

**Қазақстан Республикасы білім және ғылым министрлігі
Қостанай облысы әкімдігінің білім басқармасы
«Қостанай жоғары политехникалық колледжі» КМҚК**

**Министерство образования и науки Республики Казахстан
КГКП «Костанайский политехнический высший колледж»
Управления образования акимата Костанайской области**

**УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ПО МОДУЛЮ
(Сборник лабораторных работ)**

КМ 08 «Ақпараттық қауіпсіздік бойынша шараларды қамтамасыз ету,
тораптық есептеу желісі мен Internet-ті пайдалану және икемдеу»

ПМ 08 «Обеспечение мер по информационной безопасности,
использование и настройка локальных вычислительных сетей и Interneta»

модуль атауы/ наименование модуля

Мамандық/Специальность:

1304000 «Есептеу техникасы және бағдарламалық қамтамасыз ету
(түрлері бойынша)»

1304000 «Вычислительная техника и программное обеспечение (по
видам)»

Біліктілік/Квалификация:

130404 3 техник-бағдармалашы

130404 3 техник-программист

Лабораторная работа №1.

Программа для изучения компьютерных сетей Netemul.

Цель: ознакомиться с возможностями моделирования компьютерных сетей в программе Netemul.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Интерфейс программы

Для начала скачаем (<http://netemul.sourceforge.net/rudownload.html>), установим программу, запустим ее и русифицируем командой **Сервис-Настройки** (рис. 1).

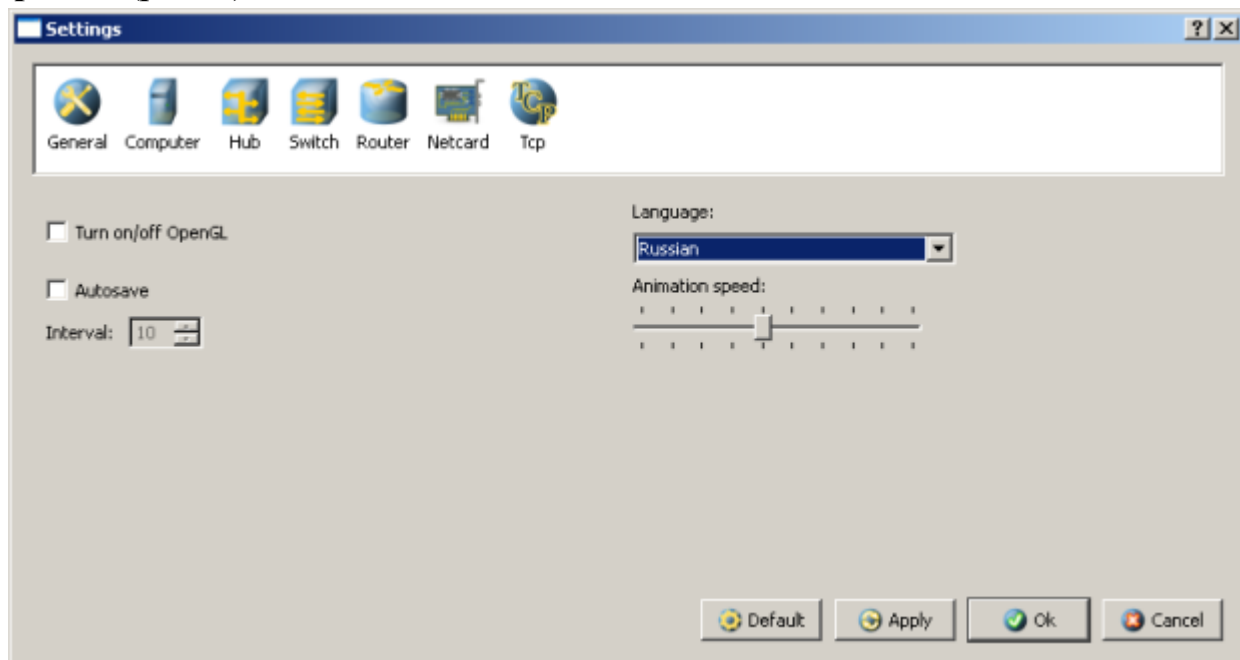


Рис. 1. Русифицируем интерфейс программы

В главном окне программы все элементы размещаются на рабочей области (на **Сцене**). На всей свободной области сцены, размеченной сеткой можно ставить устройства, при этом они не должны пересекаться. На **Панели устройств** размещены все необходимые для построения сети инструменты, а также кнопка отправки сообщений и **Запустить/Остановить**. На **Панели параметров** расположены свойства объектов. Для выделенного объекта появляются только те свойства, которые характерны для него (рис. 2).

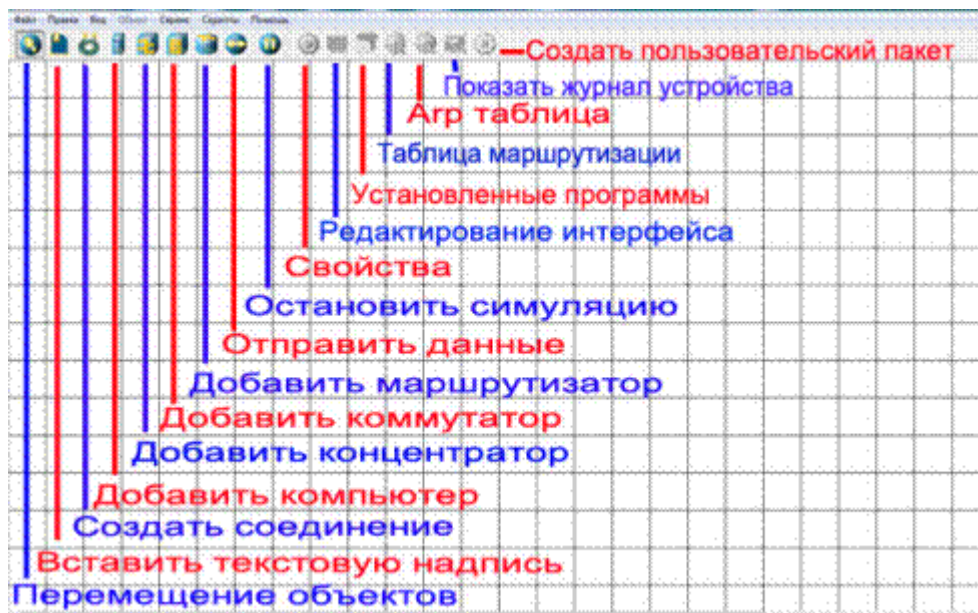
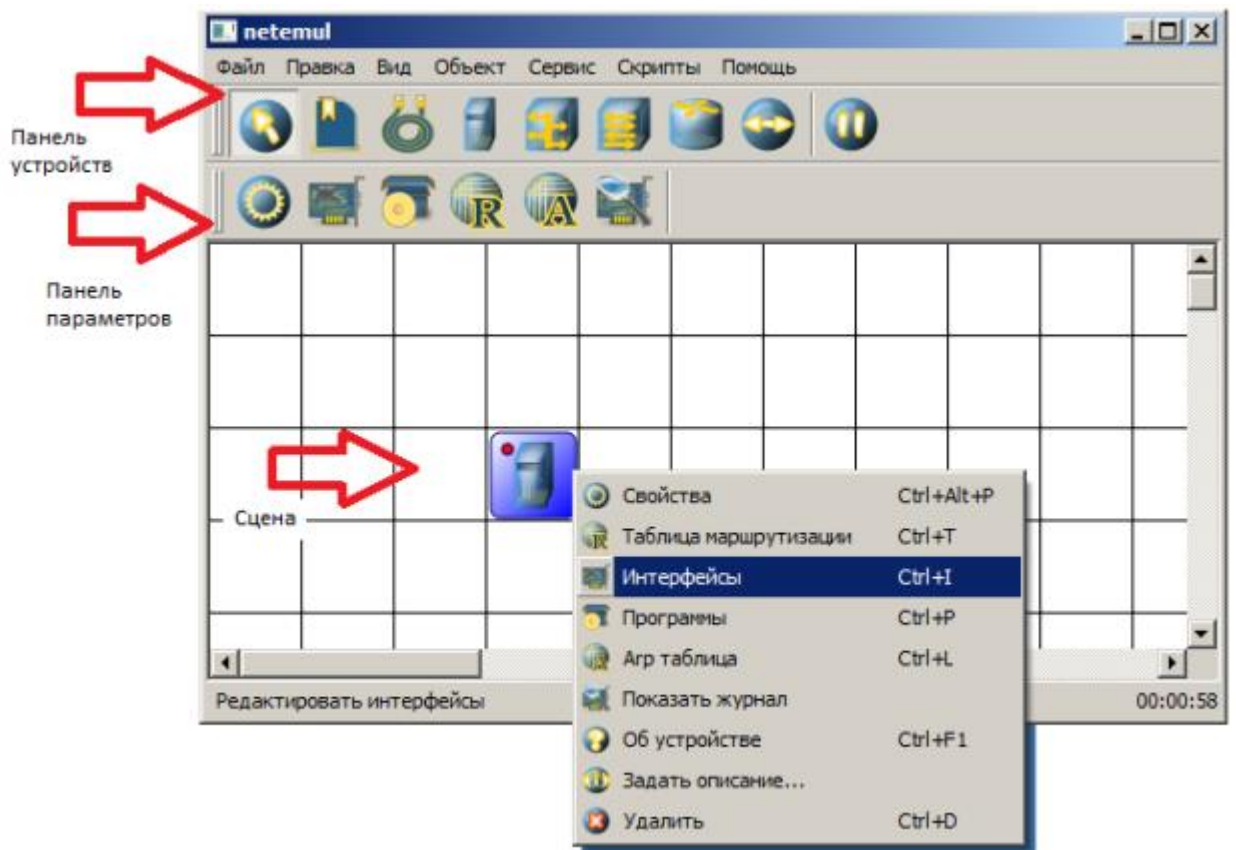


Рис. 2. Интерфейс программы Netemul

Пример 1. Строим сеть из двух ПК и коммутатора

Для начального знакомства с программой давайте построим простейшую локальную сеть и посмотрим, как она работает. Для этого выполните команду **Файл-Новый** и нарисуйте схему сети как на рис. 3.

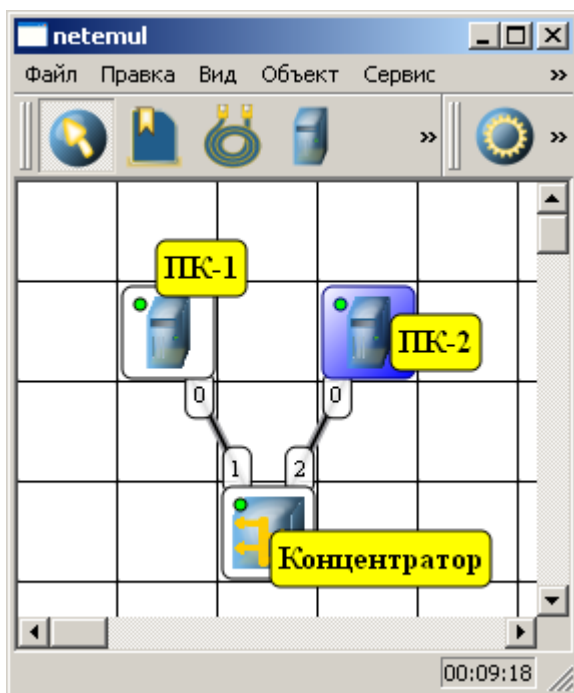


Рис. 3. Схема из двух ПК и концентратора

После рисования двух ПК и концентратора создадим их соединение (рис. 4).

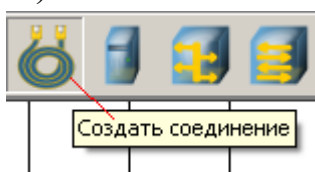


Рис. 4. Инструмент создания соединений сетевых устройств

В процессе рисования связей между устройствами вам потребуется выбрать соединяемые интерфейсы и нажать на кнопку **Соединить** (рис. 5 и 6).

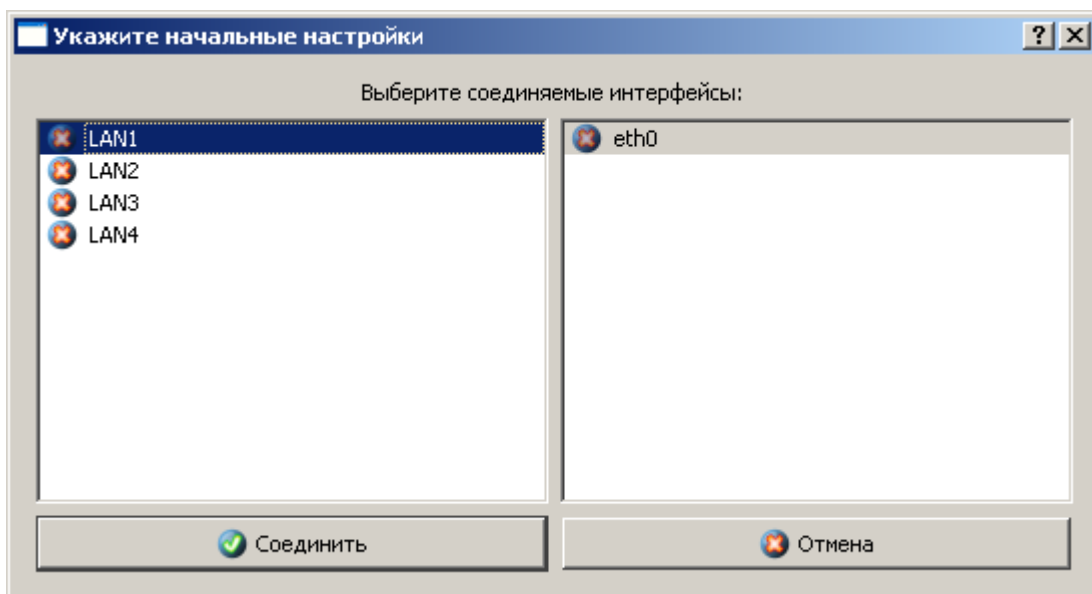


Рис. 5. Выбор начальных настроек соединения

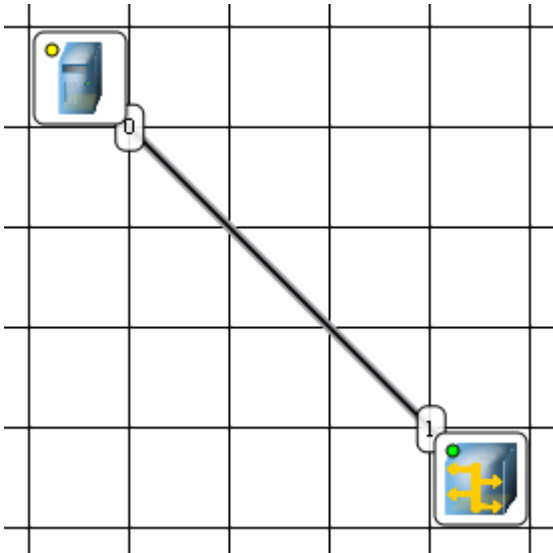


Рис. 6. Соединение устройств произведено

Теперь настроим *интерфейс* (сетевую карту) на наших ПК ее – рис. 6 и рис. 7.

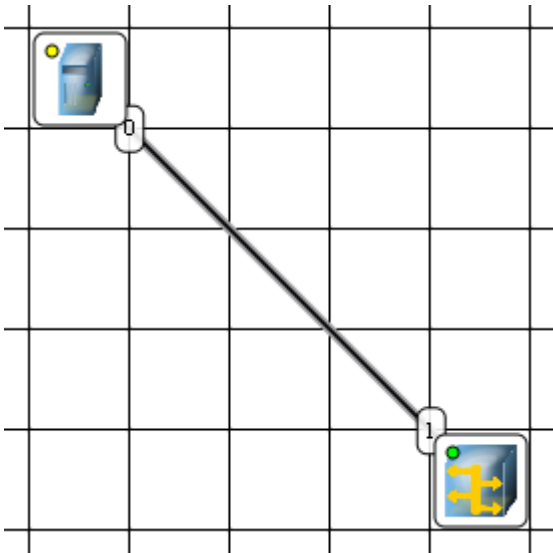


Рис. 6. Добавляем интерфейс

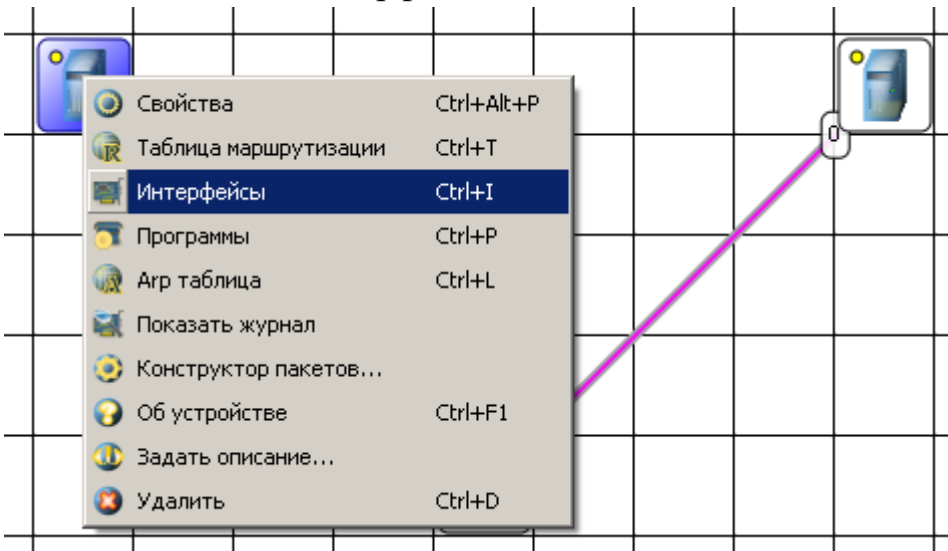


Рис. 7. Вводим IP адрес и маску сети

Примечание

Обратите внимание: после того, как вы напишете 192.168.0.1 маска появляется автоматически. После нажатия на кнопки **Применить** и **ОК** – появляется анимация движущихся по сети пакетов информации.

Все - сеть создана и настроена. Отправляем данные по протоколу TCP (рис. 8 и рис. 9).

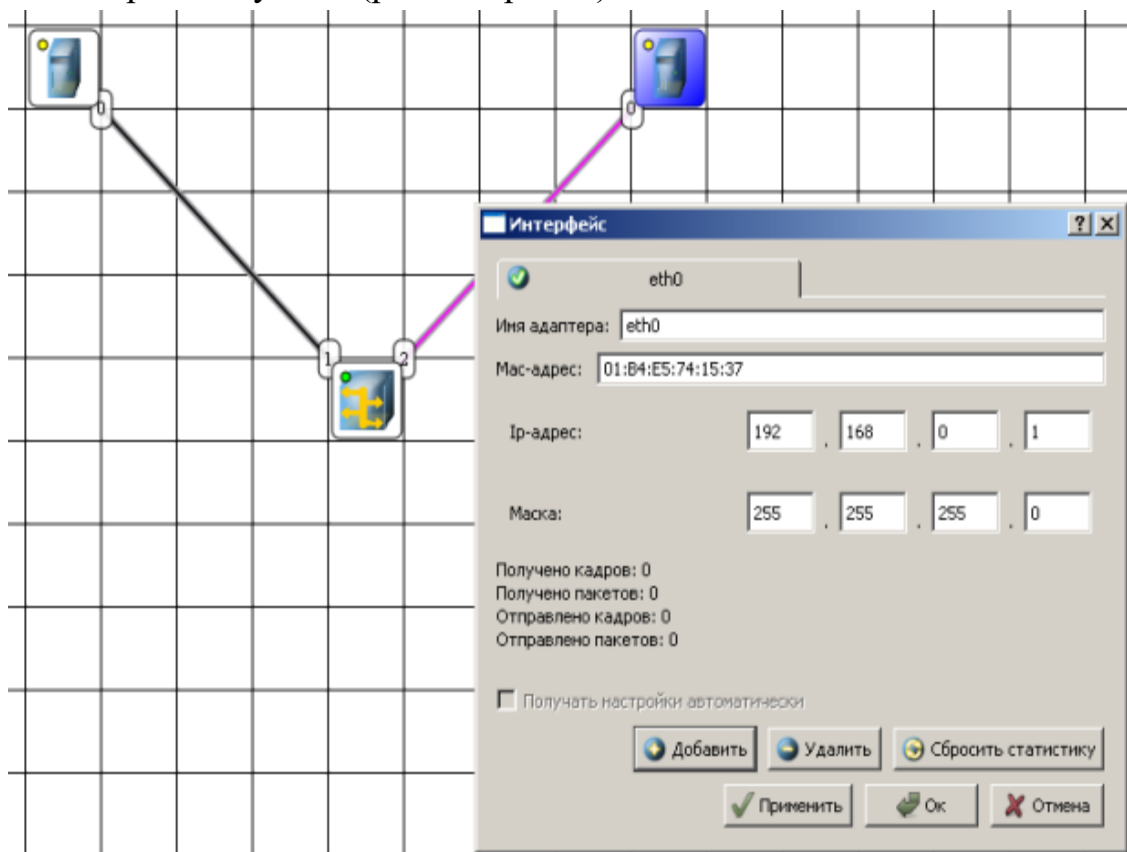


Рис. 8. Кнопка Отправить данные

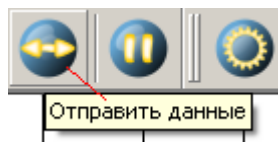


Рис. 9. Выбор протокола

Если вы где-то ошиблись, то появится соответствующее сообщение, а если все верно – то произойдет анимация движущихся по сети пакетов (рис. 10).

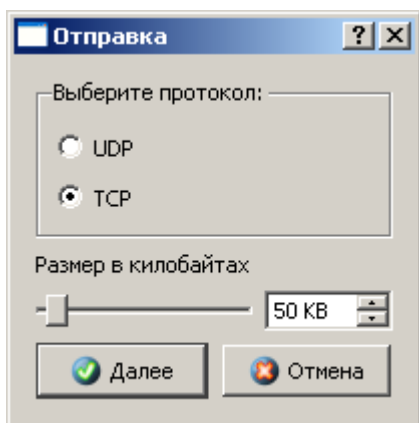


Рис. 10. Движение пакетов по сети

И еще один момент. По умолчанию каждый ПК имеет одну сетевую карту, но их может быть и несколько. Для того, чтобы добавить для ПК *адаптер* нужно щелкнуть на нем правой кнопкой мыши и выбрать пункт меню **Интерфейсы**. В результате откроется следующее *диалоговое окно* (рис. 11).

