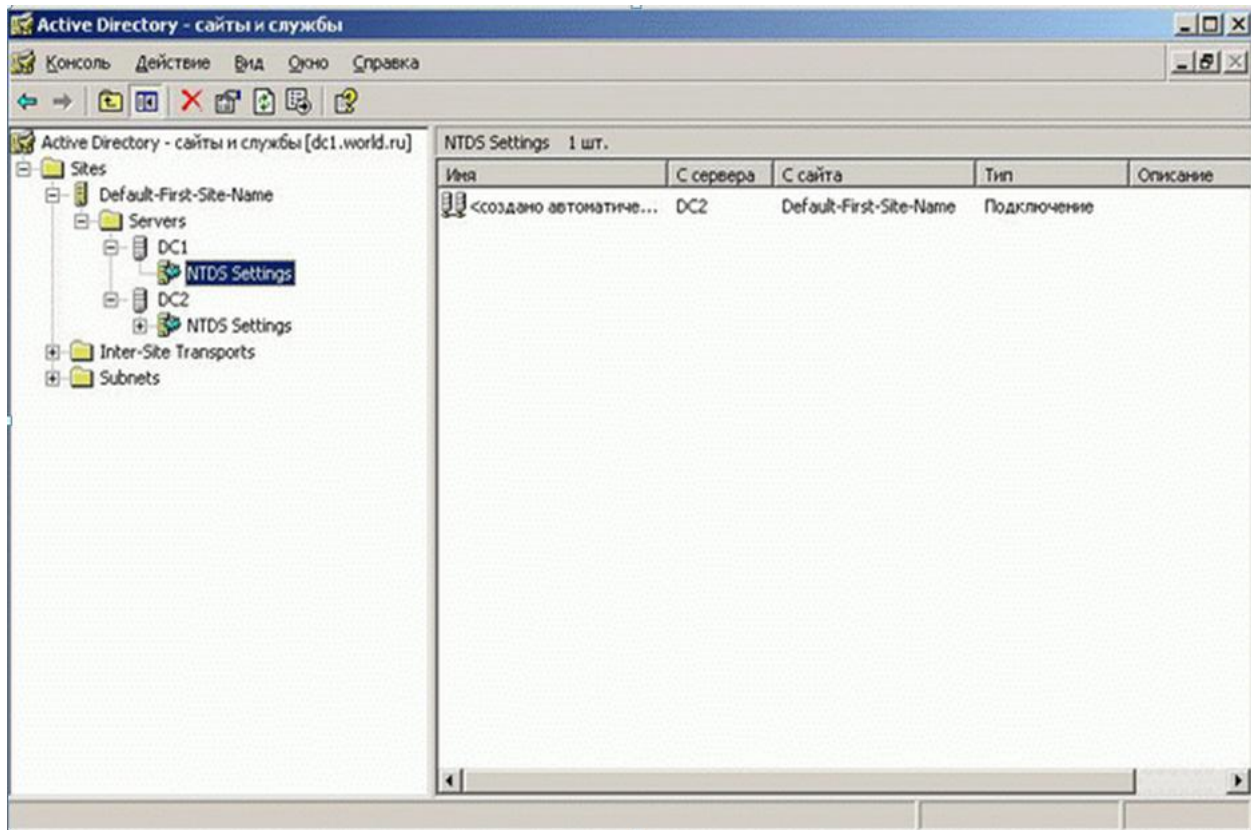


Рисунок 4 - Репликация внутри сайта

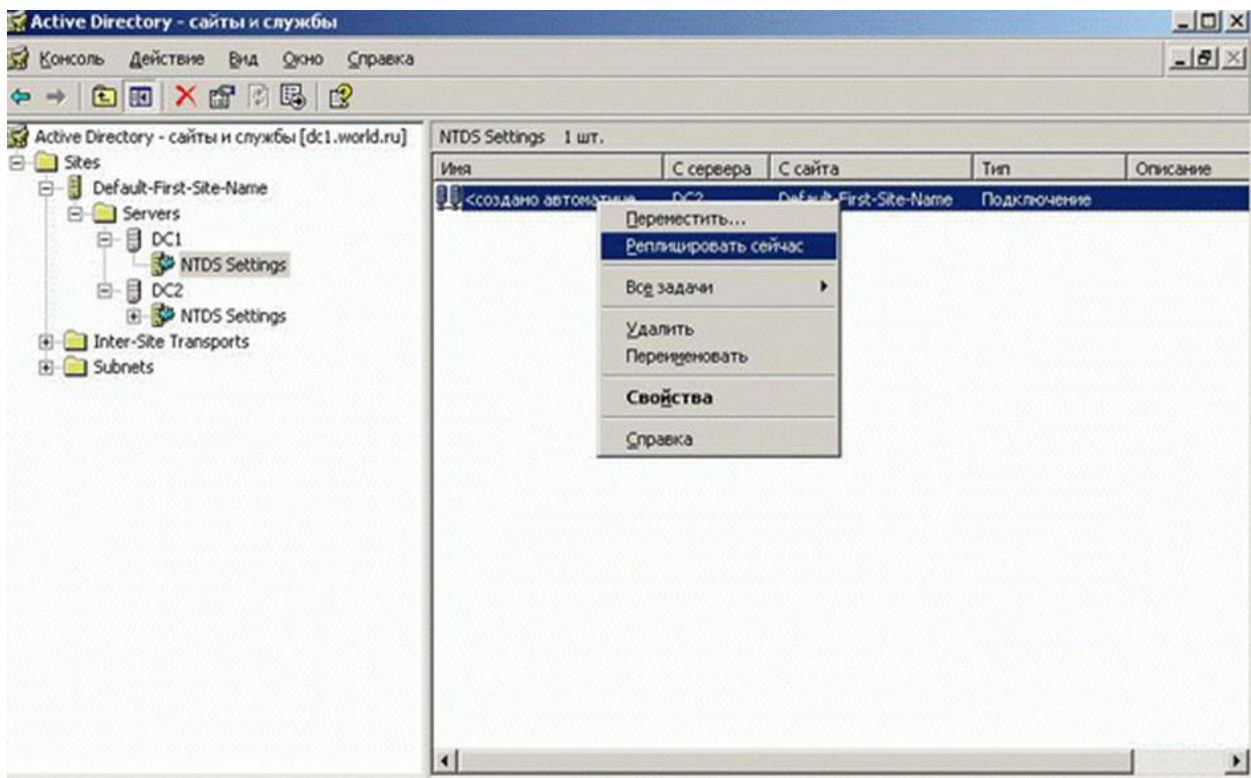


Рисунок 5 - "Репликация сейчас"

Задание к лабораторной работе:

Выполнить следующие действия при установке и настройке Windows Server.

- 1) Добавление ролей
 - Создание контроллера домена;
 - DNS Server;
 - DHCP- сервер;
- 2) Создание домена с именем типа `pmi.local`
- 3) Работа с организационными подразделениями
 - создание организационных единиц
 - именованние объектов;
 - использование групповой политики - создание объекта групповой политики (Group Policy Object, GPO)
- 4) Создание сайта (минимальные настройки, изучение возможностей – репликация внутри сайта – основные параметры репликации)
- 5) Выбор режима функционирования домена. Обоснование выбора.
- 6) Серверы Глобального каталога и Хозяева операций (Просмотреть текущих владельцев ролей с помощью административных консолей – попробовать, будет ли возможность работать с административными консолями).

Требования к отчету:

В отчет включить все скриншоты по установке и настройке контроллера домена и дополнительных ролей сервера. Кратко пояснить выполняемые действия.

Контрольные вопросы:

- 1) Что такое контроллер домена?
- 2) Какие роли сервера Вы знаете?
- 3) Что такое сайт?
- 4) Может ли быть 2 сайта в одном домене?

- 5) Может ли быть в одном сайте 2 домена?
- 6) Что такое сервер глобального каталога?
- 7) Назначение репликации.
- 8) Для чего можно использовать групповые политики?
- 9) Что такое организационное подразделение?
- 10) Правила именования объектов?

Лабораторная работа №8

Тема: Планирование AD для своего предприятия. Управление объектами через консоль Active Directory.

Цель: Изучить вопросы планирования AD и применить эти знания для проектирования AD для небольшого предприятия. Изучить основные утилиты для управления объектами через консоль Active Directory.

Методические указания к лабораторной работе:

При планировании AD необходимо учитывать следующие моменты:

- 1) тщательный выбор имен доменов верхнего уровня;
- 2) качество коммуникаций в компании (связь между отдельными подразделениями и филиалами);
- 3) организационная структура компании;
- 4) количество пользователей и компьютеров в момент планирования;
- 5) прогноз темпов роста количества пользователей и компьютеров.

При планировании имен доменов верхнего уровня можно использовать различные стратегии и правила. В первую очередь необходимо учитывать вопросы интеграции внутреннего пространства имен и пространства имен сети Интернет — т.к. пространство имен AD базируется на пространстве имен DNS, при неправильном планировании могут возникнуть проблемы с безопасностью, а также конфликты с внешними именами.