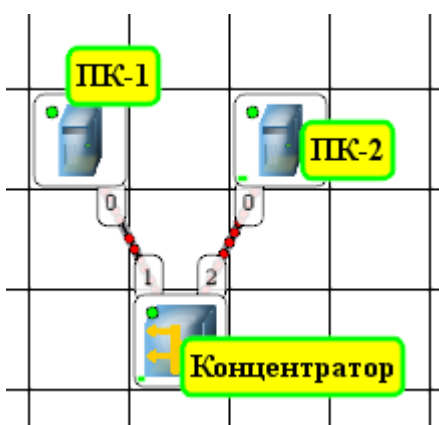


Рис. 11. Диалоговое окно работы с сетевым интерфейсом ПК

Нажимаем на кнопку **Добавить**, выбираем тип нового адаптера, нажимаем ОК, и у нас есть еще один *интерфейс*. В качестве примера на рис. 12 изображен ПК, имеющий три сетевых карты.

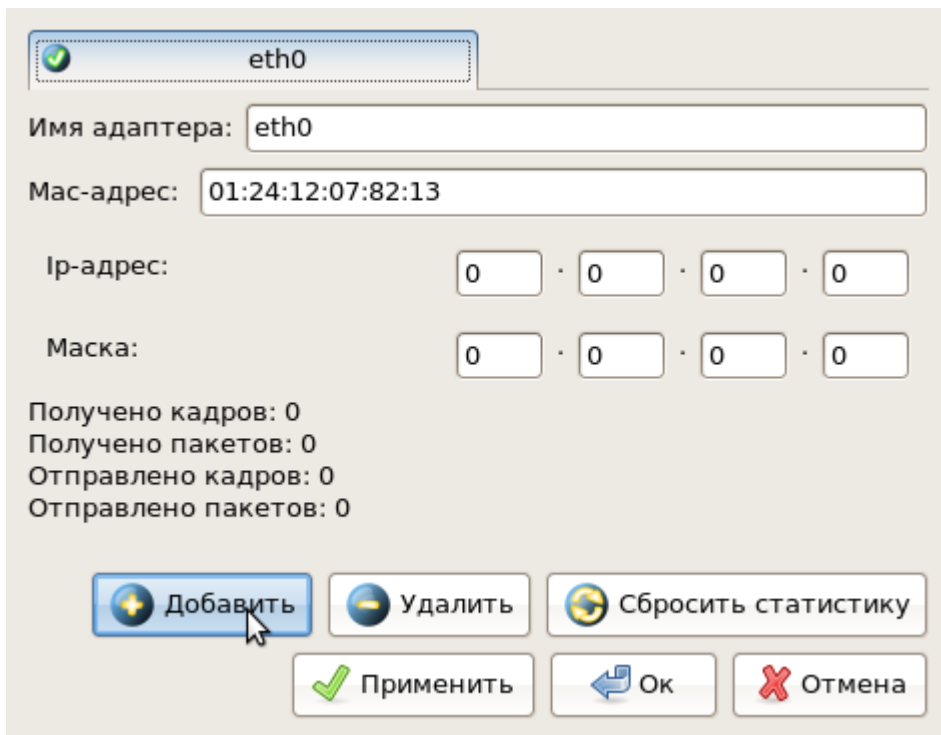


Рис. 12. В этом ПК установлены адаптеры eth0-eth3

Примечание

Каждый сетевой интерфейс (сетевой адаптер) имеет свой собственный mac-адрес. В программе Netemul в строке "Mac-адрес" можно задать новый адрес, но по умолчанию, при создании интерфейса, ему автоматически присваивается этот уникальный номер.

Задание 1. Построить сеть из двух ПК и свитча, изучить таблицу коммутации

В приведенной в этом примере схеме замените *хаб* на свитч и посмотрите у него таблицу коммутации (рис. 13).

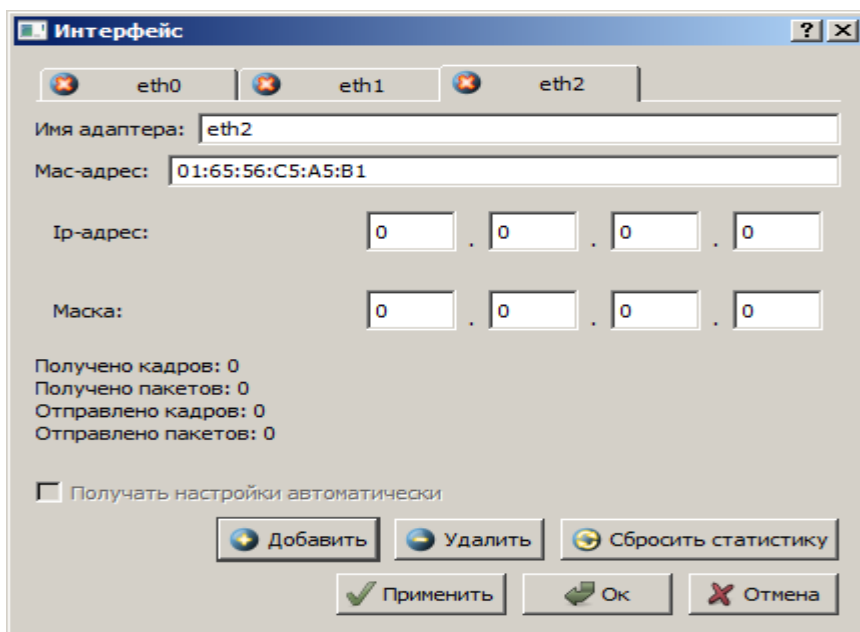


Рис. 13. Схема сети по топологии звезда построена

На рисунке:

- красный индикатор означает, что устройство не подключено;
- желтый - устройство подключено, но не настроено;
- зеленый - знак того, что устройство подключено, настроено и готово к работе.

Пример 2. Изучаем сеть из двух подсетей и маршрутизатора

Постройте новую *сеть* (рис. 14). Разобьем нашу *сеть* на 2 подсети. Допустим, у нас есть *пул* адресов сети класса С. Разобьем его на 2 части: 192.168.1.0-192.168.1.127 (слева) и 192.168.1.128-192.168.1.255 (справа) с маской 255.255.255.128.

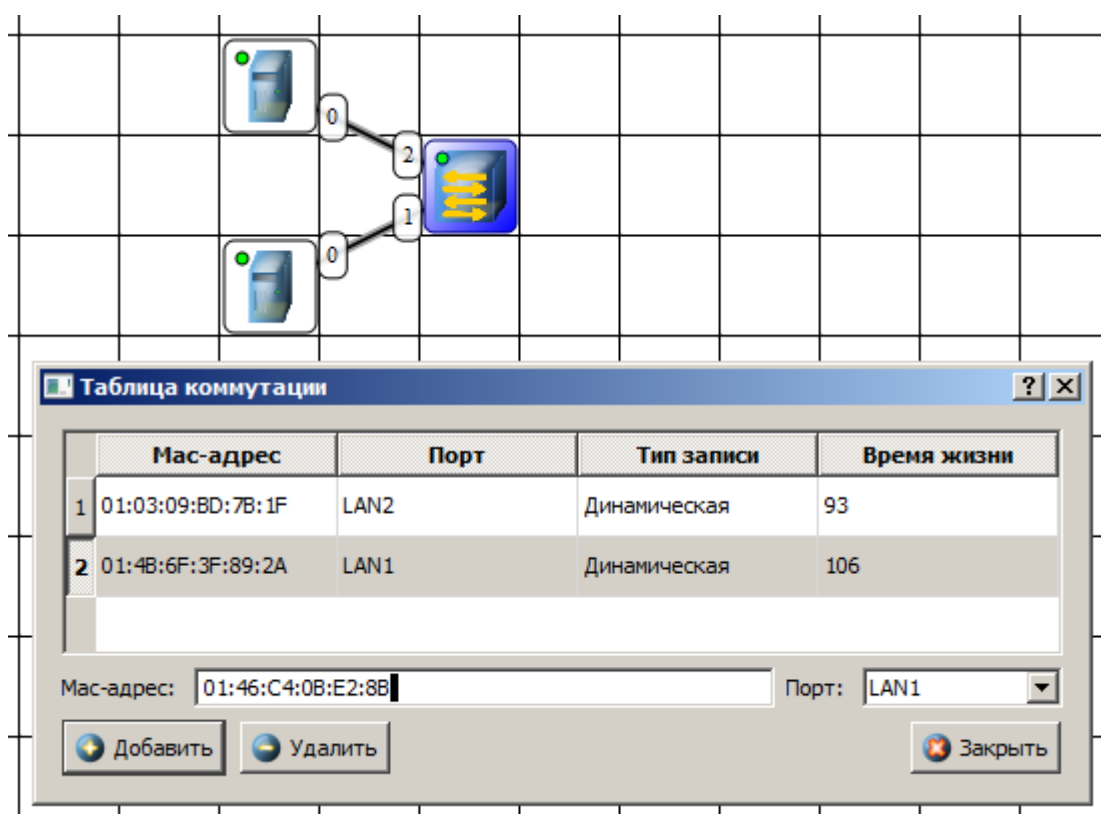


Рис. 14. Вариант сети из двух подсетей, соединенных маршрутизатором

Примечание

Обратите внимание на то, что число портов у коммутатора можно задавать. У нас на рисунке коммутатор восьмипортовый.

Настройка компьютеров

Для настройки *ip*-адреса интерфейса ПК из *меню* правой кнопки мыши открываем окно **Интерфейсы** и для левой (первой), подсети выставляем *ip*-адреса от 192.168.1.1 до 192.168.1.5 и маску подсети 255.255.255.128. Затем для правой (второй) подсети выставляем *ip*-адреса от 192.168.1.129 до 192.168.1.133 и маску подсети 255.255.255.128. После нажатия на кнопку

"ОК" или "Применить", мы можем наблюдать, как *индикатор* поменял цвет с желтого на зеленый и от нашего устройства, которому сейчас дали *адрес*, побегал *кадр* Агр-протокола. Это нужно для того, чтобы выявить, нет ли в нашей сети повторения адресов. В *поле* "Описание" необходимо имя каждому компьютеру. Оно в дальнейшем будет всплывать в подсказке при наведении мыши на устройство, а также при открытии журнала для устройства заголовков будет содержать именно это описание.

Настройка маршрутизатора

Пока послать сообщения из одной такой подсети в другую мы не можем. Необходимо дать *IP* адреса каждому интерфейсу маршрутизатора, а на конечных узлах установить шлюзы *по* умолчанию. В подсети левее маршрутизатора у всех узлов должен быть *шлюз* 192.168.1.126, правее - 192.168.1.254 (рис. 15 и рис. 16).

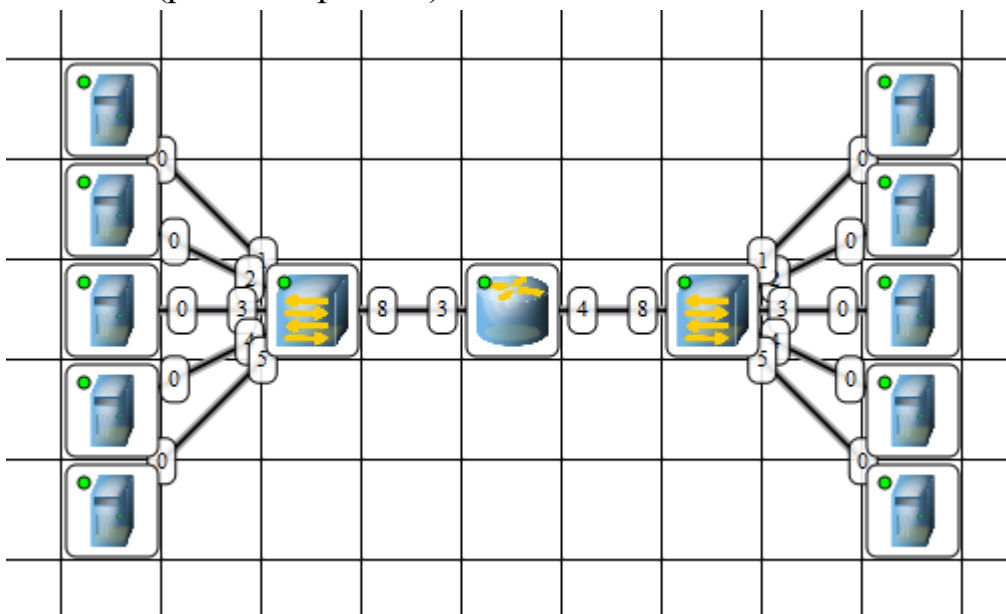
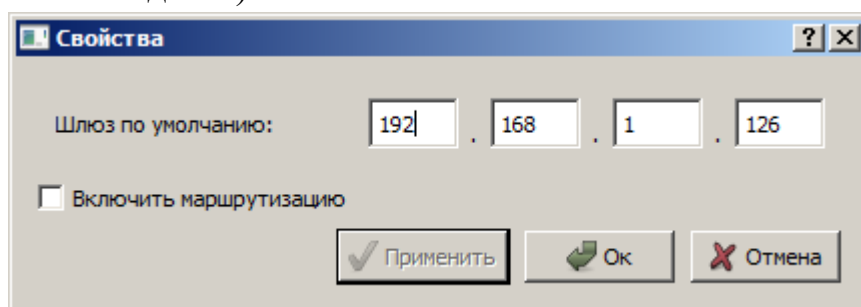


Рис. 15. Настройка шлюза по умолчанию, а также IP и маски для LAN3 (для левой подсети)



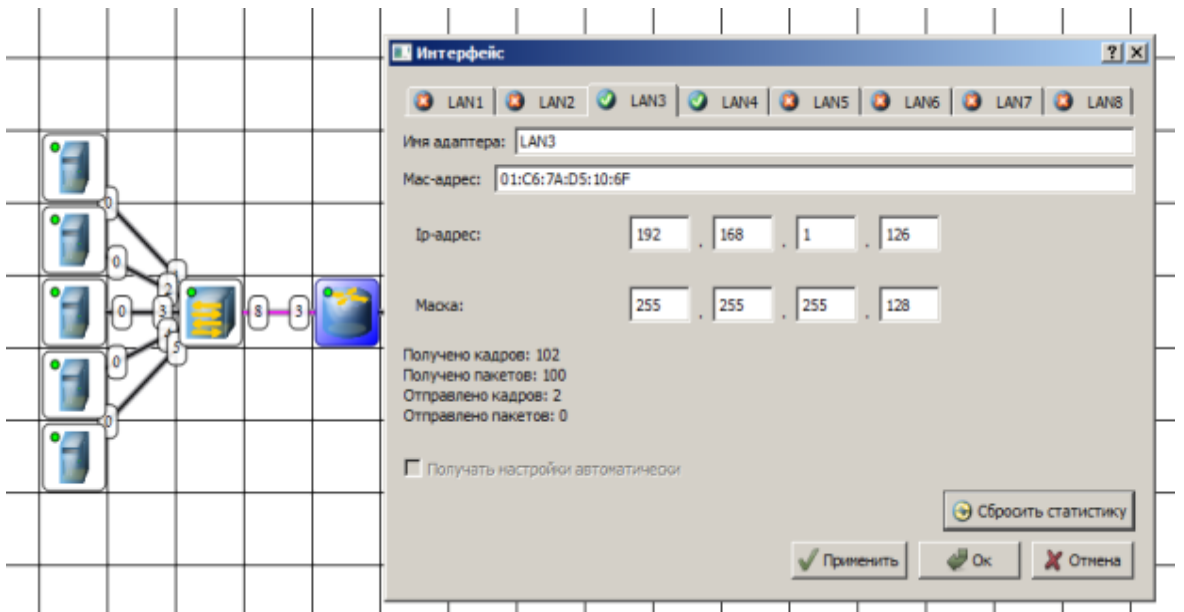


Рис. 16. Настройка шлюза по умолчанию, а также IP и маски для LAN4 (для правой подсети)

Шлюзы мы задали и теперь у нас полностью рабочая *сеть*. Давайте рассмотрим свойства ее объектов.

Свойства коммутатора. Откроем его таблицу коммутации (рис. 17). Сейчас она абсолютно пустая, т.к. не было ни одной передачи данных. Но при этом у нас есть возможность добавить статическую *запись*, для этого необходимо заполнить все поля соответствующими данными и нажать кнопку "Добавить".

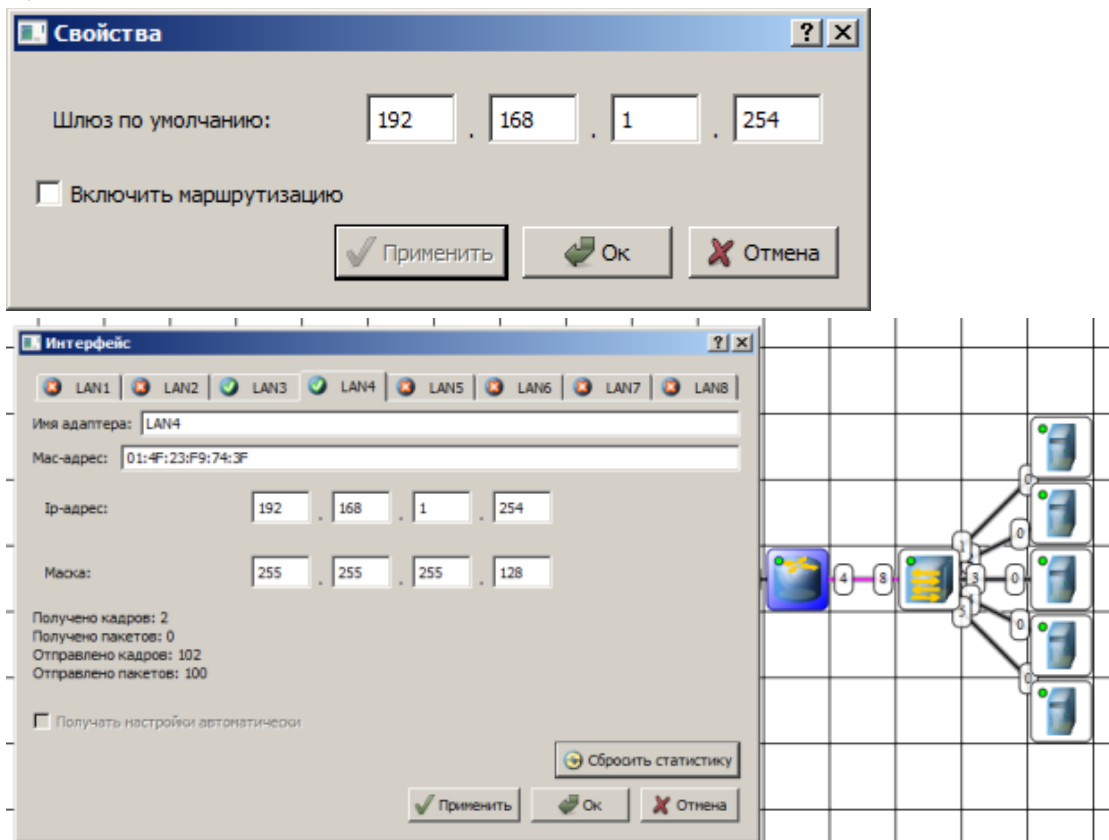


Рис. 17. Таблица коммутации коммутатора

Свойства маршрутизатора

В контекстном меню изучим пункты: *Таблица маршрутизации*, *Агр-таблица*, *Программы*. **Агр-таблица** пуста (по той же причине, что и *таблица* коммутации), но в нее также можно добавить статические записи. В **таблице маршрутизации** мы видим 2 записи (рис. 18). Эти записи соответствуют нашим подсетям, о чем говорят надписи в столбце **Источник**. В качестве источника может быть протокол *RIP*, установить который можно с помощью пункта **Программы**. В столбец **Шлюз** заносится *адрес* следующего маршрутизатора (или *адрес* шлюза, если другого маршрутизатора нет). В столбце **Интерфейс** *адрес* порта, с которого будем отправлять данные. В эту таблицу тоже можно занести статические записи, а в столбце **Источник** появится надпись **Статическая**.

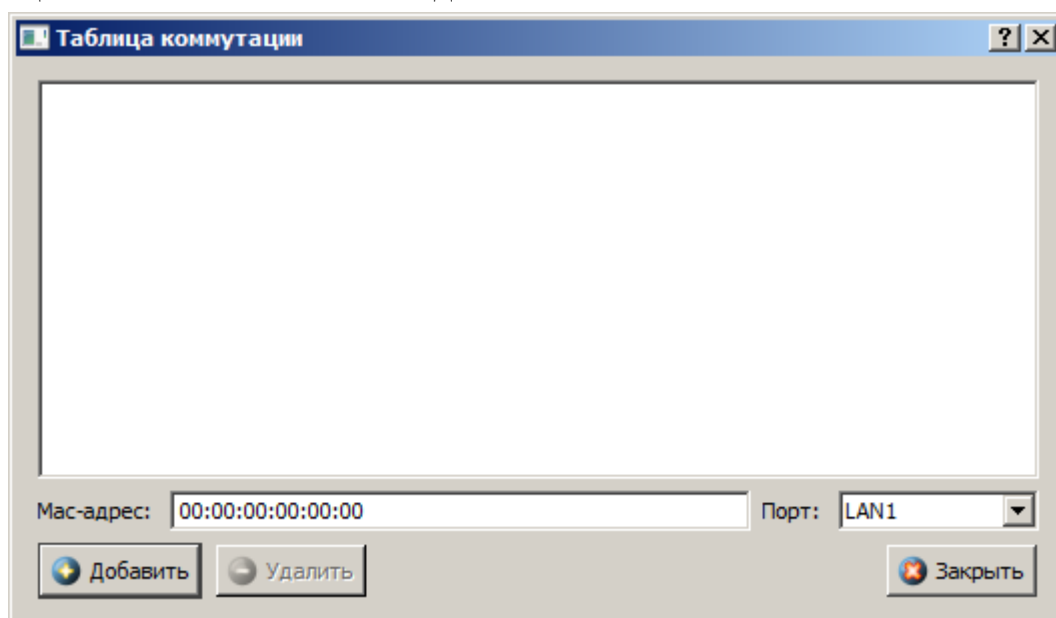



Рис. 18. Таблица маршрутизации маршрутизатора

Тестирование сети (Отправка пакетов)

Давайте проверим, насколько правильно функционирует *сеть*. Для того,

чтобы отправить пакеты, выберите на панели инструментов значок . При наведении мыши на рабочую область вы увидите оранжевый кружок, это значит, что надо указать от какого компьютера данные будут отправлены. Мы пошлем данные от компьютера, отмеченного на рисунке стрелкой (рис. 19).