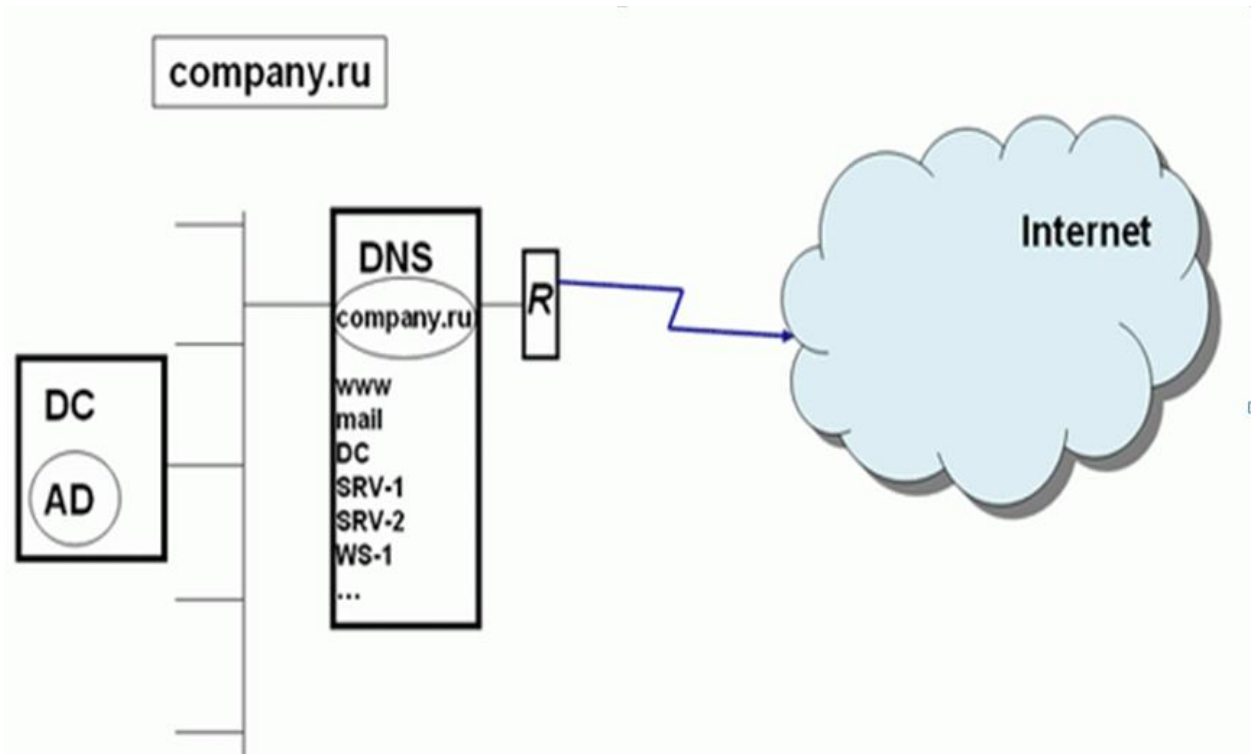


Рисунок 1 - Планирование пространства имен AD - основные варианты. 1 - Один домен, одна зона DNS