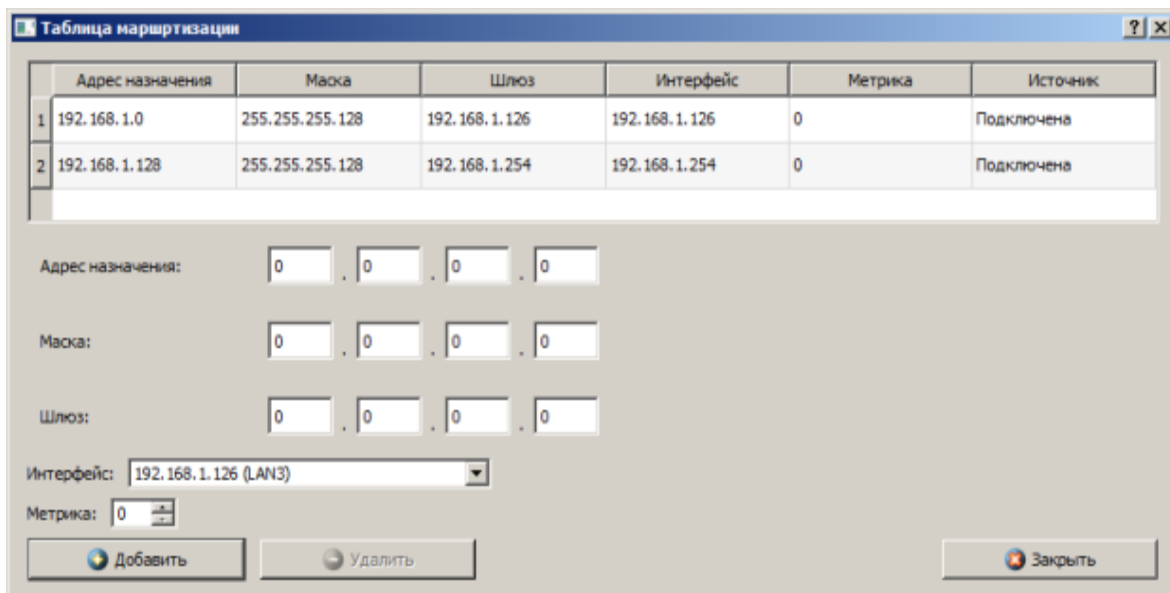


Рис. 19. Показан ПК, управляющий данными
Нажимаем на кнопку **Далее**. Теперь вам надо выбрать получателя (рис. 20).

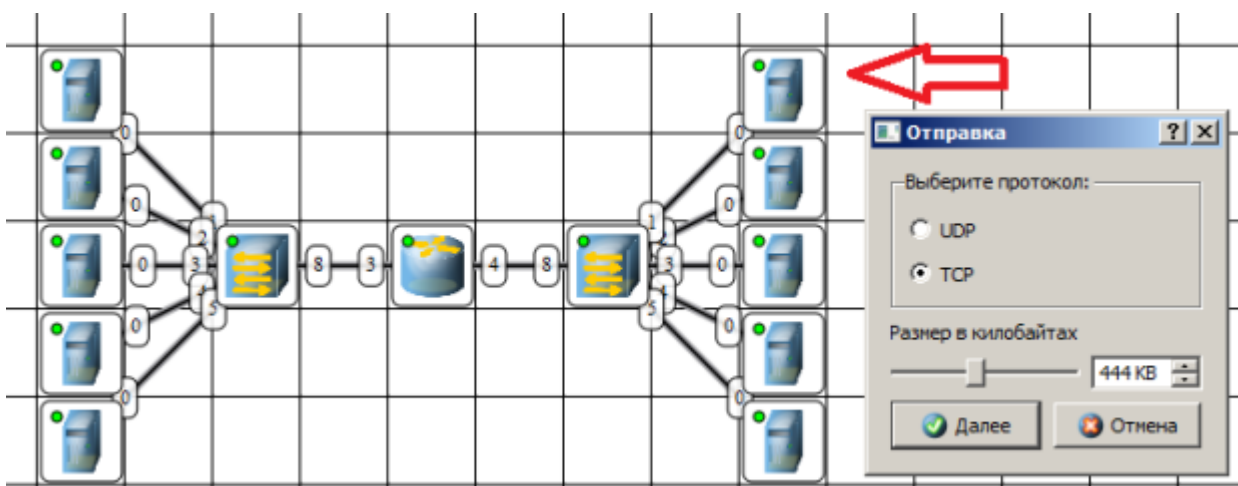


Рис. 20. Показан ПК, получающий данные
Далее нажимаем кнопку **Отправка** и наблюдаем бегущие по сети кадры (рис. 21).

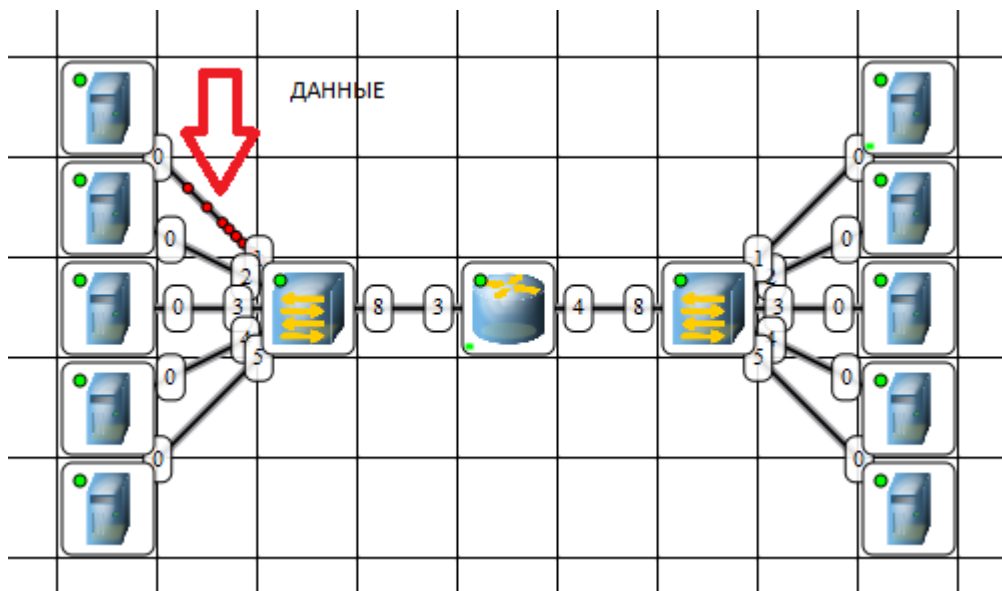


Рис. 21. По сети идут кадры данных

У каждого устройства в контекстном меню есть пункт "Показать журнал", можно открыть этот журнал и увидеть всю необходимую информацию о пакете, пришедшем (или отправленном), и его содержимое – рис. 22. На этом рисунке журнал открыт для ПК-получателя пакетов.

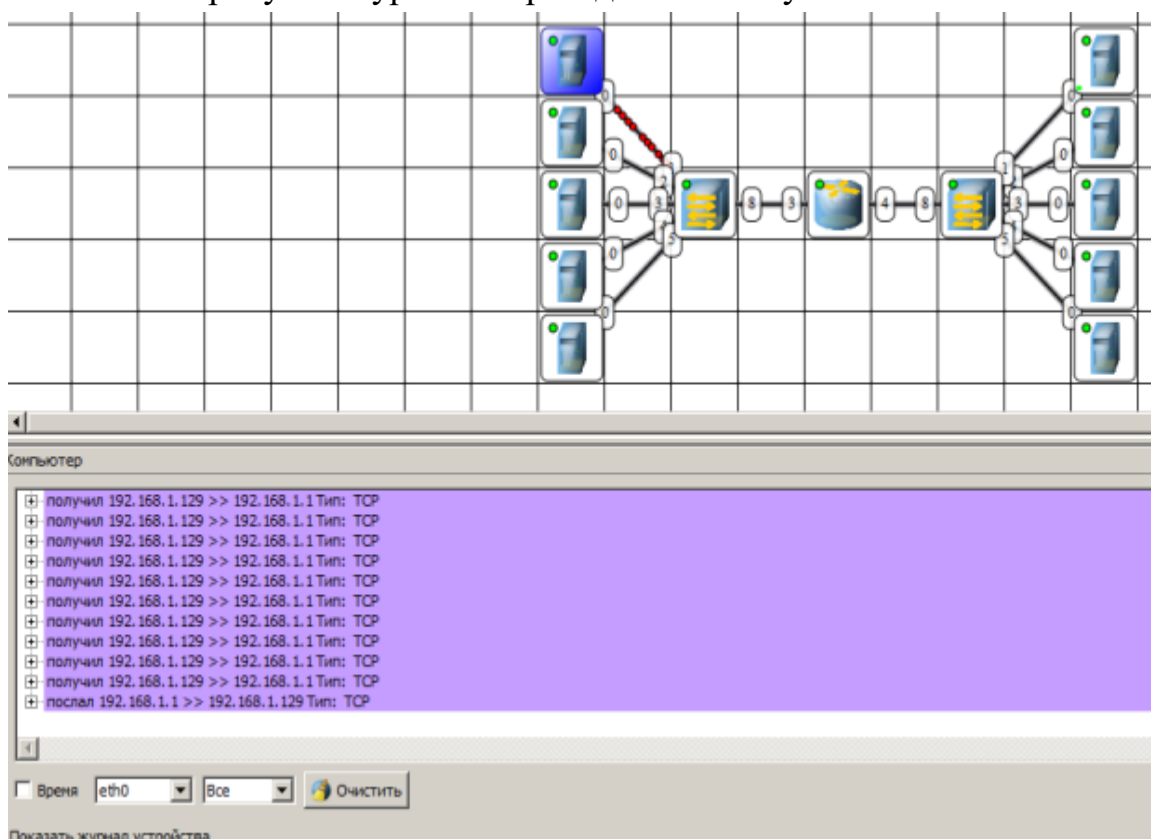


Рис. 22. Журнал устройства показывает, какую информацию содержали кадры данных

Задание 2. Построить сеть из восьми ПК, хаба, коммутатора и роутера. Настроить ее правильную работу

Построить сеть как на рис. 23 и настройте ее работу.

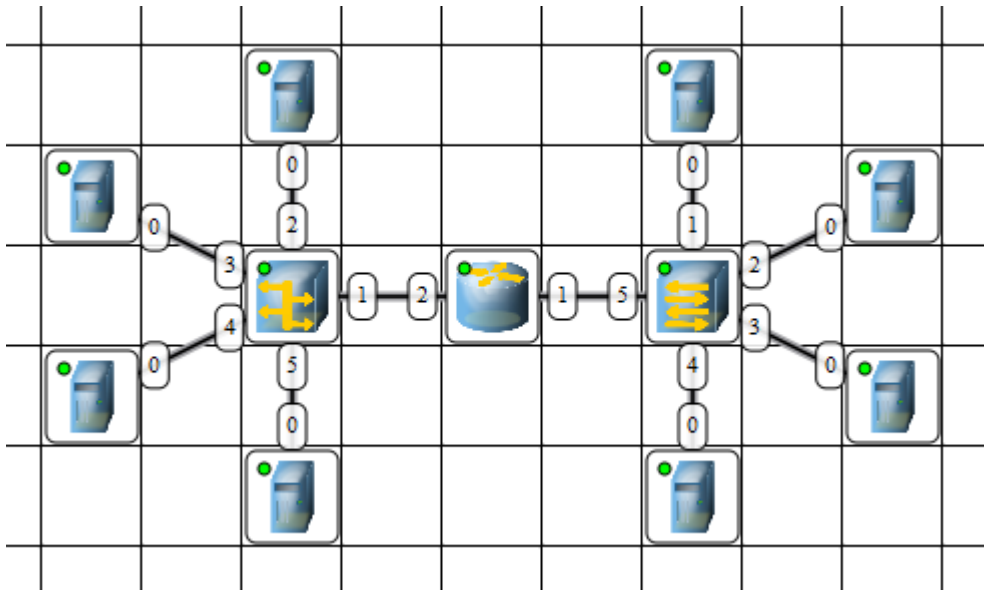


Рис. 23. Две подсети по топологии звезда

Краткие итоги

В лабораторной работе мы познакомились с интерфейсом эмулятора сети Netemul и выполнили серию практических задач, промоделировав и настроив работу серии локальных сетей (*сеть из двух ПК и коммутатора, сеть из двух ПК и свитча, сеть из двух подсетей и маршрутизатора, сеть из восьми ПК, хаба, коммутатора и роутера*). Весь практикум *по* решению этих задач отображен в скринкасте, прилагаемом к данной работе.

Ответить на контрольные вопросы

Контрольные вопросы

1. Что такое IP-адрес?
2. Что такое маска подсети?
3. Как работает концентратор?
4. Как работает коммутатор?

Лабораторная работа №2

Создание виртуальной машины и установка на ней операционной системы Windows 7

1. Копирование файлов виртуальной машины VMware Workstation 9 на физический ПК

Запустим *Setup* (рис. 7.1).

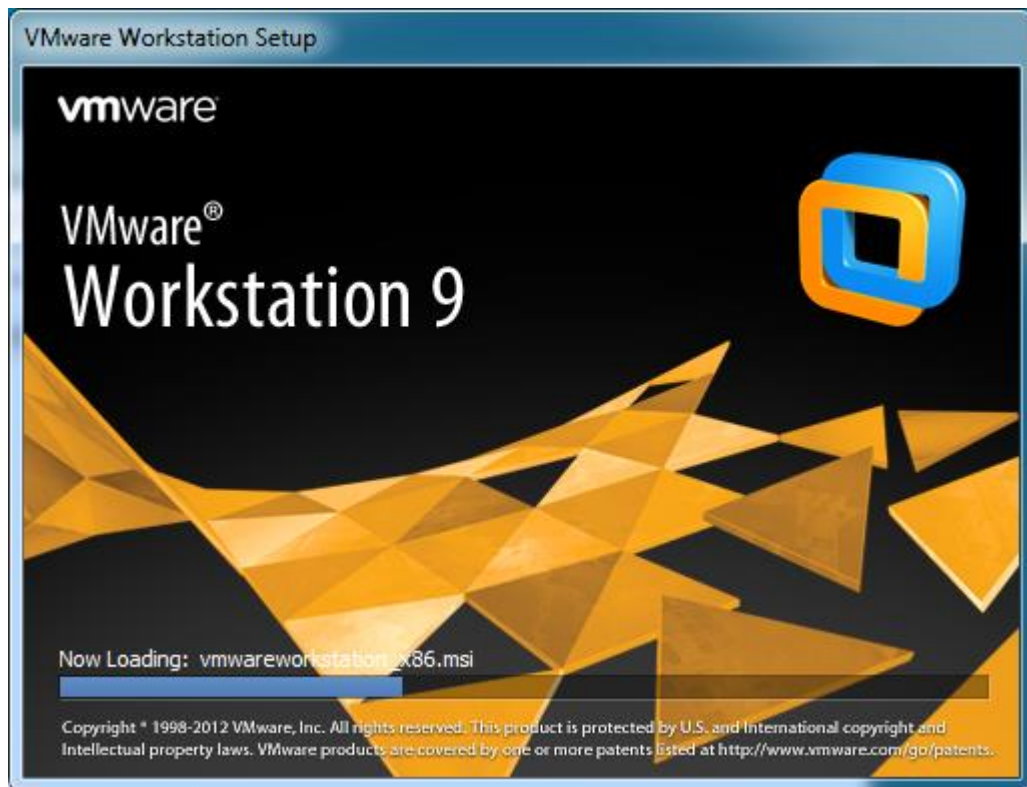


Рис. 7.1. Логотип программы VMware Workstation 9

Далее изменим *место* размещения VM на ПК и для этого будем создавать виртуальную машину не по шаблону (*переключатель Typical - Обычная*), а с нашими настройками (*переключатель Custom - Специальная*) – рис. 7.2.

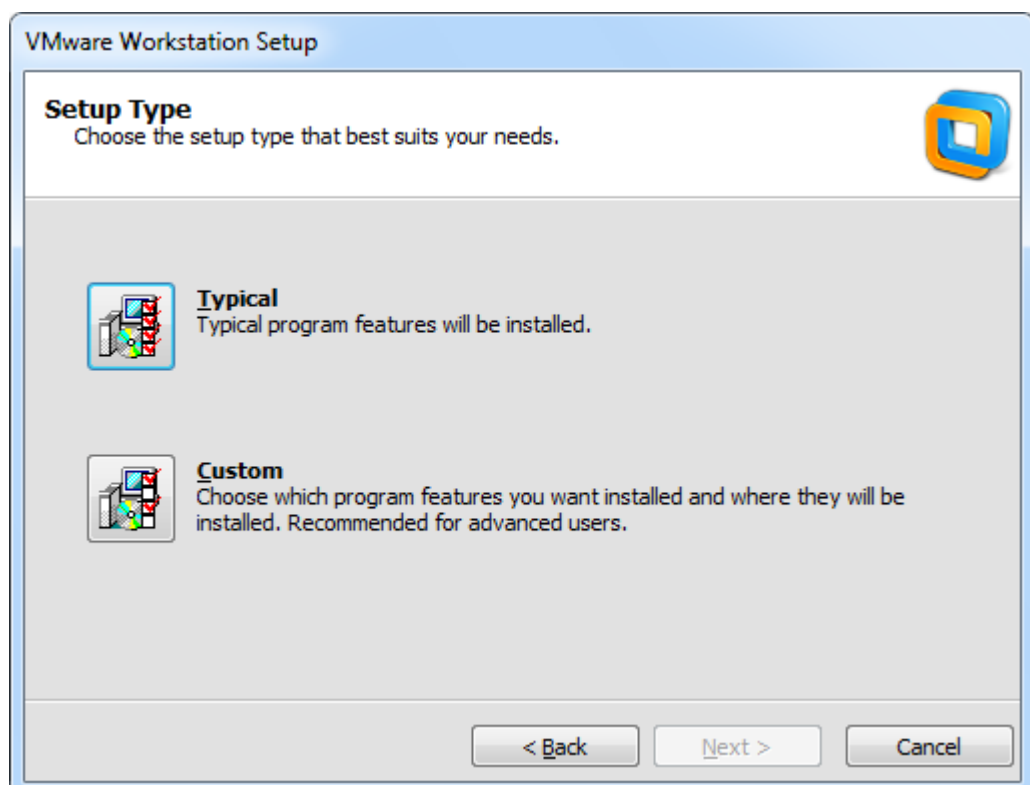


Рис. 7.2. Устанавливаем переключатель Custom (Специальная установка)

Примечание

Это не обязательно. Вы можете установить VM с настройками по умолчанию.

Стандартный путь для нахождения файлов виртуальной машины мы изменим (рис. 7.3 и рис. 7.4).