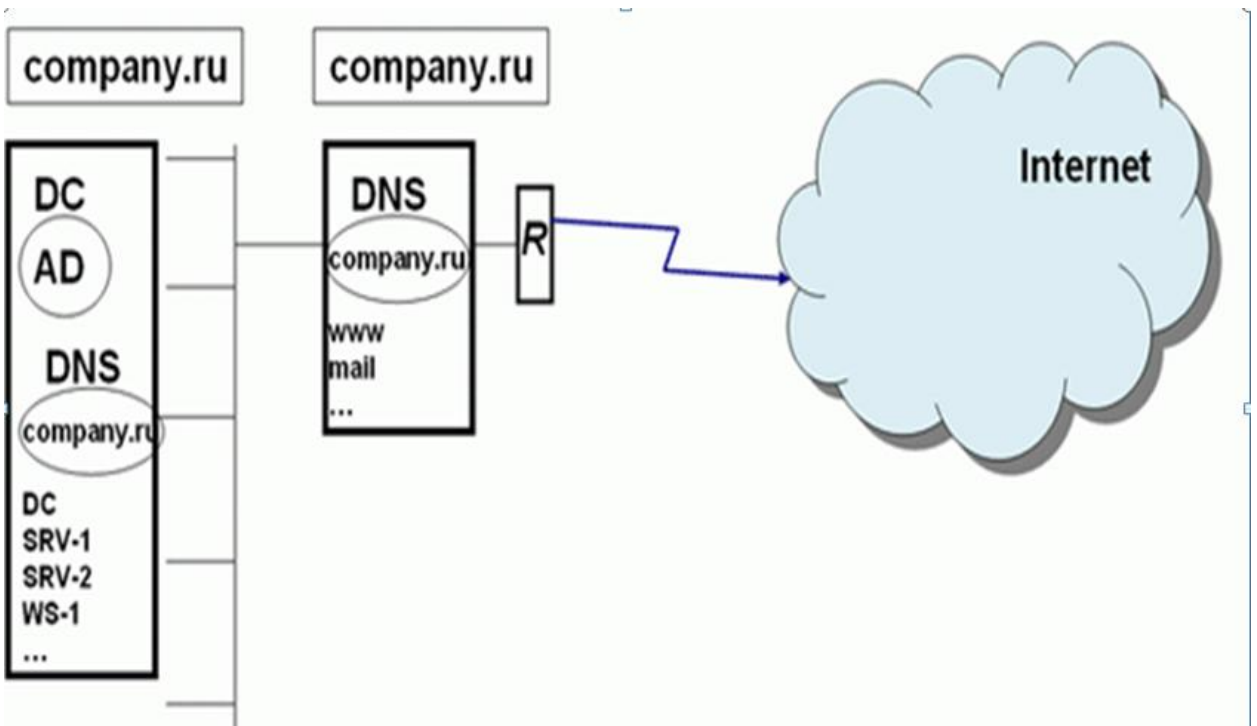


Рисунок 2 - Планирование пространства имен AD - основные варианты. 2 - "Расщепление" пространства имен DNS - одно имя домена, две различные зоны DNS

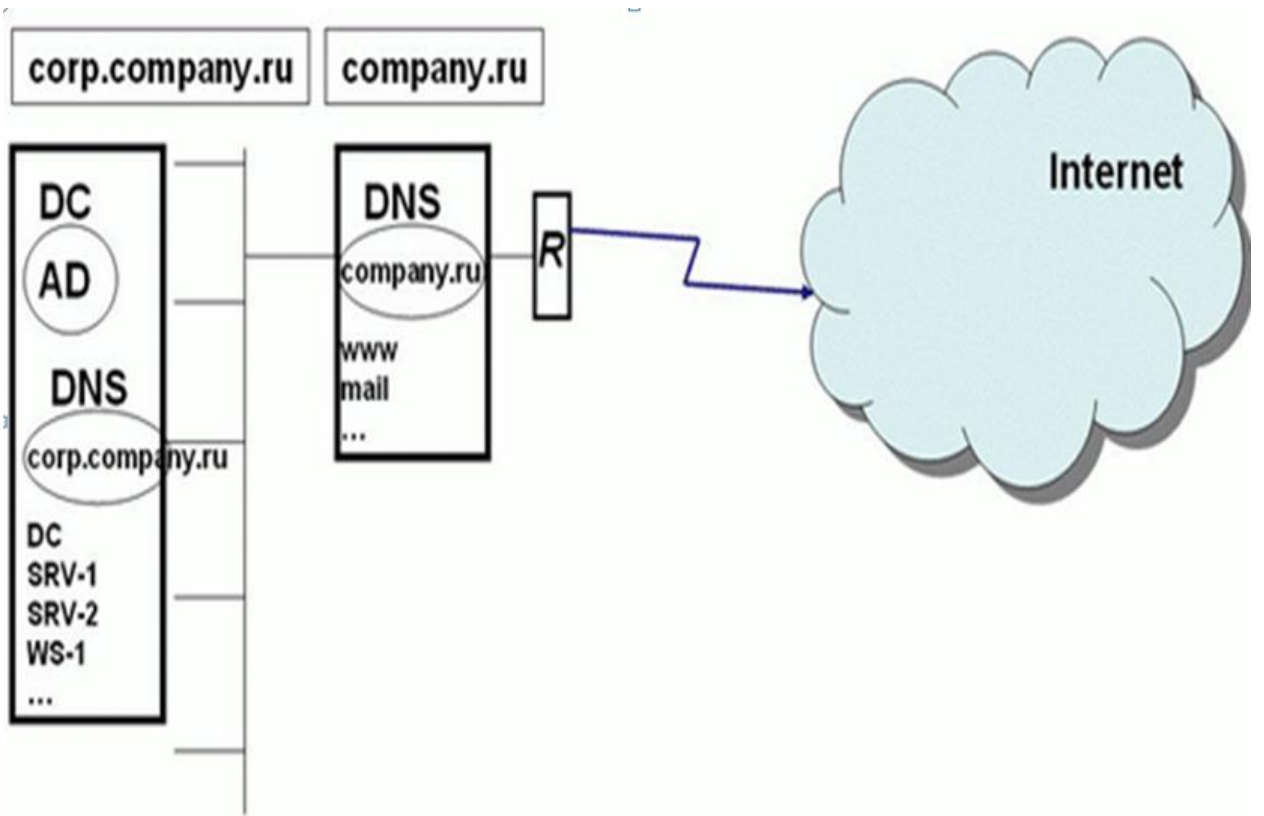


Рисунок 3 - Планирование пространства имен AD - основные варианты. 3 - Поддомен в пространстве имен DNS для поддержки Active Directory

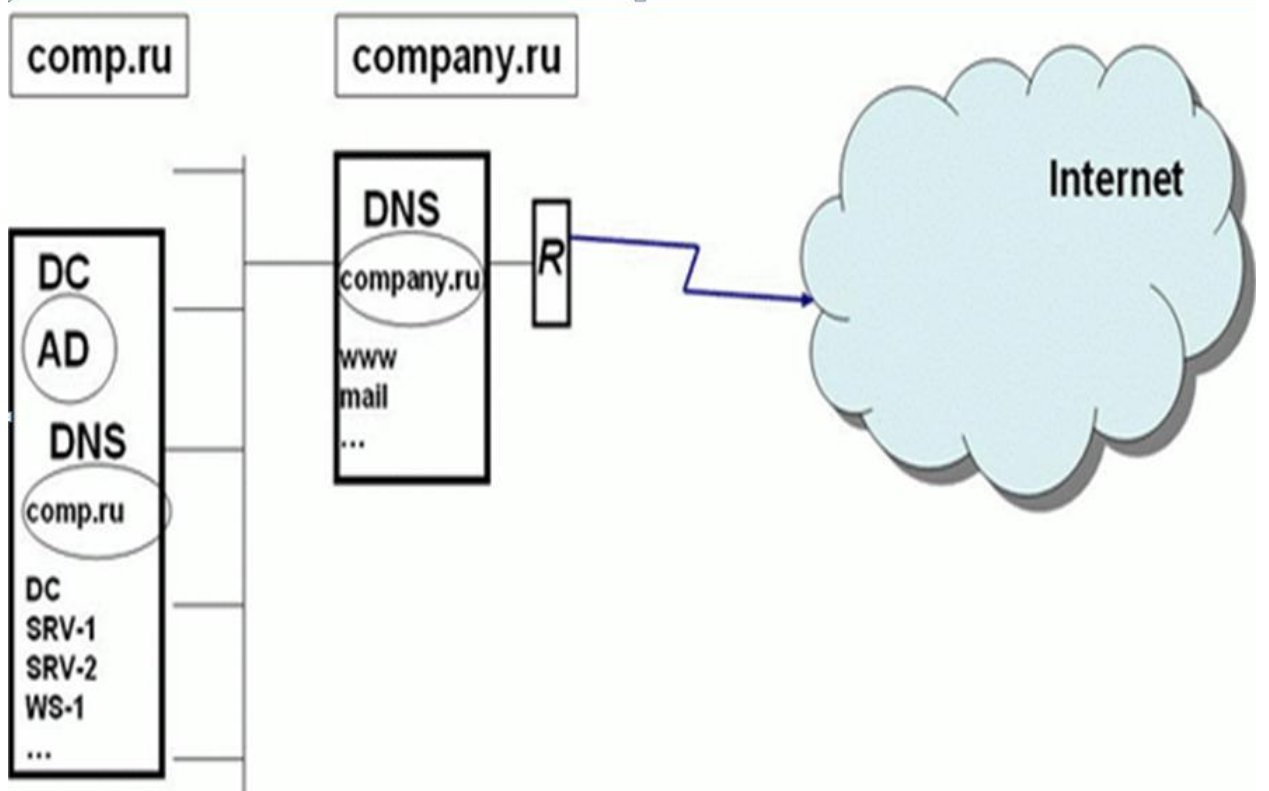


Рисунок 4 - Планирование пространства имен AD - основные варианты. 4 - Два различных домена DNS для внешних ресурсов и для Active Directory

5 - Домен с именем типа company.local. Во многих учебных пособиях и статьях используются примеры с доменными именами вида company.local. Такая схема вполне работоспособна и также часто применяется на практике. Однако в материалах разработчика системы Windows, корпорации Microsoft, нет прямых рекомендаций об использовании данного варианта.

Проектирование структуры OU.

После определения структуры домена организации и планирования доменного пространства имен необходимо разработать структуру организационных единиц (OU или подразделений - ОП). По информации, собранной о компании и ее персонале, необходимо определить, как лучше всего делегировать административные полномочия в доменах. Можно создать иерархию ОП в домене: в отдельном домене разместить пользователей и ресурсы, повторив структуру компании в конкретном подразделении. Таким образом, можно создать логичную и осмысленную модель организации и делегировать административные полномочия на любой уровень иерархии.

В каждом домене разрешается внедрять собственную иерархию ОП. Если организация имеет несколько доменов, то можно создать структуры ОП внутри каждого домена независимо от структуры в других доменах.

Организационное подразделение позволяет:

- отразить структуру компании и организации внутри домена. Без ОП все пользователи поддерживаются и отображаются в одном списке независимо от подразделения, местоположения и роли пользователя;
- делегировать управление сетевыми ресурсами, но сохранить способность управлять ими, то есть присваивать административные полномочия пользователям или группам на уровне ОП;
- изменять организационную структуру компании;
- группировать объекты так, чтобы администраторы легко отыскивали сетевые ресурсы.

Этапы проектирования административной структуры защиты.

1) Планирование использования организационных единиц (OU) в каждом домене.

2) Выработка стратегии управления учетными записями пользователей, компьютеров и групп.

3) Эффективная реализация групповой политики.

OU служит контейнером, в который можно поместить ресурсы и учетные записи домена.

Затем можно назначить OU административные разрешения и позволить содержащимся в нем объектам наследовать эти разрешения.