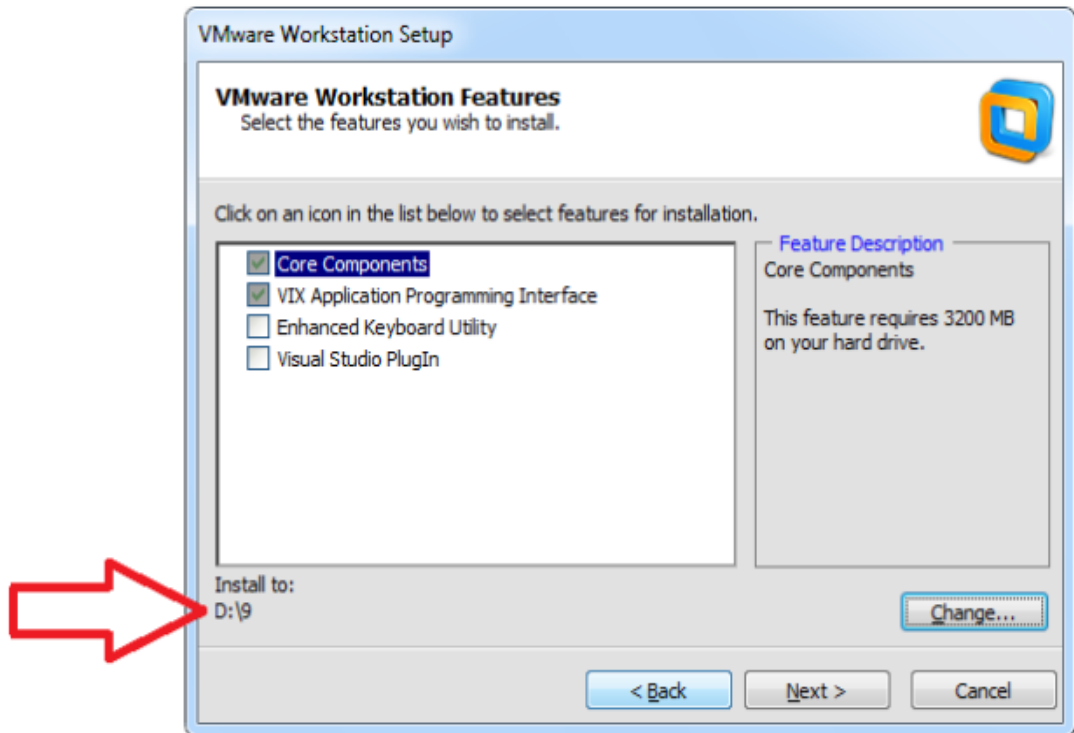


Рис. 7.3. Выбираем компоненты программы и путь их размещения

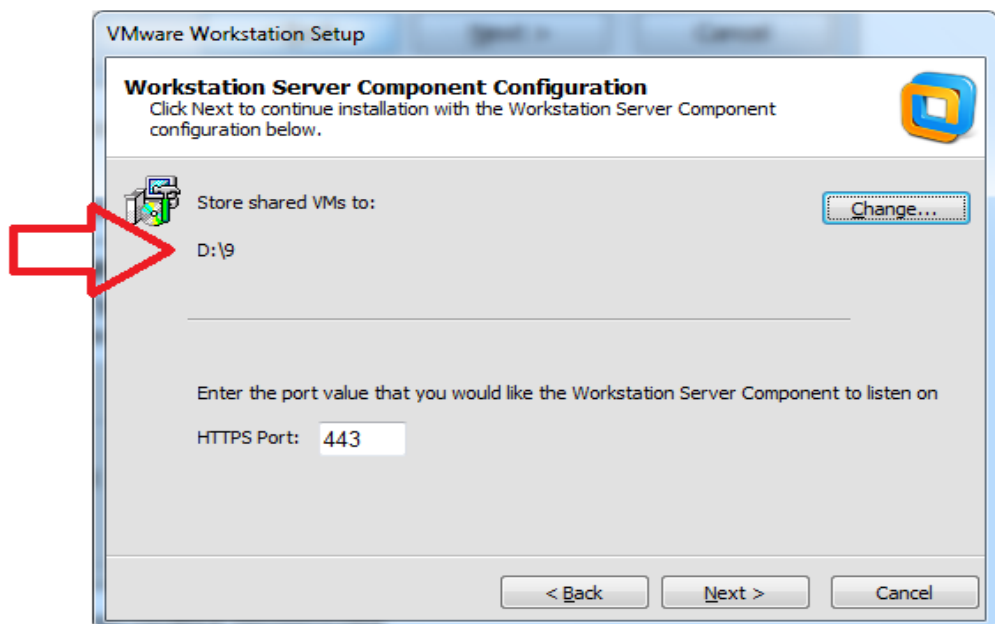


Рис. 7.4. Указываем путь для нахождения файлов виртуальной машины

Следующие окна оставляем с настройками по умолчанию и нажимаем на кнопку **Next (Следующий)**. В финале следующее сообщение (рис. 7.5)

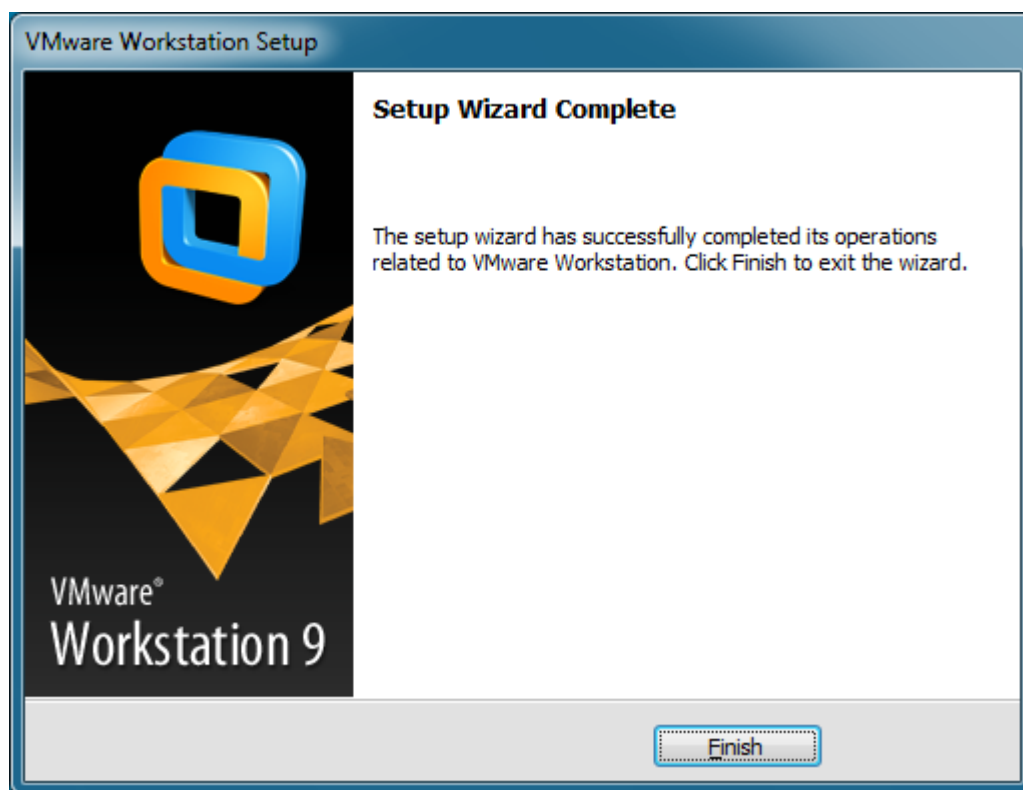


Рис. 7.5. Машина создана

С помощью русификатора английский *интерфейс* (рис. 7.6) меняем на русский (рис. 7.7).

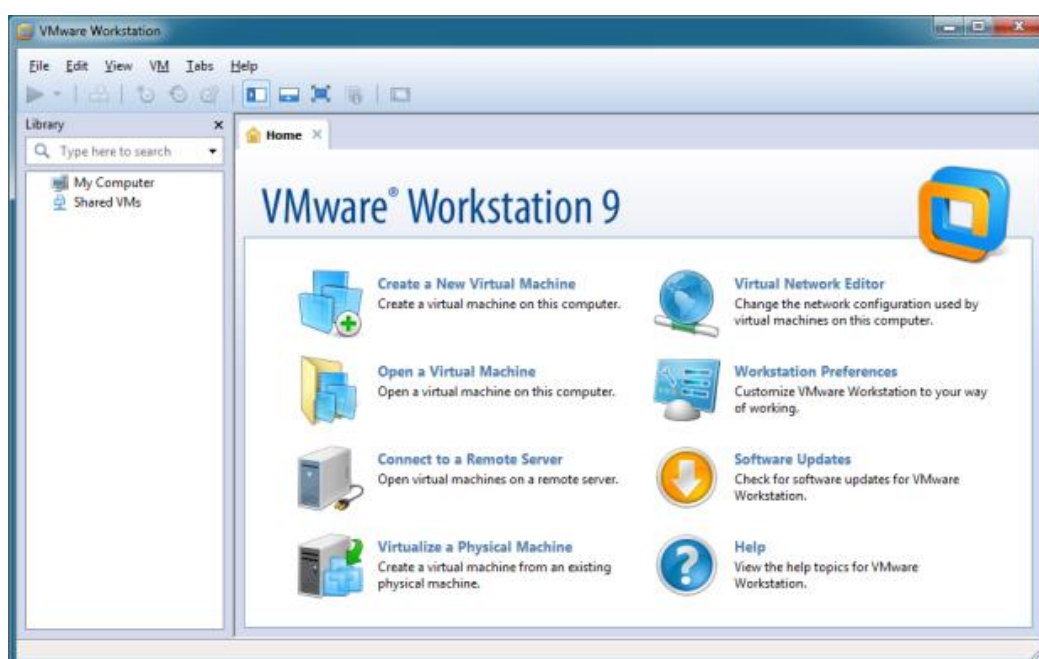


Рис. 7.6. Стартовое окно запуска VM с английским интерфейсом

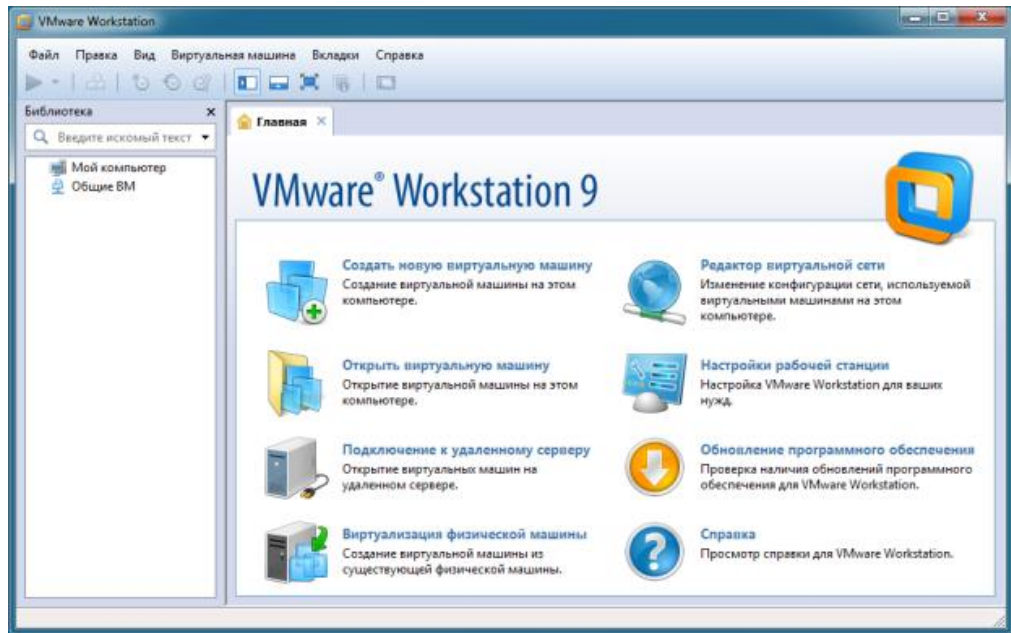


Рис. 7.7. Программа успешно русифицирована

2. Создание новой виртуальной машины и установка на нее ОС Windows 7

Щелкаем на значок машины на рабочем столе (рис. 7.8).

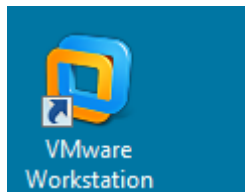


Рис. 7.8. Ярлык для VM

Выполняем команду **Файл-Новая виртуальная машина** или щелкаем мышкой на значке и устанавливаем *переключатель* **Выборочный** (рис. 7.9).

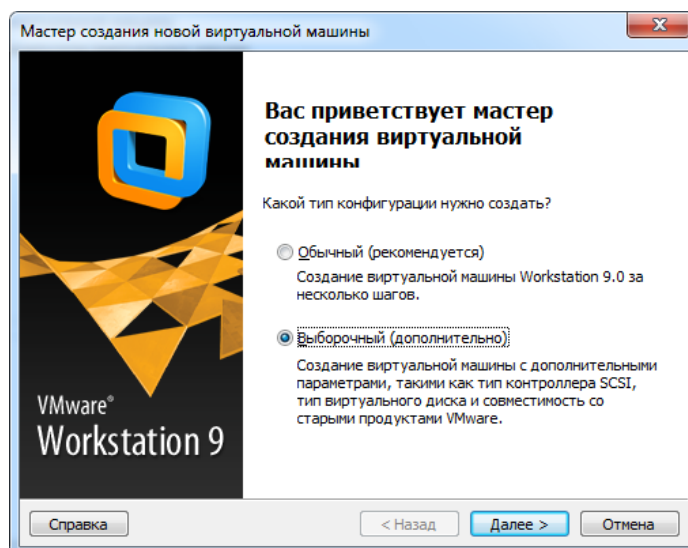


Рис. 7.9. Первое окно Мастера создания виртуальной машины

Примечание

Этот вариант для опытных пользователей. Вы можете установить VM с настройками по умолчанию.

Далее Мастер проверяет возможность установки VM на ПК, затем предлагает нам выбрать оборудование и указать источник для установки ОС (рис. 7.10 и рис. 7.11).

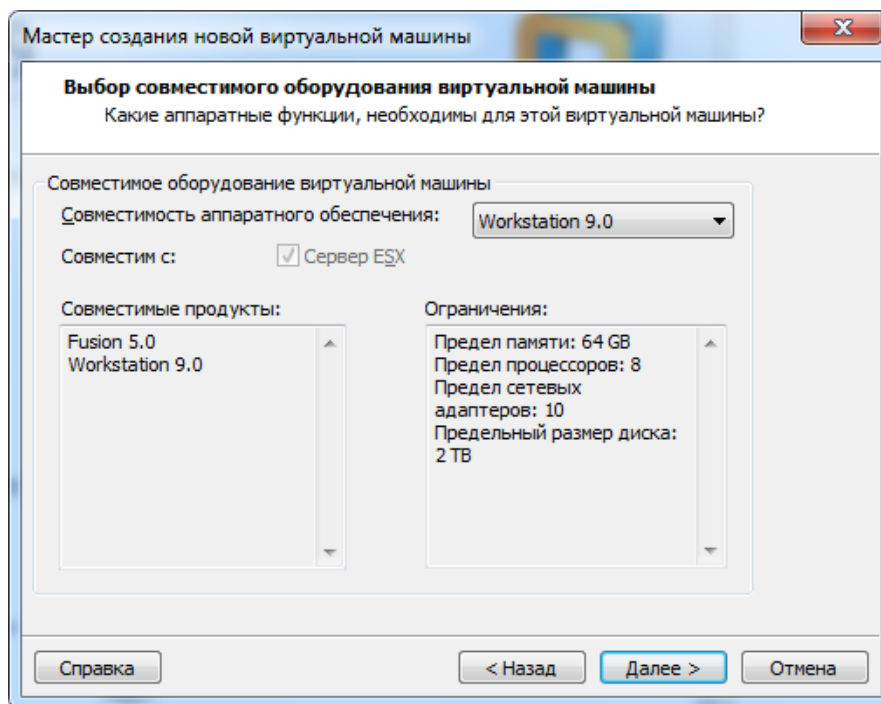


Рис. 7.10. Выбор совместимого оборудования ОС можно устанавливать с компакт диска или из ее ISO образа.

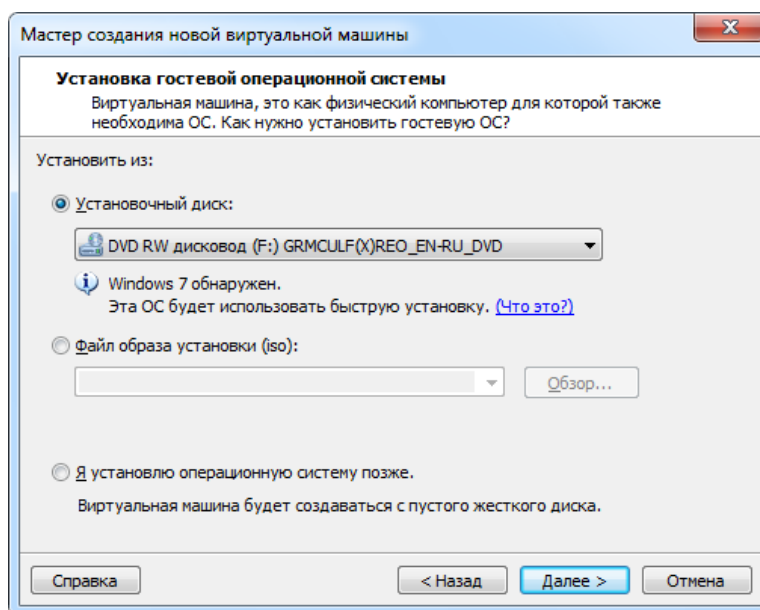


Рис. 7.11. Устанавливать систему будем с компакт-диска

Далее активируем ОС ключом и задаем имя машины (рис. 7.12).

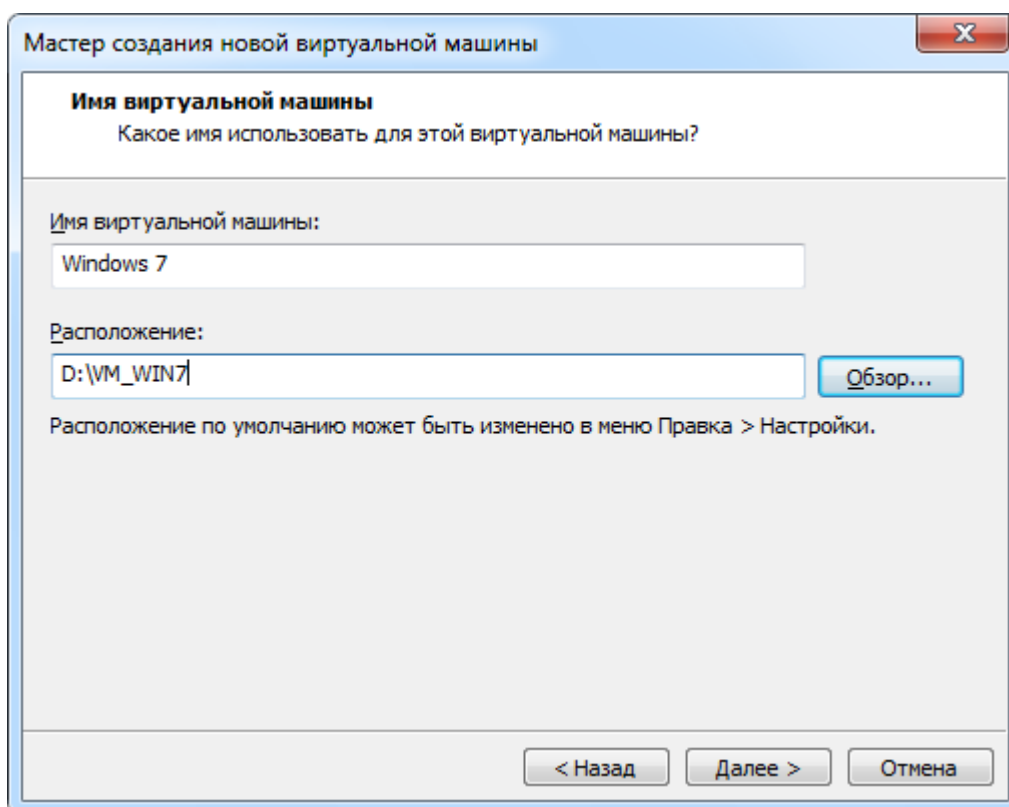


Рис. 7.12. Задаем имя машины и ее расположение
Задаем конфигурацию процессора (рис. 7.13).

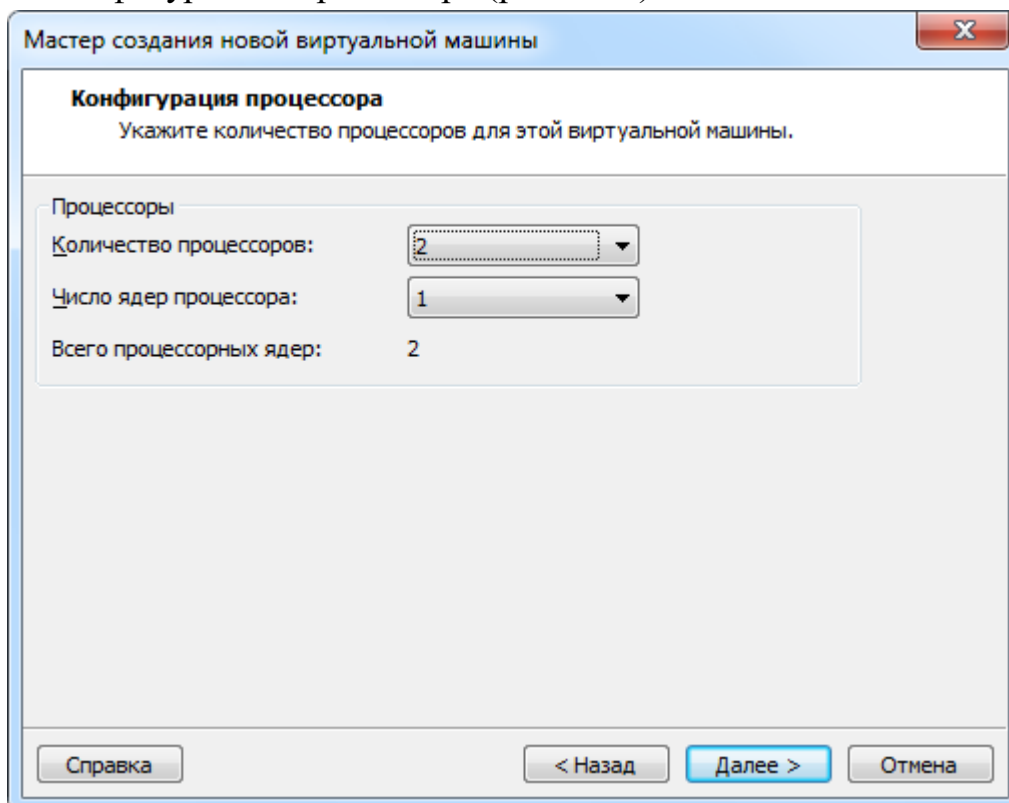


Рис. 7.13. Задаем конфигурацию процессора для VM

Остальные шаги Мастера сделаем с настройками по умолчанию. Сделаем комментарий только к окну, изображенному на рис. 7.14.

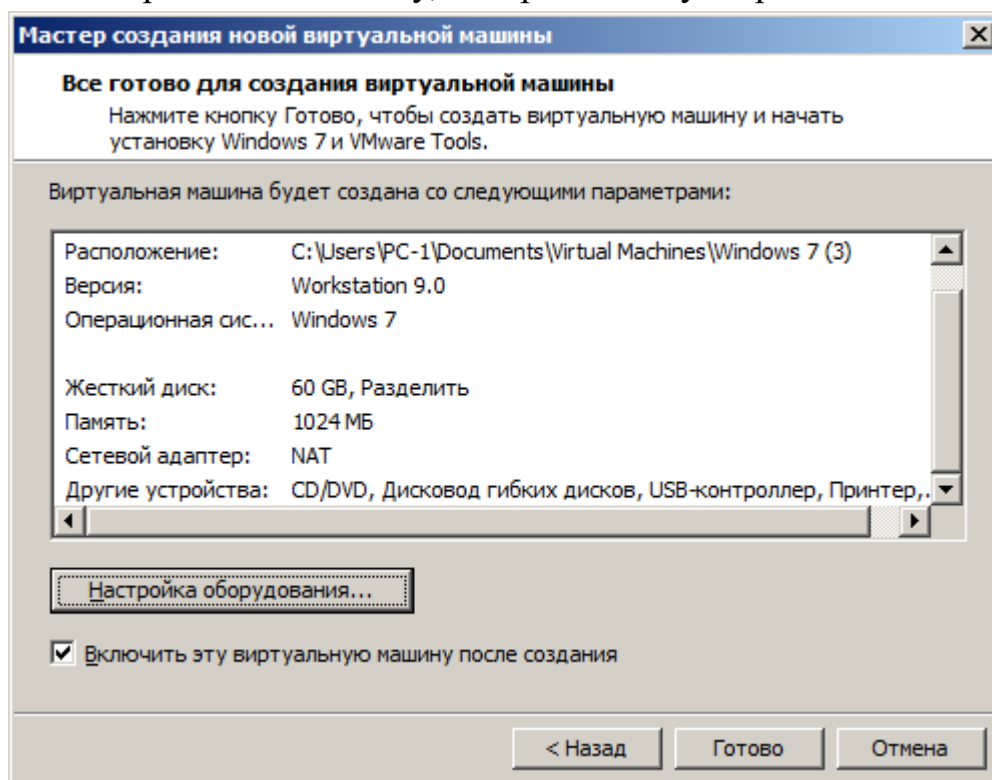


Рис. 7.14. Последнее окно Мастера создания новой виртуальной машины

В данном окне мы видим, что у нашего виртуального ПК будет сетевой *адаптер NAT* (*Network Address Translation* - технология преобразования сетевых адресов).