OU могут содержать любые объекты следующих типов:

- пользователи;
- компьютеры;
- группы;
- принтеры;
- приложения;
- политики безопасности;
- общие папки;
- другие OU.

При планировании иерархии ОП важно соблюсти следующие правила:

1) Хотя глубина иерархии ОП не ограничена, производительность мелкой иерархии выше, чем глубокой.

2) ОП должны отражать неизменные структурные единицы организации.

Модели классификации ОП в иерархии ОП.

1) Модель деления на ОП согласно выполняемым задачам.

2) Географическая модель деления на ОП. Иногда при создании ОП учитывается местоположение филиалов компании.

3) Модель деления на ОП согласно выполняемым задачам и географическому местоположению.

Стандартные модели структуры OU:

- 1) Модель структуры OU на основе местоположения.
- 2) Модель структуры OU на основе структуры организации.
- 3) Модель структуры OU на основе функций.
- 4) Смешанная модель структуры OU - сначала по местоположению, затем по структуре организации.
- 5) Смешанная модель структуры OU - сначала по структуре, затем по местоположению.

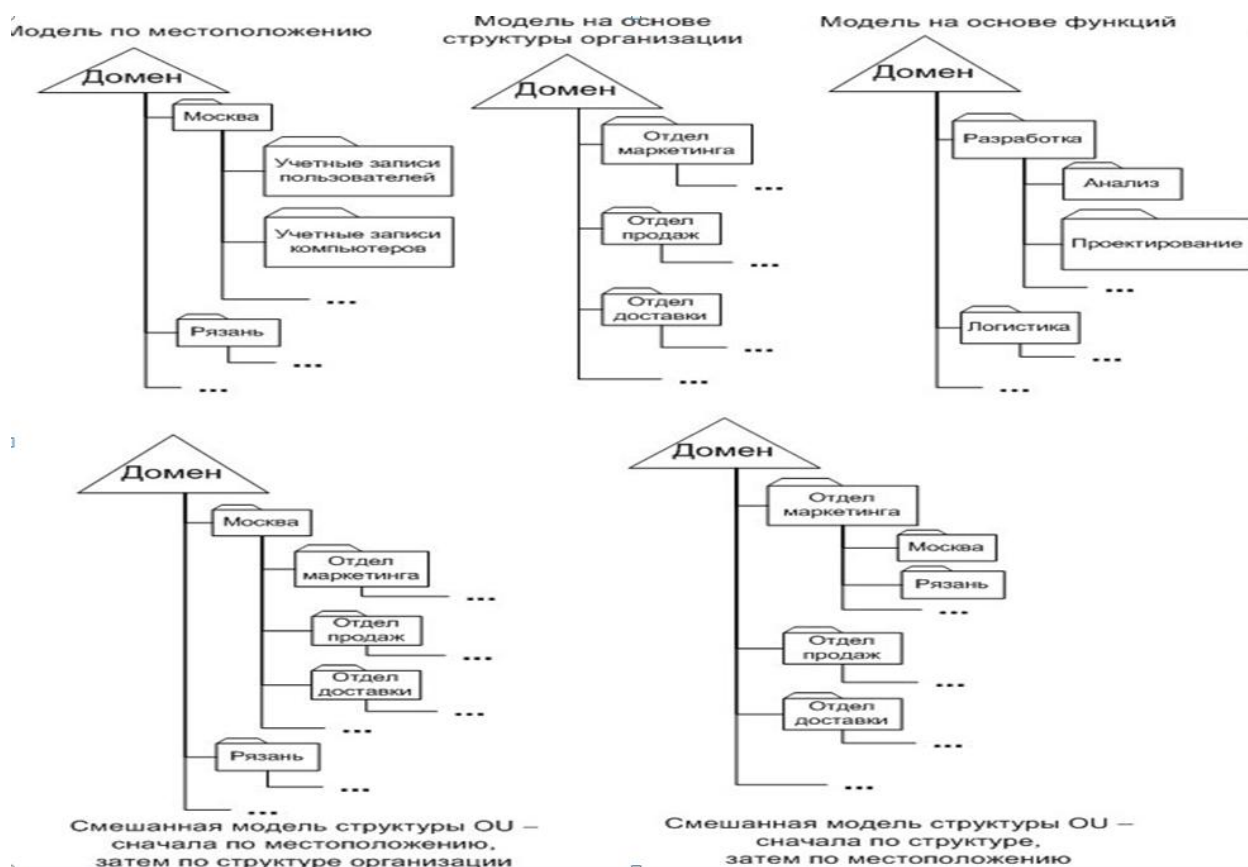


Рисунок 5 - Стандартные модели структуры OU

Управление объектами через консоль Active Directory

Windows Server поддерживает множество мощных средств командной строки, упрощающих управление Active Directory:

- DSADD — добавляет объекты в каталог;
- DSGET — отображает («получает») свойства объектов каталога;

- DSMOD — изменяет выбранные атрибуты существующего объекта каталога;
- DSMOVE — перемещает объект из текущего контейнера в новое местоположение;
- DSRM — удаляет объект или все дерево ниже объекта по иерархии, либо удаляет и объект, и дерево;
- DSQUERY — запрашивает в Active Directory объекты, отвечающие указанным условиям поиска.