Примечание

При помощи механизма NAT несколько машин из одной сети могут выходить в другую сеть, в нашем случае — несколько машин из виртуальной локальной сети смогут выходить в глобальную сеть Интернет, используя только один IP адрес. Иначе говоря, вся сеть пользуется одним IP адресом. В нашем случае это будет IP адрес роутера (маршрутизатора), к которому подключен физический ПК (Позднее мы изобразим карту такой сети на рисунке). IP адреса пакетов из виртуальной локальной сети, проходя через NAT (в сторону Интернет), перезаписываются адресом внешнего сетевого интерфейса, а возвращаясь обратно (из Интернет в локальную сеть), на пакетах восстанавливается правильный (локальный) IP адрес машины, которая и послала исходный пакет данных. С точки зрения провайдера Интернет, в такой сети работает лишь одна машина (маршрутизатор с активированным на нем механизмом NAT), а все другие компьютеры, находящейся за маршрутизатором, для провайдера не видны совсем. Таким образом, получив лишь один IP адрес (одно подключение) от провайдера,

можно вывести в глобальную сеть несколько ПК. И такая локальная сеть автоматически защищается от злоумышленников, поскольку она им просто не видна (за исключением самого компьютера-маршрутизатора). Для подавляющего большинства программ механизм NAT полностью прозрачен, т.е. они его просто не заметят.

Итак, продолжим. Процесс установки *Windows 7* как на физический ПК, так и на виртуальный ПК полностью идентичен (рис. 7.15 и рис. 7.16).



Рис. 7.15. Окно начальной установки Windows 7 на виртуальный ПК

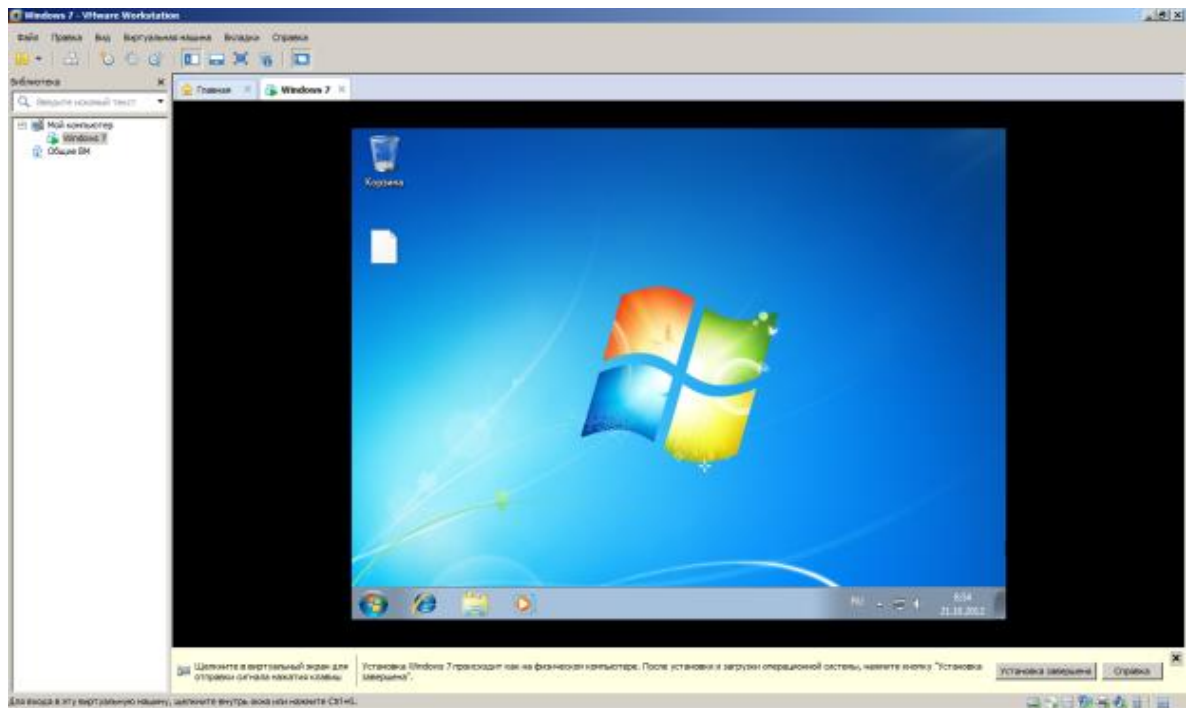


Рис. 7.16. Установка Windows 7 на виртуальный ПК завершена

В заключение выполним следующее: **Пуск-Панель управления-Учетные записи пользователей-Создание пароля своей учетной записи** (рис. 7.17).

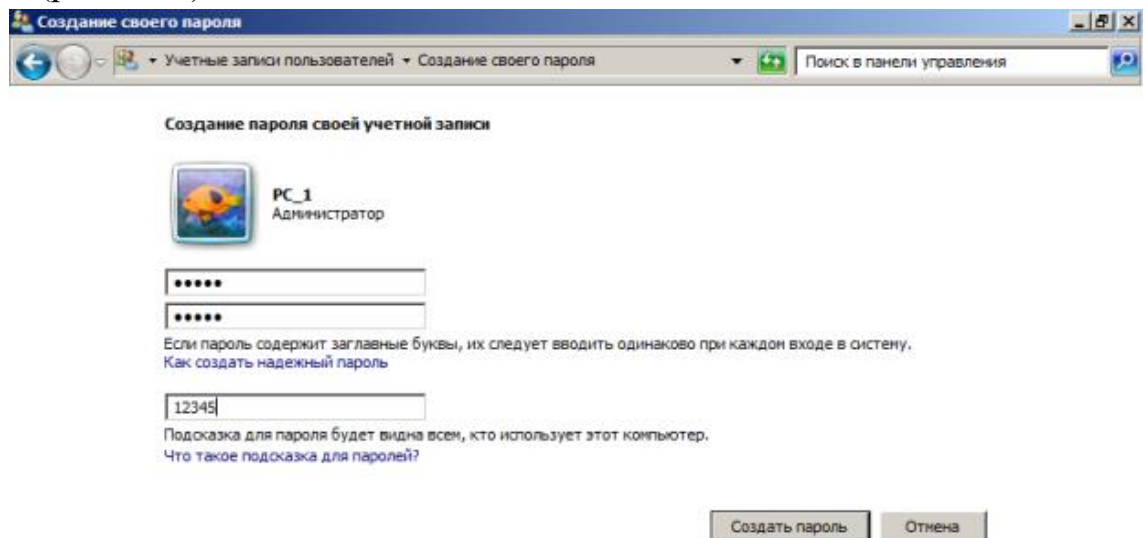


Рис. 7.17. Создание пароля своей учетной записи

И еще одна команда: **Панель управления-Система и безопасность-Система-Изменить параметры-Изменить**. Здесь мы включим наш ПК в рабочую группу (рис. 7.18).

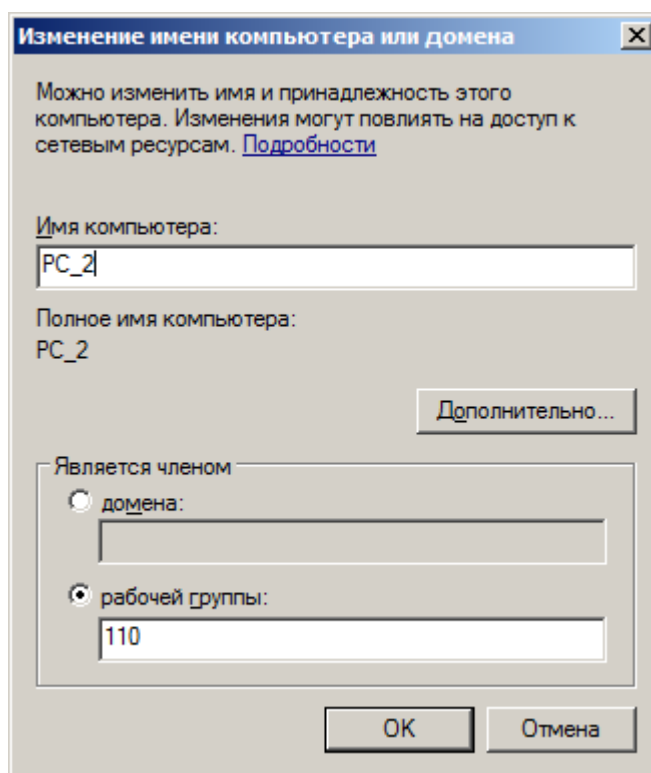


Рис. 7.18. Окно Изменение имени компьютера или домена

3. Клонирование виртуальной машины с ОС Windows 7

Создадим еще одну машину, для этого выполним команду **Виртуальная машина-Управление-Клонировать** (рис. 7.19).

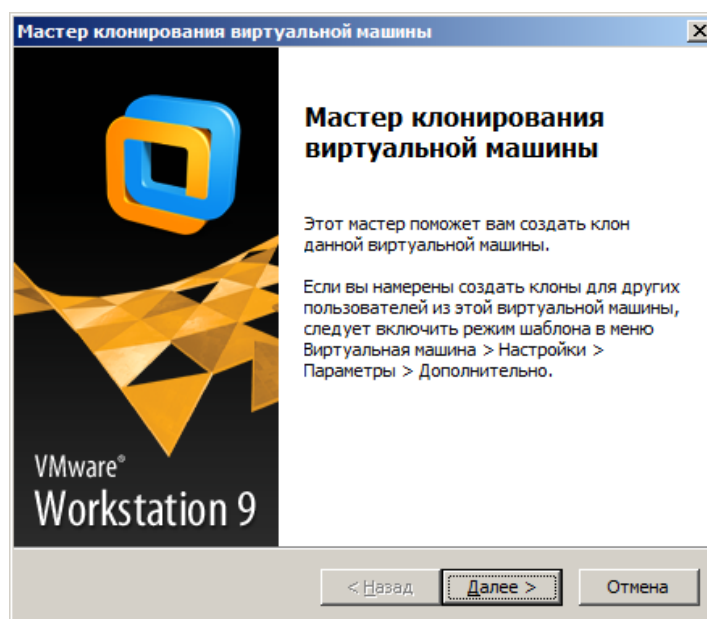


Рис. 7.19. Окно Мастер клонирование виртуальной машины

Далее мы покажем только те окна, где мы отклонились от шагов мастера по умолчанию (рис. 7.20).

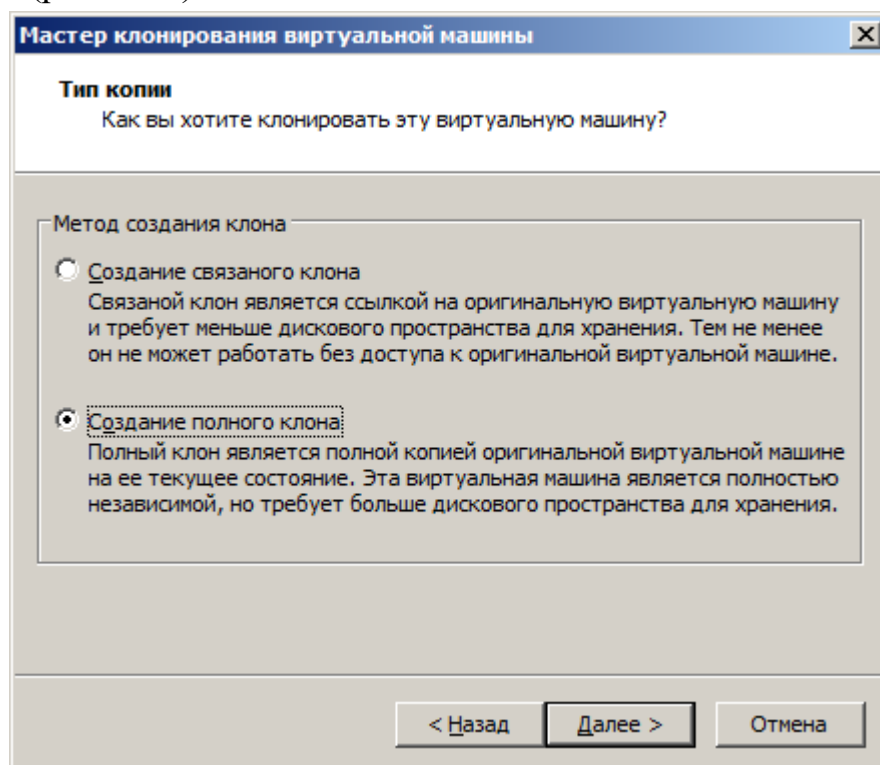


Рис. 7.20. Устанавливаем переключатель Создание полного клона
Клонирование – процесс существенно более быстрый, чем установка виртуальной машины с нуля (рис. 7.21).

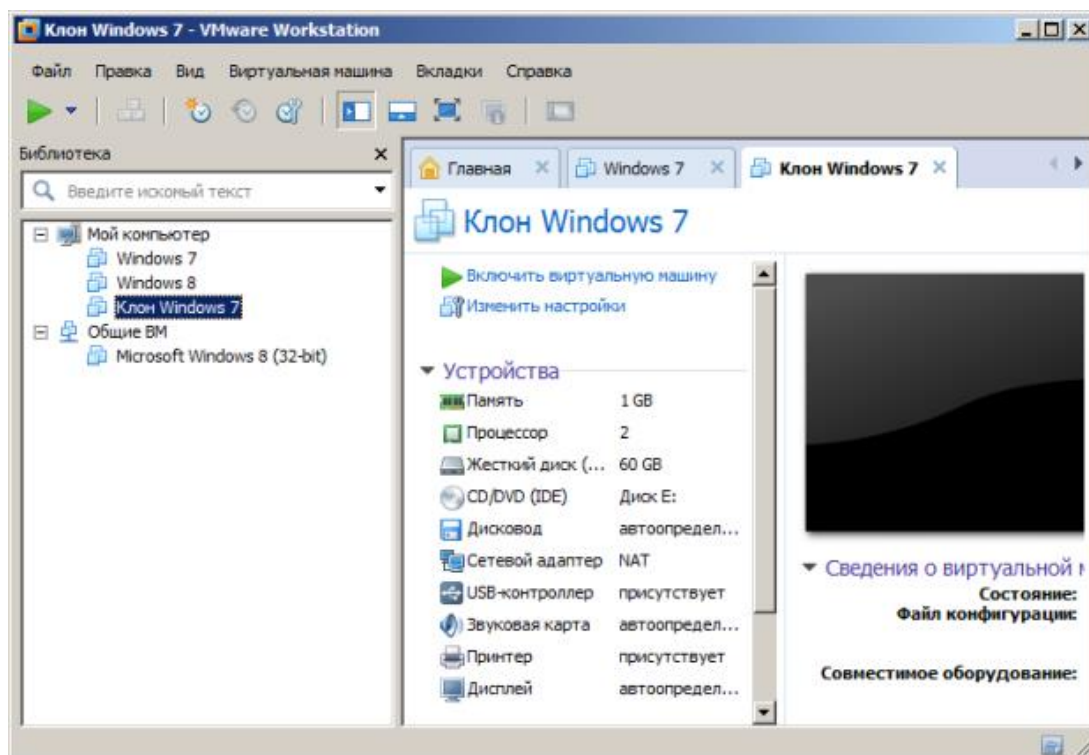


Рис. 7.21. Клон создан

Примечание

Мы также установили виртуальную машину на ОС Windows 8. Предлагаем вам сделать эту работу самостоятельно – ничего принципиально нового в этом процессе нет.

Установка средств Wmware

Чтобы получить *доступ* из виртуальной машины к файлам на физическом ПК потребуется команда **Виртуальная машина-Установить пакет Wmware Tools** (рис. 7.22).

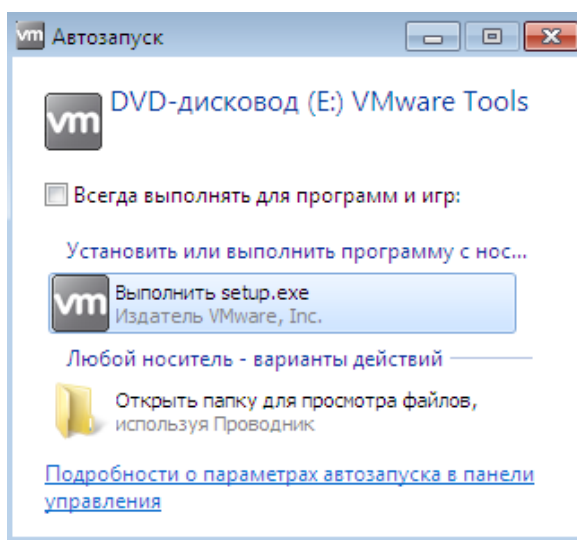


Рис. 7.22. Окно начала установки средств Wmware

После инсталляции средств и перезагрузки виртуальной машины выполним команду **Виртуальная машина-Параметры**, откроем вкладку **Параметры** и встанем курсором на строчку папок с общим доступом. Активируем *переключатель* **Всегда включено** (рис. 7.23).

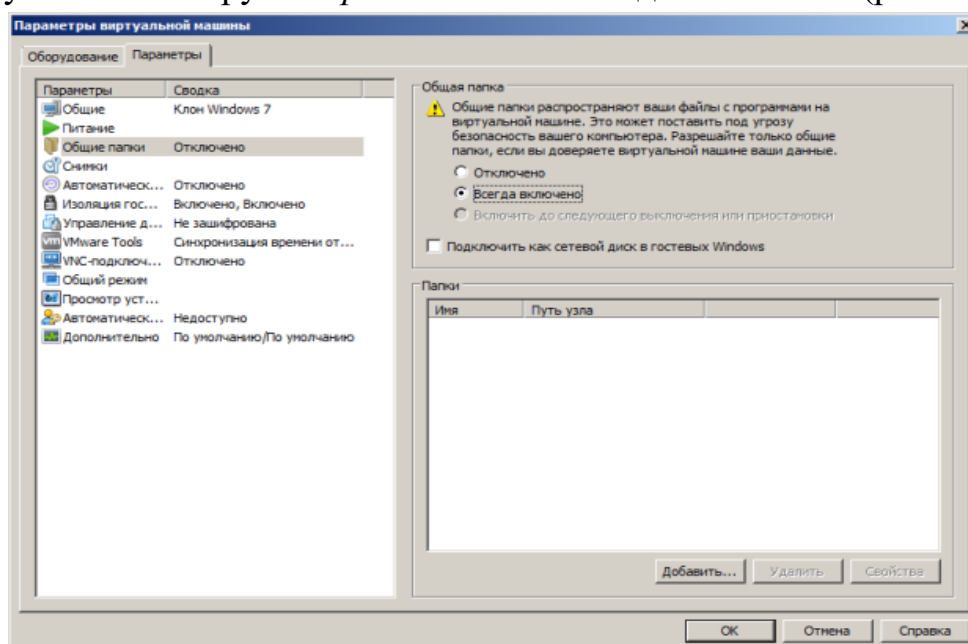


Рис. 7.23. Папки с общим доступом (общие папки) пока недоступны

Нажимаем на кнопку **Добавить** и на физическом ПК укажем папку, которую мы хотим сделать общей для физического и виртуального компьютеров (рис. 7.24).

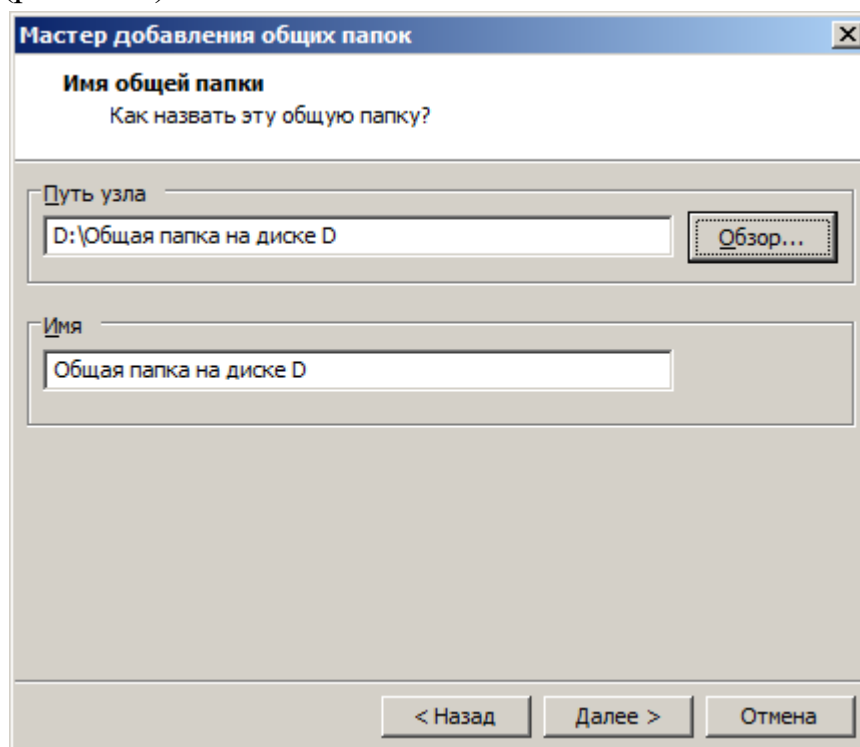


Рис. 7.24. Кнопкой **Обзор** находим нужную нам папку. Далее активируем атрибуты папки (рис. 7.25).