Локальные профили пользователей. По умолчанию профили пользователей хранятся локально в папке %Systemdrive%\Documents and Settings\%Username%.

Когда пользователь входит в систему впервые, система создает для него профиль путем копирования профиля Пользователь по умолчанию (Default User). Имя для нового профиля формируется на основе имени для входа, указанного при первом входе в систему.

Все изменения рабочего стола пользователя и программной среды хранятся в локальном профиле пользователя. Для каждого пользователя существуют отдельные профили, поэтому все параметры индивидуальны.

Пользовательская среда расширена за счет профиля Все пользователи (All Users), который может включать ярлыки на рабочем столе или в меню Пуск (Start), адреса компьютеров в сети и даже данные приложений. Для создания среды пользователя элементы профиля Все пользователи (All Users) соединяются с профилем пользователя. По умолчанию только члены группы Администраторы (Administrators) могут модифицировать профиль Все пользователи (All Users).

Профиль является локальным в полном смысле. Если пользователь входит в другую систему, документы и параметры, являющиеся частью его профиля, не перемещаются. Вместо этого, когда пользователь впервые входит в систему, она генерирует для него новый локальный профиль.

Перемещаемые профили пользователей. Если пользователь работает на нескольких компьютерах, можно настроить перемещаемый профиль пользователя (roaming user profile, RUP), чтобы гарантировать сохранность и неизменность его документов и параметров вне зависимости от того, в какую систему он входит. RUP хранит профили на сервере, а значит их можно архивировать, проверять на наличие вирусов и централизованно управлять ими.

Чтобы настроить RUP, создайте общую папку на сервере. В идеальном случае это должен быть файловый сервер, на котором часто проводится архивирование.

На вкладке Профиль (Profile) диалогового окна Свойства (Properties) пользователя

введите Путь к профилю (Profile Path) в следующем формате:
\\<имя_сервера>\<имя_общего_ресурса >\% Username%.

Вместо переменной % Username% будет автоматически подставлено имя входа пользователя.

Когда пользователь выходит из системы, его профиль выгружается на сервер профилей. Теперь пользователь может входить в эту или в любую другую систему в домене, и его документы и настройки, являющиеся частью RUP, всегда будут под рукой.

Создание преднастроенного профиля пользователя

Для упорядочения и предварительной настройки рабочего стола и программной среды можно создавать настроенные профили пользователя, чтобы:

- создать простой доступ к необходимым сетевым ресурсам и приложениям;
- исключить доступ к ненужным ресурсам и приложениям;
- упростить работу службы поддержки по устранению неполадок.

Для создания преднастроенного профиля пользователя не требуются никакие специальные средства. Просто войдите в систему и измените

рабочий стол и настройки приложений по своему усмотрению. Лучше использовать для этого отдельную учетную запись, чтобы без необходимости не изменять собственный профиль.

Создав профиль, войдите в систему с административными реквизитами. Из Панели управления раскройте окно Система (System), перейдите на вкладку Дополнительно (Advanced) и в области Профили пользователей (User Profiles) щелкните Параметры (Settings). Выберите созданный профиль и щелкните Копировать (Copy To). Введите путь к профилю в стандартном формате записи пути (UNC): \\<имя_сервера>\<имя_общего_ресурса>\<имя_пользователя>. В области Разрешить использование (Permitted To Use) щелкните Изменить (Change), чтобы выбрать пользователя, для которого вы настроили этот профиль. Таблица управления доступом (ACL) для папки профиля будет настроена так, чтобы разрешить доступ этому пользователю.

Теперь раскройте свойства объекта пользователя и на вкладке Профиль (Profile) в поле Путь к профилю (Profile Path) введите тот же NСпуть. При следующем входе пользователя в домен этот профиль будет загружен и определит среду пользователя.

При помощи перемещаемых профилей можно создать стандартную среду рабочего стола для нескольких пользователей с одинаковыми должностными обязанностями. Этот процесс схож с процессом создания преднастроенного профиля для одного пользователя, но результирующий профиль будет доступен нескольким пользователям.

Настройка обязательного профиля.

Обязательный профиль не позволяет пользователям изменять среду профиля. Точнее, обязательный профиль не сохраняет изменения от сеанса к сеансу. Хотя пользователь и может внести изменения, при следующем входе в систему его рабочий стол будет выглядеть так же, как раньше.

Чтобы сделать профиль обязательным, просто переименуйте файл в корневой папке профиля. Интересно, что обязательные профили не

настраиваются путем назначения разрешений. Файл, который вам требуется переименовать, — Ntuser.dat. Это скрытый файл.