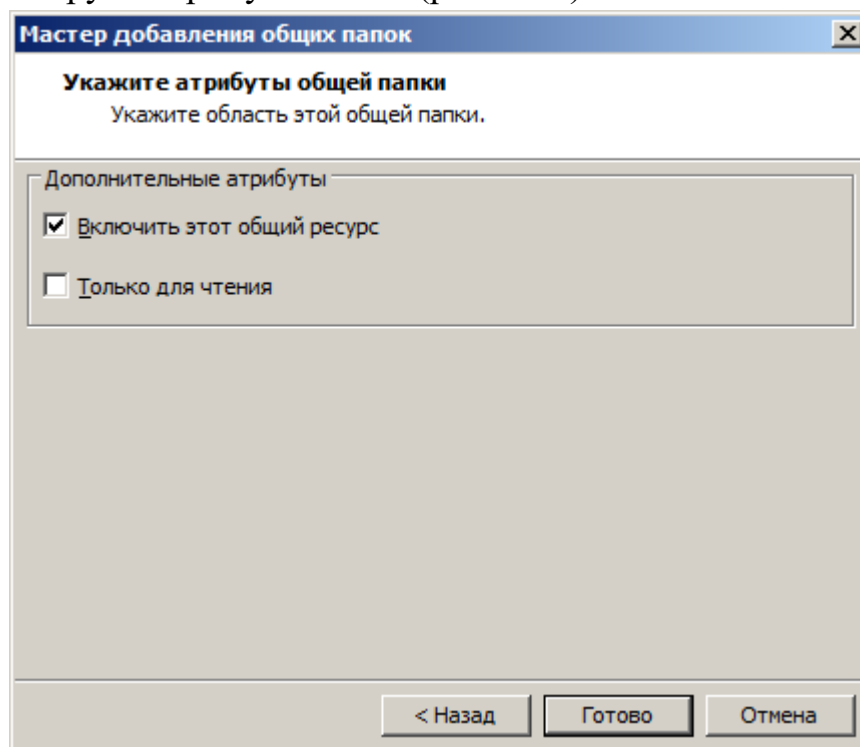


Рис. 7.25. В этом окне нам нужен верхний флажок

Теперь при просмотре всей сети мы увидим папку на нашем физическом ПК, т.е. у нас появилась *связь* физического ПК с виртуальными машинами (рис. 7.26).

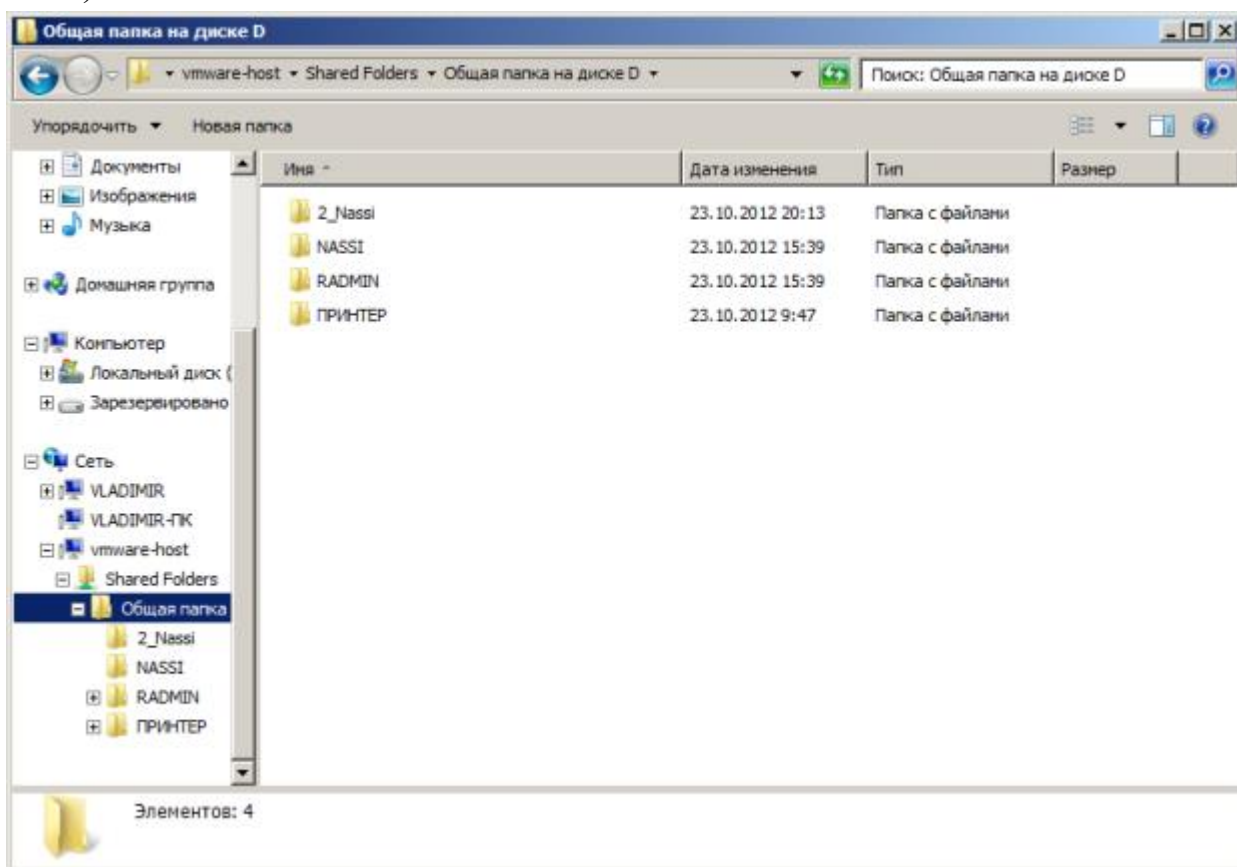


Рис. 7.26. Связь физической машины с виртуальной установлена

Краткие итоги

В этой работе мы создали виртуальную машину VMware Workstation 9 на физическом ПК и установили на ней ОС Windows 7. Далее научились производить операцию клонирования виртуальной машины, а также устанавливать на ней средства Wmware.

Лабораторная работа №3

Обеспечение безопасности локальной сети

Цели:

1. Ознакомиться с уязвимостями связанными с учетной записью и сетевых уязвимостей портов ПК.
2. Изучить возможности устранения уязвимостей связанными с учетной записью и сетевых уязвимостей портов ПК.
3. Выработать практические навыки работы с программами: NetStat Agent, сканер портов Nmap (Zenmap), монитор портов TCPView

Шаг 1. Меняем учетную запись администратора (Пользователь Администратор с пустым паролем - это уязвимость)

Часто при установке *Windows* пароль администратора пустой и этим может воспользоваться злоумышленник. Иначе говоря, при установке *Windows XP* в автоматическом режиме с настройками по умолчанию мы имеем пользователя **Администратор** с пустым паролем и любой **User** может войти в такой ПК с правами администратора. Чтобы решить проблему выполним команду **Мой компьютер-Панель управления-Администрирование-Управление компьютером-Локальные пользователи-Пользователи** (рис. 5.1).

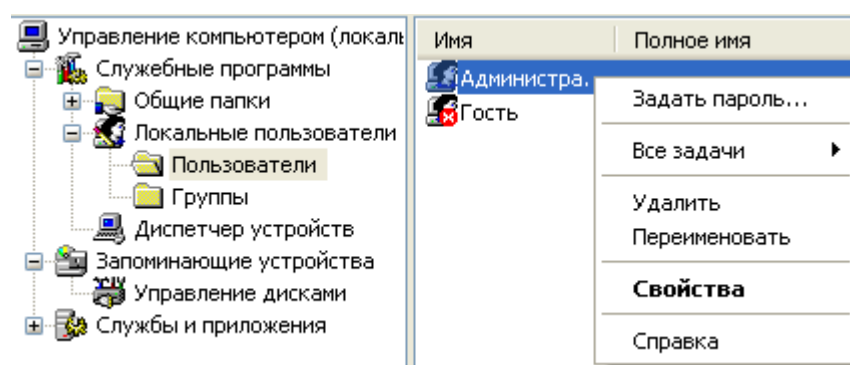


Рис. 5.1. Окно Управление компьютером

Здесь по щелчку правой кнопкой мыши на **Администраторы** зададим администратору *пароль*, например, 12345. Это плохой *пароль*, но лучше, чем ничего. Теперь в окне **Администрирование** зайдем в **Локальную политику безопасности**. Далее идем по веткам дерева: **Локальные политики-Параметры безопасности-Учетные записи: Переименование учетной записи Администратор** (рис. 5.2).

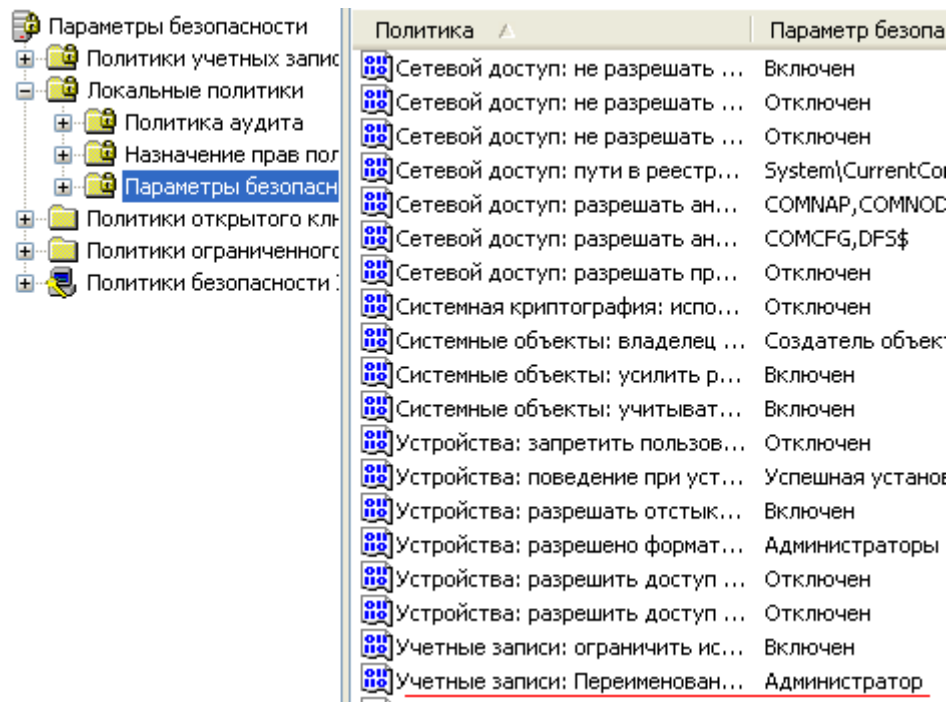


Рис. 5.2. Находим в системном реестре запись Переименование учетной записи Администратор

Здесь пользователя **Администратор** заменим на **Admin** (рис. 5.3).

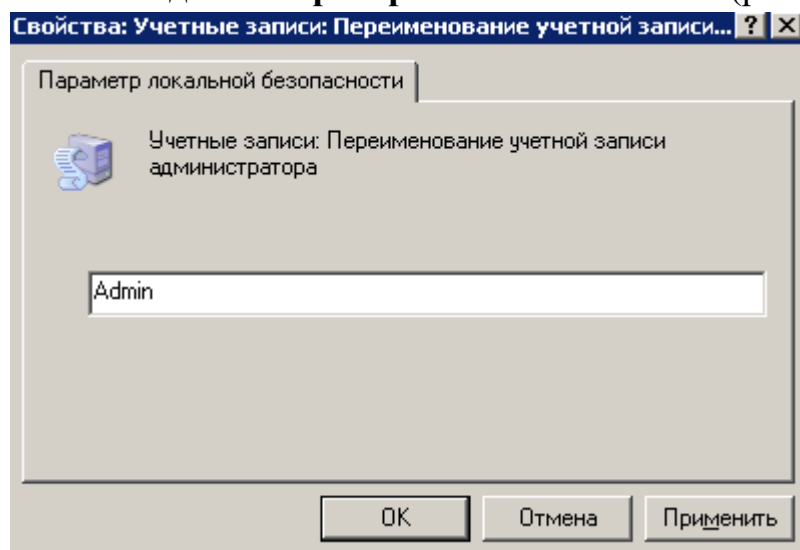


Рис. 5.3. Пользователю Администратор присваиваем новое имя
 Перезагружаем ОС. После наших действий у нас получилась учетная запись *Admin* с паролем 12345 и правами администратора (рис. 5.4).

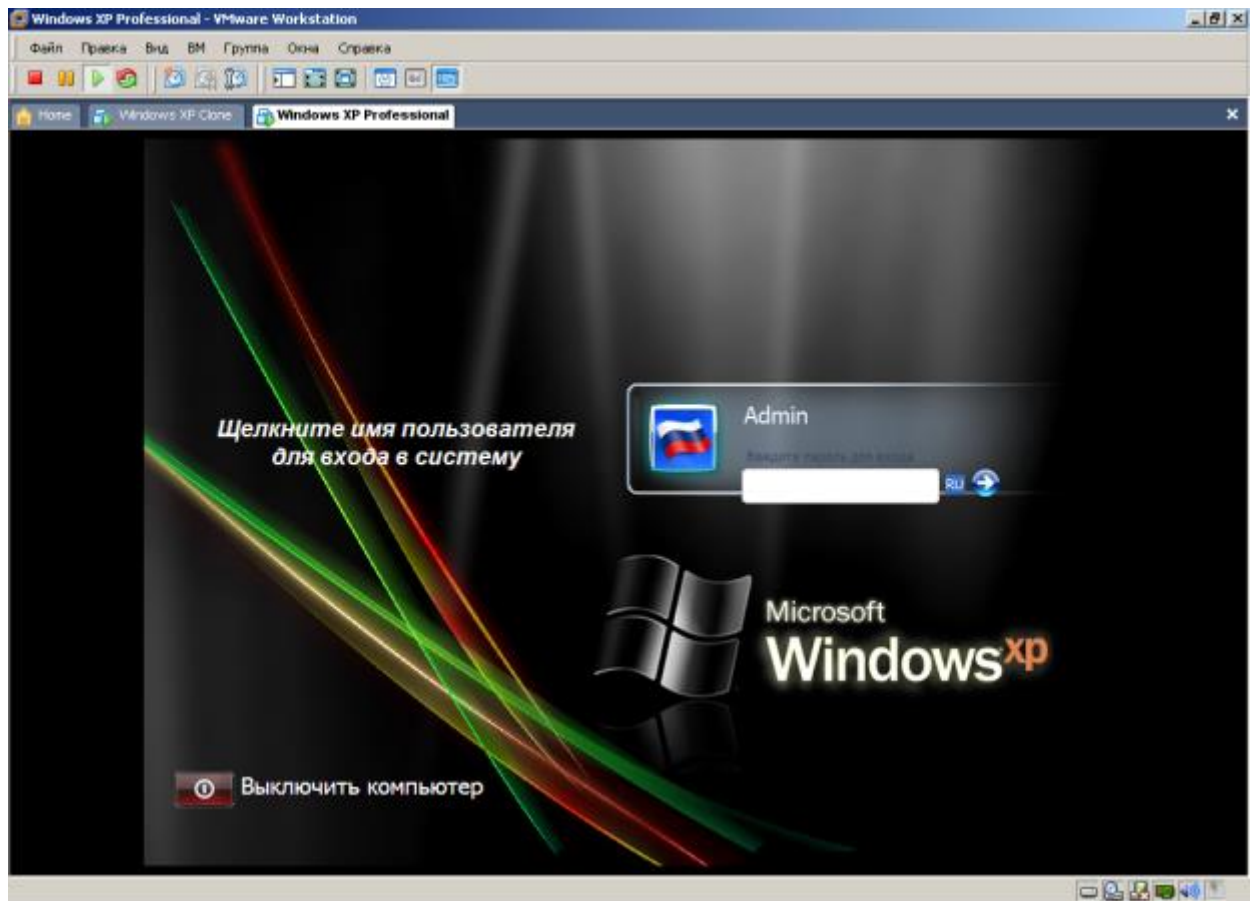


Рис. 5.4. Окно входа в ОС [Windows](#)

XP

Все, теперь мы имеем пользователя **Администратор** с паролем, одна из уязвимостей системы устранена.

Примечание

Операцию по изменению имени пользователя и заданию пароля мы также могли бы выполнить без использования системного реестра, использовав окно **Учетные записи пользователей**, что гораздо проще (рис. 5.5).

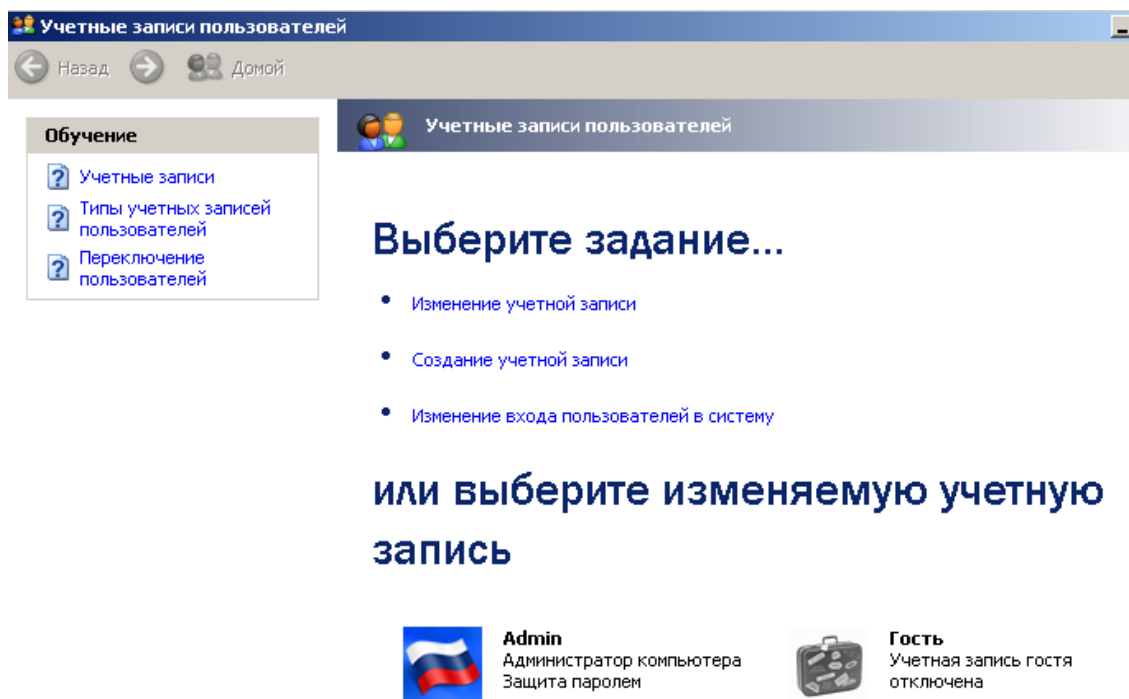


Рис. 5.5. Окно Учетные записи пользователей

Примечание

Учетная запись Гость позволяет входить в ПК и работать на нем (например, в Интернет) без использования специально созданной учетной записи. Запись Гость не требует ввода пароля и по умолчанию заблокирована. Гость не может устанавливать или удалять программы. Эту учетную запись можно отключить, но нельзя удалить.

Шаг 2. Делаем окно приветствия пустым (убираем уязвимость 2)

У нас окно входа в систему содержит подсказку *Admin*, давайте ее уберем, сделав окно пустым. Для начала в окне **Учетные записи пользователей** жмем на кнопку **Изменение входа пользователей в систему** и уберем флажок **Использовать страницу приветствия** (рис. 5.6 и рис. 5.7).

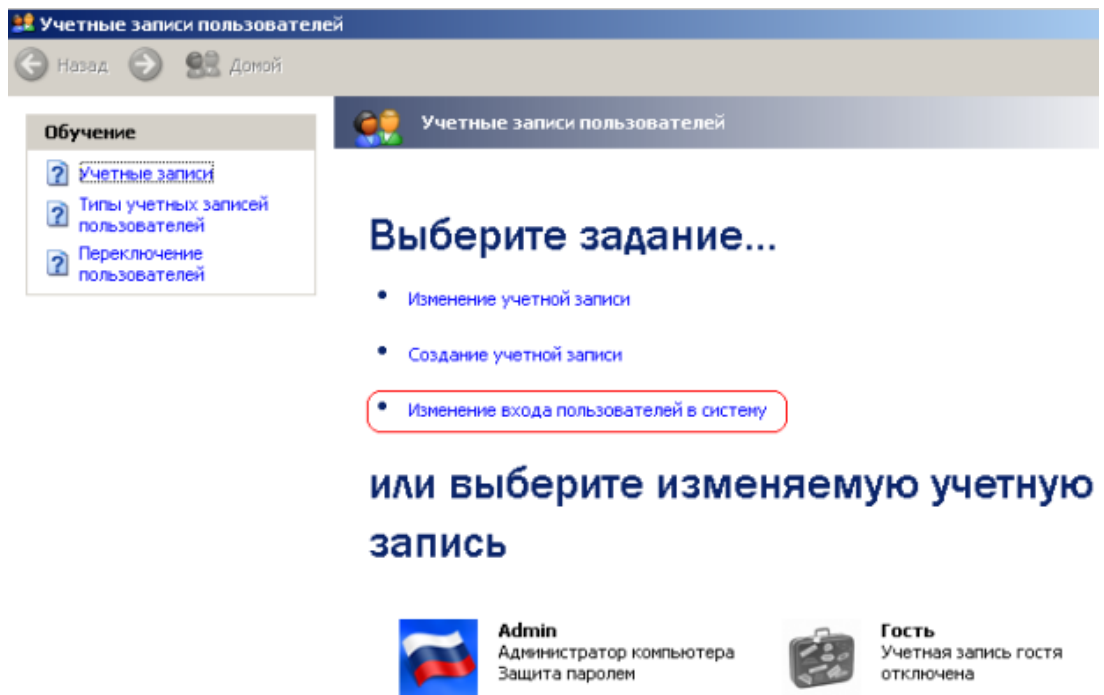


Рис. 5.6. Окно Учетные записи пользователей

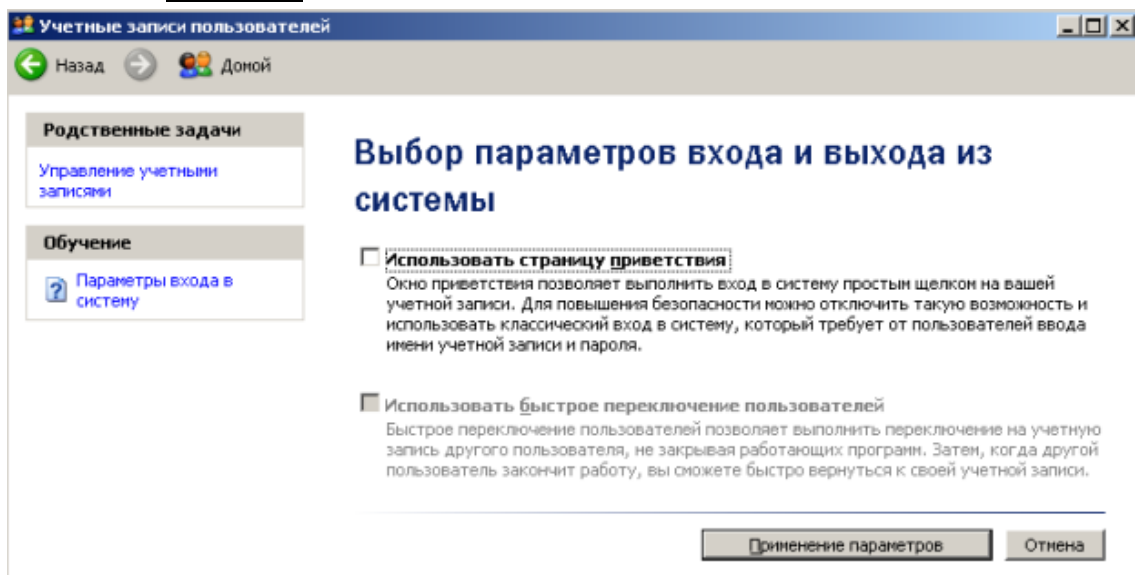


Рис. 5.7. Убираем флажок Использовать страницу приветствия

Но, это только половина дела. Теперь повысим *безопасность* сети еще на одну условную ступень, сделав оба поля окна приветствия пустыми (рис. 5.8).