Найдите файл Ntuser.dat в профиле, который собираетесь сделать обязательным.

Переименуйте его в Ntuser.man. Профиль (перемещаемый или локальный) теперь является обязательным.

Задание к лабораторной работе:

В ходе выполнения лабораторной работы необходимо ознакомиться со следующими вопросами и представить проект планирования AD для своего предприятия:

- стратегия именования (механизм отличительных имен (Distinguished Name, DN), относительное отличительное имя (Relative Distinguished Name, RDN), основное имя объекта);

- выбор схемы планирования пространства имен AD.

- проектирование структуры OU;

- выбор модели классификации ОП в иерархии ОП;

- правила разбиения на сайты;

- принципы размещения сетевых сервисов, глобальных каталогов (ГК)

и других служб Active Directory;

- политика модификации схемы.

- Создание и управление объектами пользователей в консоли Active Directory;

- Создание и использование шаблонов объектов пользователей

- Импорт объектов пользователей при помощи CSVDE

- Использование DSADD, DSGET, DSMOD, DSMOVE, DSRM, DSQUERY для управления объектами пользователей.

- Управление профилями пользователей (локальные, преднастроенные, обязательные профили). Какими профилями можно в рамках лабораторной работы управлять?

- Работа с группами (группа безопасности - 1) локальная доменная, 2) глобальная, 3) универсальная). Какие группы целесообразно назначать и когда?
- Изучить, какие специальные группы есть и их представление – обосновать, какие группы для чего используются.
- Управление учетными записями групп из консоли Active Directory
- Использование Ldifde.exe.
- Использование DSADD, DSGET, DSMOD, DSMOVE, DSRM, DSQUERY для управления объектами групп.
- Создание учетных записей компьютеров из консоли Active Directory
- Использование DSADD, DSGET, DSMOD, DSMOVE, DSRM, DSQUERY для управления объектами компьютеров.
- Присоединение компьютера к домену

Контрольные вопросы:

- 1) Какие утилиты командной строки Вы знаете для работы с объектами AD?
- 2) Что такое локальный профиль?
- 3) Что такое преднастроенный профиль? Приведите пример использования.
- 4) Какие стандартные модели структуры OU Вы знаете?
- 5) Какие схемы планирования пространства имен AD Вы знаете?

Литература:

- 1 Cisco Systems, Программа сетевой академии Cisco CCNA 1 и 2. Вспомогательное руководство, 3-издание, исправленное, «Вильямс», 2007 г., 1168 стр.
2. Шиндер Д.Л., Основы компьютерных сетей, Вильямс, 2002 г., 615 стр.
3. Олифер В.Г., Н.А.Олифер «Компьютерные сети», учебник, СПб-Петербург, 2001 г.
4. Страницы руководства ОС Linux. (man)
5. Windows Server 2003 Наиболее полное руководство, БХВ, 2003 г.
6. Тодд Мазерс, Администрирование Windows Server 2003 на терминальном сервере, Москва-Санкт-Петербург-Киев, 2007 г., 1058 стр.
7. Ф. Зубанов, Active Directory. Подход профессионала., Русская редакция, Москва, 2003 г., 544 стр.
8. В. Столлингс, Компьютерные сети, протоколы и технологии интернета, БХВ-Петербург, 2005, 832 стр.
9. В.Г. Олифер , Н.А. Олифер «Компьютерные сети, принципы, технологии, протоколы» - ИД «Питер» 1999 г.
10. М.Гук “Аппаратные средства локальных сетей - энциклопедия”, ИД “Питер”, 2001 г.
11. Microsoft Corporation. Официальное учебное пособие Microsoft для самостоятельной подготовки, Компьютерные сети.Сертификация Network : учебный курс, М. : Изд.-торг.дом "Русская Редакция", 2002., 659 стр.
12. Дэн Холме, Орин Томас, Управление и поддержка Windows Server 2003, Москва, Русская редакция, 2004 г., 443 стр.
13. Рэнд Маримото и др., Windows Server 2003. Решения экспертов, Кулиц-образ, Москва, 2005 г., 784 стр.
14. Уильям Станек, Windows Server 2003, Справочник администратора, Москва, Русская редакция, 2004 г., 640 стр.

Содержание:

| | |
|------------------------------|--|
| Лабораторная работа №1 | |
| Лабораторная работа №2 | |
| Лабораторная работа №3 | |
| Лабораторная работа №4 | |
| Лабораторная работа №5 | |
| Лабораторная работа №6 | |
| Лабораторная работа №7 | |
| Лабораторная работа №8 | |
| Литература | |

МЕТОДИЧЕСКИЕ УКАЗАНИЯ И ЗАДАНИЯ
к лабораторным работам по дисциплине
«Организация компьютерных сетей»

(для студентов направления подготовки 09.03.04 “Программная инженерия ”)

Составители:

Алла Викторовна Чернышова