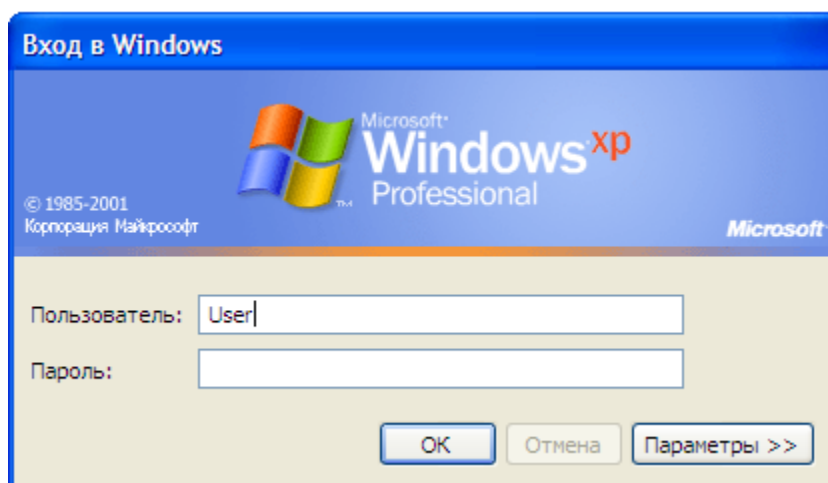


Рис. 5.8. Обе строки данного окна сделаем пустыми
 Выполним команду **Панель управления-Администрирование – Локальные политики безопасности- Локальные политики-Параметры безопасности-Интерактивный вход: не отображать последнего имени пользователя.** Эту запись необходимо включить (рис. 5.9).

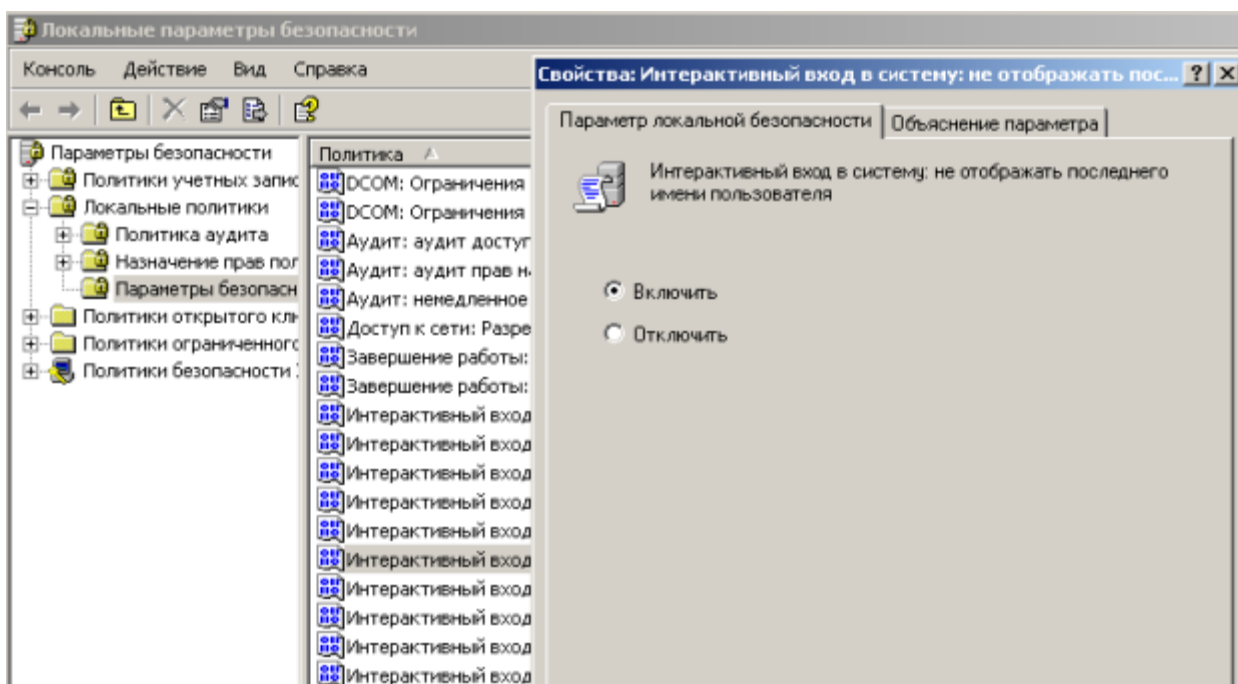


Рис. 5.9. Активируем переключатель Включить
 Теперь после завершения сеанса пользователь должен угадать не только пароль, но и имя пользователя (рис. 5.10).

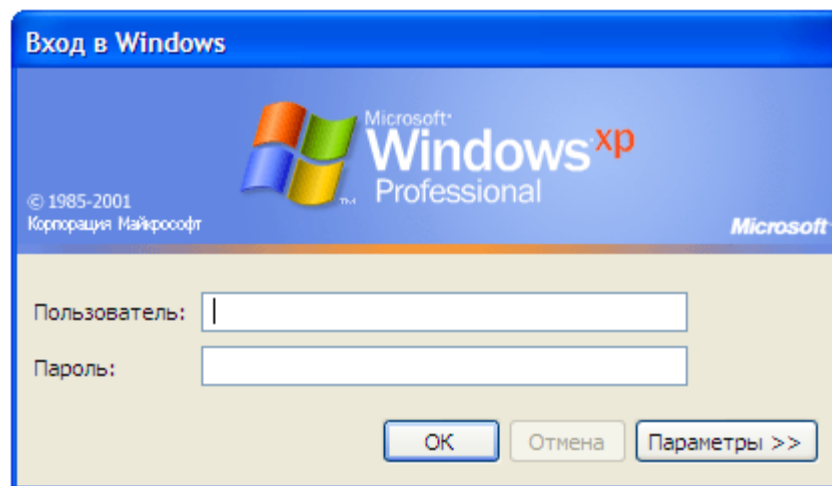


Рис. 5.10. Обе строки окна приветствия пусты

Выявление сетевых уязвимостей сканированием портов ПК

Злоумышленники используют сканирование портов ПК для того, чтобы воспользоваться ресурсами чужого ПК в Сети. При этом необходимо указать **IP адрес ПК** и **открытый port**, к примеру, **195.34.34.30:23**. После этого происходит соединение с удаленным ПК с некоторой вероятностью входа в этот ПК.

• **TCP/IP port** — это адрес определенного сервиса (программы), запущенного на данном компьютере в Internet. Каждый открытый порт — потенциальная лазейка для взломщиков сетей и ПК. Например, SMTP (отправка почты) — 25 порт, WWW — 80 порт, FTP — 21 порт.

• **Хакеры сканируют порты для того, чтобы найти дырку (баг) в операционной системе. Пример ошибки, если администратор или пользователь ПК открыл полный доступ к сетевым ресурсам для всех или оставил пустой пароль на вход к компьютер.**

Одна из функций администратора сети (сисадмина) - выявить недостатки в функционировании сети и устранить их. Для этого нужно просканировать *сеть* и закрыть (блокировать) все необязательные (открытые без необходимости) сетевые порты. Ниже, для примера, представлены службы *TCP/IP*, которые можно отключить:

- **finger** - получение информации о пользователях
- **talk**- возможность обмена данными по сети между пользователями
- **bootp** - предоставление клиентам информации о сети
- **systat** - получение информации о системе
- **netstat** - получение информации о сети, такой как текущие соединения
- **rusersd** - получение информации о пользователях, зарегистрированных

в данный момент

Просмотр активных подключений утилитой Netstat

Команда **netstat** обладает набором ключей для отображения портов, находящихся в активном и/или пассивном состоянии. С ее помощью можно получить список серверных приложений, работающих на данном компьютере. Большинство серверов находится в режиме **LISTEN** - ожидание запроса на соединение. Состояние **CLOSE_WAIT** означает, что соединение разорвано. **TIME_WAIT** - соединение ожидает разрыва. Если соединение находится в состоянии **SYN_SENT**, то это означает наличие процесса, который пытается, установить соединение с сервером. **ESTABLISHED** — соединения установлены, т.е. сетевые службы работают (используются).

Итак, команда **netstat** показывает содержимое различных структур данных, связанных с сетью, в различных форматах в зависимости от указанных опций. Для сокетов (программных интерфейсов) *TCP* допустимы следующие значения состояния

- **CLOSED** - Закрыт. Сокет не используется.
- **LISTEN** - Ожидает входящих соединений.
- **SYN_SENT** - Активно пытается установить соединение.
- **SYN_RECEIVED** - Идет начальная синхронизация соединения.
- **ESTABLISHED** - Соединение установлено.
- **CLOSE_WAIT** - Удаленная сторона отключилась; ожидание закрытия сокета.
- **FIN_WAIT_1** - Сокет закрыт; отключение соединения.
- **CLOSING** - Сокет закрыт, затем удаленная сторона отключилась; ожидание подтверждения.
- **LAST_ACK** - Удаленная сторона отключилась, затем сокет закрыт; ожидание подтверждения.
- **FIN_WAIT_2** - Сокет закрыт; ожидание отключения удаленной стороны.
- **TIME_WAIT** - Сокет закрыт, но ожидает пакеты, ещё находящиеся в сети для обработки

Примечание

Что такое "сокет" поясняет рис. 5.11. Пример сокета – 194.86.6..54:21

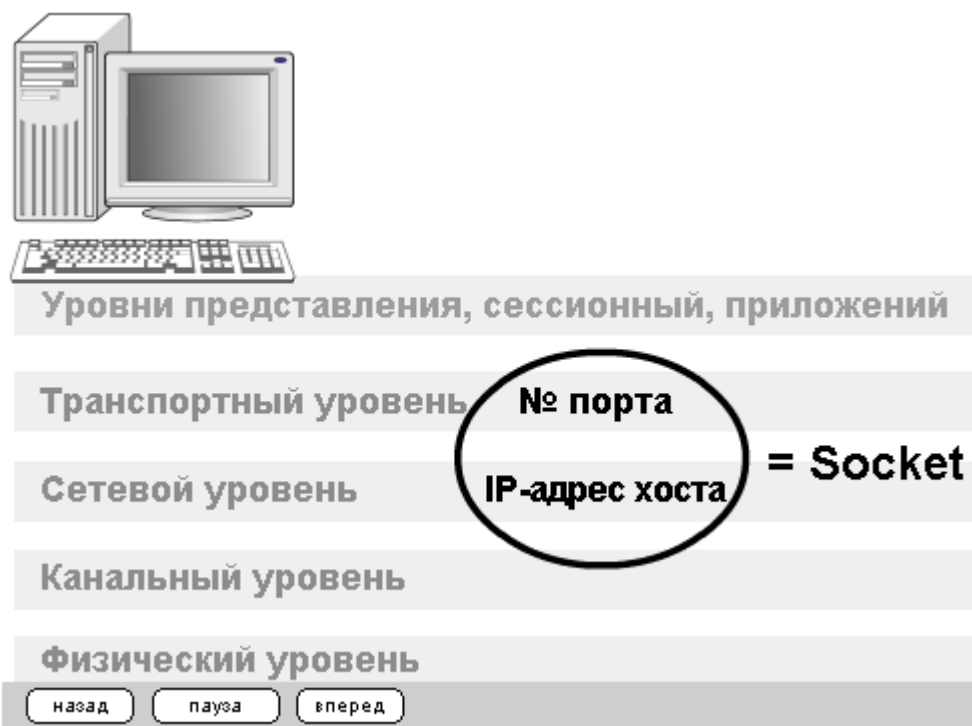


Рис. 5.11. Сокет это № порта + IP адрес хоста

Практический пример. Обнаружение открытых на ПК портов утилитой Netstat

Для выполнения практического задания на компьютере необходимо выполнить команду **Пуск-Выполнить**. Откроется окно **Запуск программы**, в нем введите команду **cmd** (рис. 5.12).

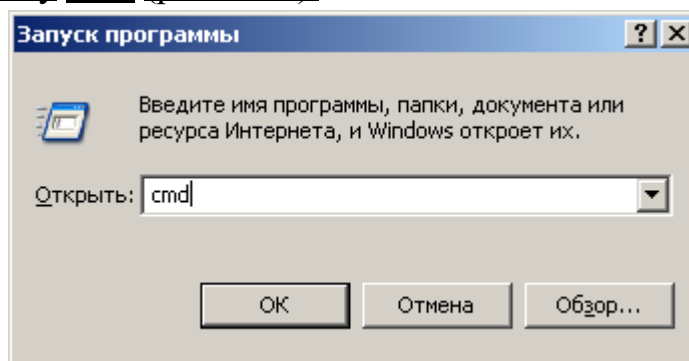


Рис. 5.12. Окно Запуск программы

Чтобы вывести все активные подключения *TCP* и прослушиваемые компьютером порты *TCP/UDP* введите команду **netstat** (рис. 5.13). Мы видим Локального адреса (это ваш ПК) прослушиваются 6 портов. Они нужны для поддержки сети. На двух портах мы видим режим **ESTABLISHED** — соединения установлены, т.е. сетевые службы работают (используются). Четыре порта используются в режиме **TIME WAIT** - соединение ожидает разрыва.

```

Активные подключения
Имя      Локальный адрес      Внешний адрес      Состояние
TCP      D:3086                localhost:3087     ESTABLISHED
TCP      D:3087                localhost:3086     ESTABLISHED
TCP      D:3414                localhost:1110     TIME_WAIT
TCP      D:3416                localhost:1110     TIME_WAIT
TCP      D:3415                OCSP.AMS1.VERISIGN.COM:http  TIME_WAIT
TCP      D:3417                OCSP.AMS1.VERISIGN.COM:http  TIME_WAIT

D:\Documents and Settings\110>

```

Рис. 5.13. Список активных подключений на тестируемом ПК. Запустите на вашем ПК Интернет и зайдите, например на www.yandex.ru. Снова выполните команду **netstat** (рис. 5.14). Как видим, добавилось несколько новых активных портов с их различными состояниями.

```

D:\Documents and Settings\110>netstat
Активные подключения
Имя      Локальный адрес      Внешний адрес      Состояние
TCP      D:1110                localhost:3433     TIME_WAIT
TCP      D:1110                localhost:3436     TIME_WAIT
TCP      D:1110                localhost:3441     TIME_WAIT
TCP      D:1110                localhost:3442     TIME_WAIT
TCP      D:1110                localhost:3443     TIME_WAIT
TCP      D:1110                localhost:3448     ESTABLISHED
TCP      D:1110                localhost:3452     TIME_WAIT
TCP      D:1110                localhost:3454     ESTABLISHED
TCP      D:1110                localhost:3456     TIME_WAIT
TCP      D:3430                localhost:3431     ESTABLISHED
TCP      D:3431                localhost:3430     ESTABLISHED
TCP      D:3432                localhost:1110     TIME_WAIT
TCP      D:3438                localhost:1110     TIME_WAIT
TCP      D:3440                localhost:1110     TIME_WAIT
TCP      D:3448                localhost:1110     ESTABLISHED
TCP      D:3450                localhost:1110     TIME_WAIT
TCP      D:3454                localhost:1110     ESTABLISHED
TCP      D:3458                localhost:1110     TIME_WAIT
TCP      D:3460                localhost:1110     TIME_WAIT
TCP      D:3461                localhost:1110     TIME_WAIT
TCP      D:3462                localhost:1110     TIME_WAIT
TCP      D:3434                addons-star.zlb.phx.mozilla.net:https  TIME_WAIT

TCP      D:3445                static.yandex.net:http  TIME_WAIT
TCP      D:3449                mc.yandex.ru:http      ESTABLISHED
TCP      D:3455                suggest.yandex.net:http  ESTABLISHED
TCP      D:3463                suggest.yandex.net:http  TIME_WAIT
TCP      D:3464                www.yandex.ru:http     TIME_WAIT
TCP      D:3465                yabs.yandex.ru:http    TIME_WAIT

```

Рис. 5.14. Активные подключения при работе ПК в Интернет. Команда **netstat** имеет следующие опции – табл. 10.1.

Таблица 5.1. Ключи для команды netstat

<u>Опция</u> (ключ)	<u>Назначение</u>
<u>-a</u>	<u>Показывать состояние всех сокетов; обычно сокет, используемый серверными процессами, не показывается.</u>
<u>-A</u>	<u>Показывать адреса любых управляющих блоков протокола, связанных с сокетами; используется для отладки.</u>

<u>-i</u>	<u>Показывать состояние автоматически сконфигурированных (auto-configured) интерфейсов. Интерфейсы, статически сконфигурированные в системе, но не найденные во время загрузки, не показываются.</u>
<u>-n</u>	<u>Показывать сетевые адреса как числа. netstat обычно показывает адреса как символы. Эту опцию можно использовать с любым форматом показа.</u>
<u>-r</u>	<u>Показать таблицы маршрутизации. При использовании с опцией -s, показывает статистику маршрутизации.</u>
<u>-s</u>	<u>Показать статистическую информацию по протоколам. При использовании с опцией -r, показывает статистику маршрутизации.</u>
<u>-f</u> <u>семейство адресов</u>	<u>Ограничить показ статистики или адресов управляющих блоков только указанным семейством адресов, в качестве которого можно указывать:</u> <u>inet Для семейства адресов AF_INET,</u> <u>или unix Для семейства адресов AF_UNIX.</u>
<u>-I интерфейс</u>	<u>Выделить информацию об указанном интерфейсе в отдельный столбец; по умолчанию (для третьей формы команды) используется интерфейс с наибольшим объёмом переданной информации с момента последней перезагрузки системы. В качестве интерфейса можно указывать любой из интерфейсов, перечисленных в файле конфигурации системы, например, emd1 или lo0.</u>
<u>-p</u>	<u>Отобразить идентификатор/название процесса создавшего сокет (-p, --programs display PID/Program name for sockets)</u>

Программа NetStat Agent

Представьте ситуацию: ваше Интернет-соединение стало работать медленно, компьютер постоянно что-то качает из Сети. Вам поможет программа **NetStat Agent**. С ее помощью вы сможете найти причину проблемы и заблокировать ее. Иначе говоря, **NetStat Agent** - полезный набор инструментов для мониторинга Интернет соединений и диагностики сети. Программа позволяет отслеживать TCP и UDP соединения на ПК, закрывать нежелательные соединения, завершать процессы, обновлять и

освобождать DHCP настройки адаптера, просматривать сетевую статистику для адаптеров и TCP/IP протоколов, а также строить графики для команд Ping и TraceRoute (рис. 5.15).

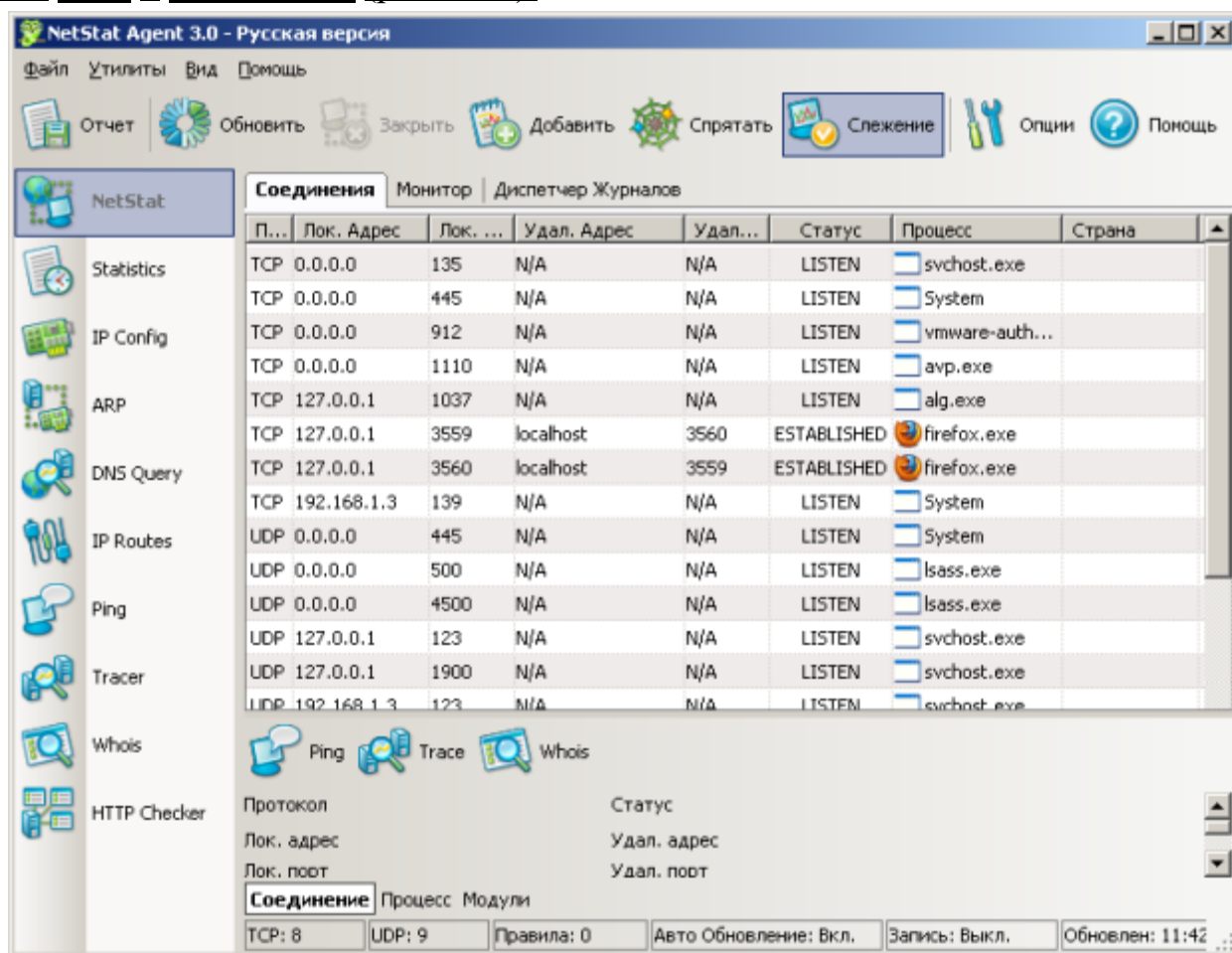


Рис. 5.15. Главное окно программы NetStat Agent

В состав программы NetStat Agent вошли следующие утилиты:

- **NetStat** - отслеживает TCP и UDP соединения ПК (при этом отображается географическое местоположение удаленного сервера и имя хоста).
- **IPConfig** - отображает свойства сетевых адаптеров и конфигурацию сети.
- **Ping** - позволяет проверить доступность хоста в сети.
- **TraceRoute** - определяет маршрут между вашим компьютером и конечным хостом, сообщая все IP-адреса маршрутизаторов.
- **DNS Query** - подключается к DNS серверу и находит всю информацию о домене (IP адрес сервера, MX-записи (Mail Exchange) и др.).
- **Route** - отображает и позволяет изменять IP маршруты на ПК.
- **ARP** - отслеживает ARP изменения в локальной таблице.
- **Whois** - позволяет получить всю доступную информацию об IP-адресе или домене.

- **HTTP Checker** - помогает проверить, доступны ли Ваши веб-сайты.
- **Statistics** - показывает статистику сетевых интерфейсов и TCP/IP протоколов.

Сканер портов Nmap (Zenmap)

Nmap - популярный сканер портов, который обследует сеть и проводит аудит защиты. Использовался в фильме "*Матрица: Перезагрузка*" при взломе компьютера. Наша задача не взломать, а защитить ПК, поскольку одно и то же оружие можно использовать как для защиты, так и для нападения. Иначе говоря, сканером портов **nmap** можно определить открытые порты компьютера, а для безопасности сети пользователям рекомендуется закрыть доступ к этим портам с помощью брандмауэра (рис. 5.16).

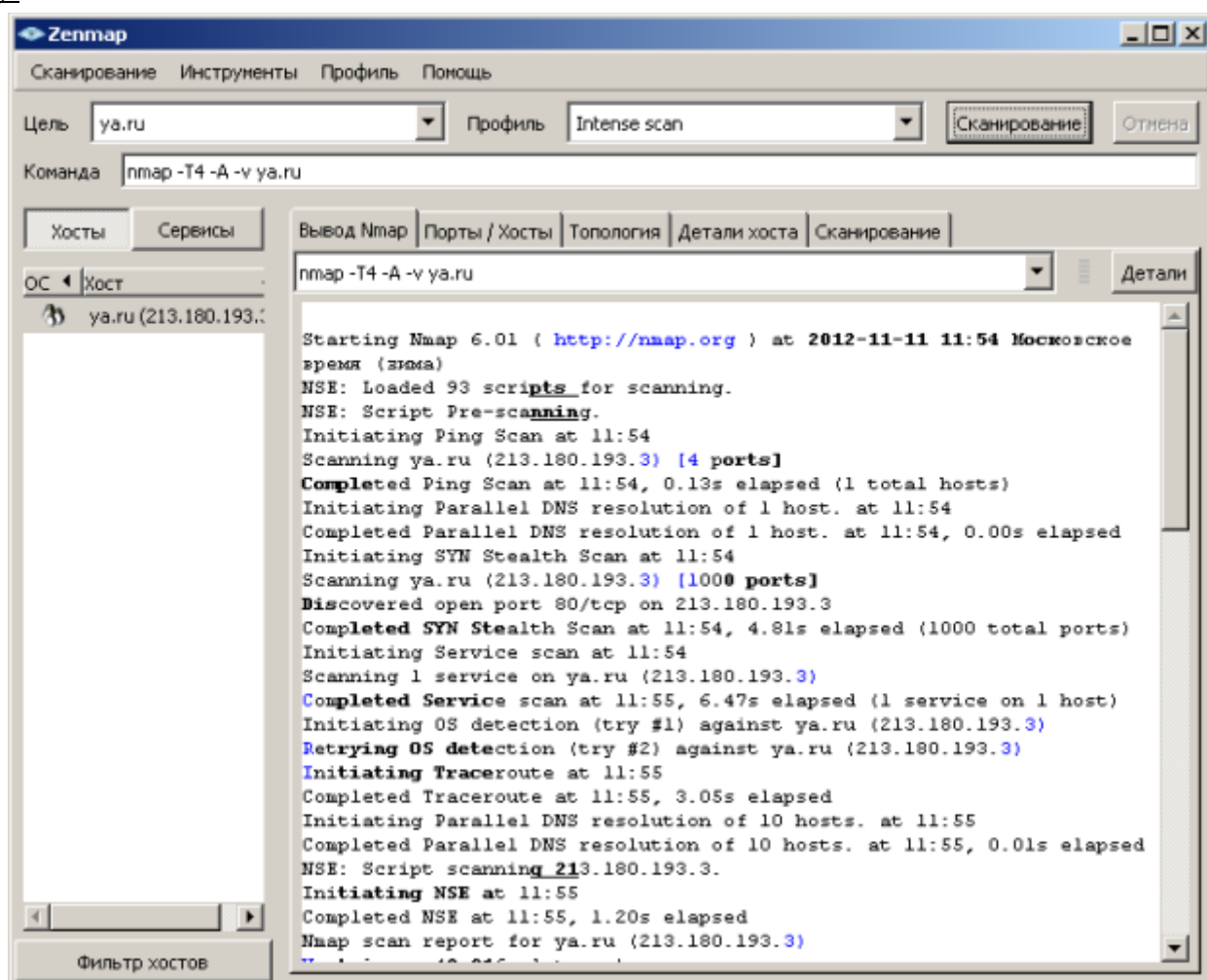


Рис. 5.16. Интерфейс программы Nmap

Обычно для того, чтобы просканировать все порты какого-либо компьютера в сети вводится команда **nmap -p1-65535 IP-адрес компьютера** или **nmap -sV IP-адрес компьютера**, а для сканирования сайта - команда **nmap -sS -sV -O -P0 адрес сайта**.

Монитор портов TCPView

TCPView - показывает все процессы, использующие Интернет-соединения. Запустив TCPView, можно узнать, какой порт открыт и какое приложение его использует, а при необходимости и немедленно разорвать соединение – рис. 5.17.

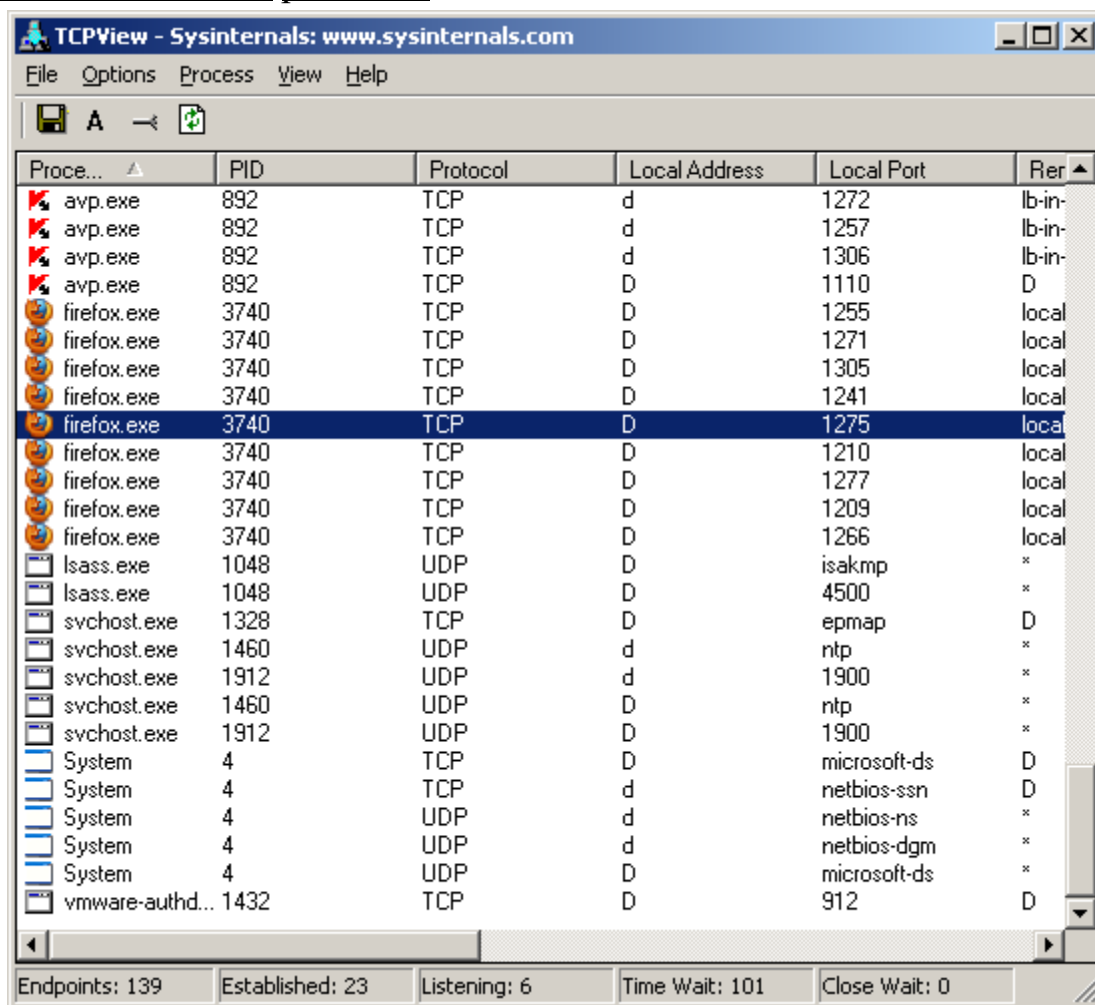


Рис. 5.17. Главное окно программы TCPView

Просмотрите активные сетевые подключения локального ПК с помощью монитора портов triview. Определите потенциально возможные угрозы (какие порты открыты, и какие приложения их используют). При необходимости можно закрыть установленное приложением TCP-соединение или процесс правой кнопкой мыши

Образец офисной политики безопасности

Информационная безопасность – огромная тема, здесь мы только немного прикоснулись к ней. Вот только несколько положений из правил поведения сотрудников предприятий малого бизнеса (рис. 5.18).

Документы категории "Комплект документов по информационной безопасности для малого бизнеса"

Д	П (ПРОДОЛЖЕНИЕ)	П (ПРОДОЛЖЕНИЕ)
<ul style="list-style-type: none"> ▪ Должностная инструкция инженера-электроника отдела информационных технологий (малый бизнес) ▪ Должностная инструкция начальника отдела информационных технологий (малый бизнес) 	<ul style="list-style-type: none"> ▪ Положение об использовании мобильных устройств и носителей информации (малый бизнес) ▪ Положение об использовании программного обеспечения (малый бизнес) ▪ Положение об использовании сети Интернет (малый бизнес) ▪ Положение об использовании электронной почты (малый бизнес) 	<ul style="list-style-type: none"> ▪ Положение об отделе информационных технологий (малый бизнес)
П	Т	
<ul style="list-style-type: none"> ▪ Положение о конфиденциальной информации (малый бизнес) ▪ Положение об использовании информационной системы (малый бизнес) 	<ul style="list-style-type: none"> ▪ Трудовой договор (малый бизнес) 	

Рис. 5.18. Список документов по информационной безопасности для малого бизнеса

Пользователям не разрешается устанавливать на компьютерах и в сети Компании программное обеспечение без разрешения системного администратора. Пользователи не должны пересылать электронную почту другим лицам и организациям без разрешения отправителя. Пользователям запрещается изменять и копировать файлы, принадлежащие другим пользователям, без разрешения владельцев файлов. Пользователь несет ответственность за сохранность своих паролей для входа в систему. Запрещается распечатывать, хранить в сети или передавать другим лицам индивидуальные пароли. И так далее...

Краткие итоги

В лабораторной работы мы научились убирать две уязвимости ОС Меняем учетную запись администратора (Пользователь Администратор с пустым паролем - это уязвимость) скринкаст 1

Шаг 2. Делаем окно приветствия пустым (убираем уязвимость 2) скринкаст 4

Выявление сетевых уязвимостей сканированием портов ПК 9

Просмотр активных подключений утилитой Netstat 9

Пример 1. Обнаружение открытых на ПК портов утилитой Netstat 10

Программа NetStat Agent 12

Сканер портов Nmap (Zenmap) 14

Монитор портов TCPView 15

Образец офисной политики безопасности

Контрольные задания:

1. Запущена только командная строка с командой netstat, без браузеров, мессенджеров торрентов и любых других программ использующих интернет.

2. Запущена командная строка с командой netstat с запущенным браузером, с открытой страницей любой поисковой системы.

Лабораторная работа №4

Изучение компьютерных сетей в программе S2 Netest

Эмулятор сети S2 Netest

Ранее мы уже познакомились с программой для изучения и моделирования компьютерных сетей NetEmul. Однако, *программа S2 Netest* отлична от нее по своим возможностям. Так, например, в ней нет анимации, зато появилась возможность моделировать беспроводные сети, а также такое понятие, как варианты проектирования сети (оптимальный и не оптимальный). Изначально *программа S2 Netest* создавалась в учебных целях, для того, чтобы учащиеся могли визуализировать работу компьютерных сетей и для облегчения понимания студентами происходящих в сетях процессов. В S2 Netest можно проверить свои знания в создании локальных сетей из доступных элементов сети. К сожалению, *программа* работает только на Windows XP, а на Windows 7 не запускается. Это можно исправить запуском в режиме совместимости. Если будут ошибки с библиотекой «msvbvm50.dll», скачать и установить ее можно на сайте <http://dlltop.ru/m/123-msvbvm50-dll> или <https://www.microsoft.com/en-us/download/details.aspx?id=24417>

Сетевое оборудование

В программе S2 Netest вам придется работать со следующим оборудованием. Сетевой адаптер с функцией Wake-On-LAN Complex RE100ATX/WOL RTL TP, скорость 10/100 Мбит в сек, PCI, socket for BootRom. Режим передачи полный и полудуплекс (рис. 6.1).



Рис. 6.1. Сетевой адаптер

В режиме дуплекс устройства могут передавать и принимать информацию одновременно. Полудуплекс - *передача данных*, когда данные

могут передаваться в обоих направлениях, но в каждый момент времени - только в одну сторону.

Следующий *тип оборудования - точка доступа*. Производитель - COMPEX. Разъемы-RJ-45 10/100 Мбит/сек. Максимальная скорость беспроводной передачи данных при соединении между беспроводными узлами: 11/5.5/2/1 Мбит/с. Частота беспроводной связи 2.4 - 2.497 ГГц. Радиус действия 25 - 100 м в помещении, 100 - 250 м на открытом пространстве (рис. 6.2).



Рис. 6.2. Точка доступа

Точка доступа применяется как для подключения группы компьютеров (каждый с беспроводным сетевым адаптером) в самостоятельные сети (**режим Ad-hoc**), так и для выполнения функции моста между беспроводными и кабельными участками сети (**режим Infrastructure**). Для режима Ad-hoc максимально возможное количество станций — 256. В *Infrastructure*-режиме допустимо до 2048 беспроводных узлов.

Асорп *Ethernet Hub 5 Port (5UTP)* - **концентратор** для небольшой рабочей группы, производитель –Асорп, тип сети *Ethernet*. Кол-во базовых портов 5. *Среда передачи - Ethernet 10baseT, скорость передачи* до 100 Мбит/сек, *длина сегмента* до 100 м. Интерфейсы 4 x *Ethernet 10baseT • RJ-45 (базовый порт)*. *Ethernet 10baseT • RJ-45 (uplink / базовый порт)*. *Ethernet 10base2 • BNC* (рис. 6.3).



Рис. 6.3. Асорп Ethernet Hub 5 Port (5UTP)

Trendnet *NWay switch 8 port* - **неуправляемый коммутатор** с двумя *Gigabit Ethernet* портами. *Интерфейс* - стандарт - *IEEE 802.3 10Base-T, IEEE 802.3u 100Base-TX, IEEE 802.3ab 1000Base-T, ANSI/IEEE 802.3 NWay auto-negotiation, IEEE 802.3x Flow Control*. Кол-во портов 8 x *RJ-*

45 скорость 10/100 Мбит. сек. + 2 x RJ-45 10/100/1000 Мбит/сек. Буфер 256К на устройство. Режим дуплекса - полу- / полный (рис. 6.4).



Рис. 6.4. Восемипортовый коммутатор

Кабель UTP (*unshielded twisted pair*) 5е - неэкранированная витая пара – рис. 6.5. Кабель категории 5-Е предназначен для передачи сигналов с частотой до 100 МГц. Он используется почти во всех современных приложениях для передачи речи, данных и видео. Минимальный срок службы кабеля UTP 5-Е - 10 лет



Рис. 6.5. Витая пара

Главное окно программы (Интерфейс)

Главное окно программы показано на рис. 6.6.

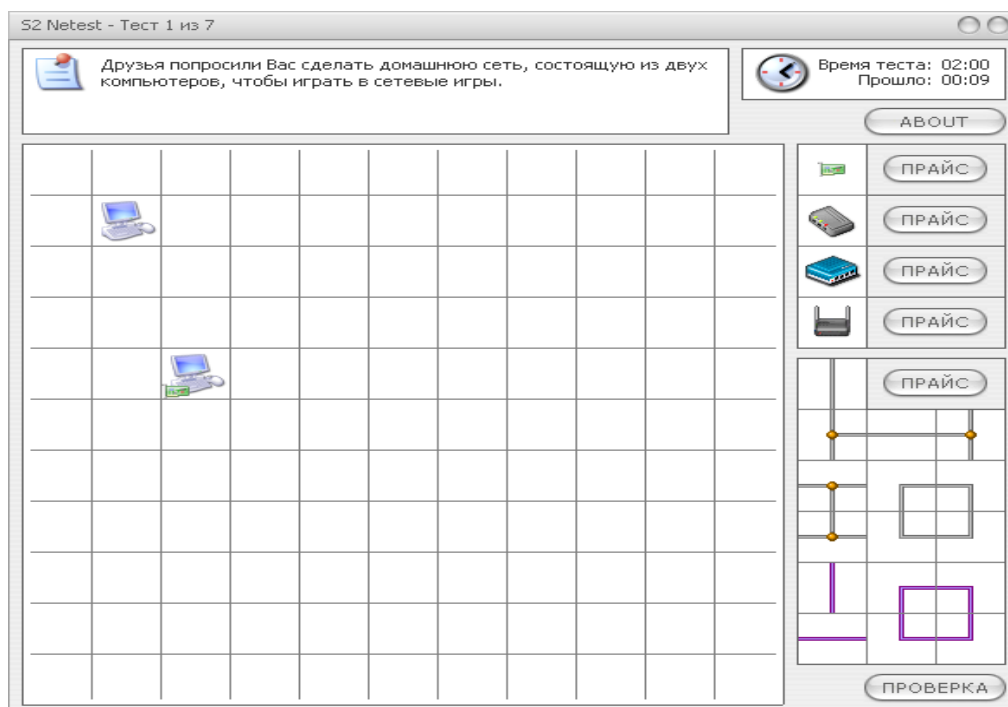











Рис. 6.6. Главное окно программы S2 Netest

Вверху главного окна программы находится *поле* задания текущего

теста  и *поле* , указывающее время, отведенное на тест, и время, прошедшее с начала выполнения текущего теста. Ниже поля задания располагается *поле*, представляющее собой квадрат, поделенный на 121 часть (*поле* из клеток 11x11). Справа от игрового поля находятся элементы сети (сетевое оборудование и кабели двух видов) с помощью которых необходимо смонтировать локальную *сеть*. В самом низу справа расположена кнопка **Проверка**, нажатием на которую можно проверить правильность смонтированной сети.

На *поле* из  клеток могут находиться компьютеры , принтеры  и сетевое оборудование. Ваша задача, перетаскивая мышью элементы сети, объединить все компьютеры в одну локальную *сеть*.

В начале нужно установить в каждый *компьютер* сетевую карту . Перетащите мышью сетевую карту на *компьютер*, и она появится слева от компьютера . Затем возьмите необходимое количество пятипортовых хабов (учитывая количество портов на каждом хабе и количество соединений). Теперь осталось все устройства соединить кабелем. Если прокладка кабеля затруднена, т.е. имеются препятствия в виде серых клеток , то необходимо воспользоваться беспроводным решением .

.Примеры выполнения тестов

Чтобы объединить два компьютера между собой в *сеть* дополнительных устройств, кроме сетевых карт, не требуется. Достаточно соединить компьютеры так называемым перекрестным кабелем (его еще называют "перевернутая витая пара") – рис. 6.7.

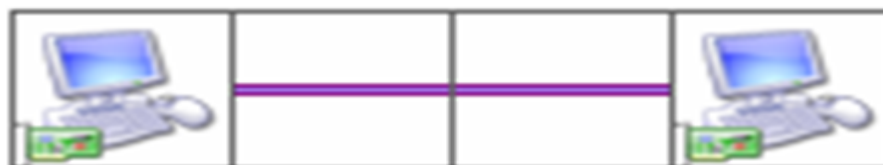


Рис. 6.7. Прямое соединение двух ПК перекрестным кабелем (красного цвета)

Если же компьютеров три и более, то необходимо использовать *хаб*. В этом случае используется *прямой кабель* (рис. 6.8).



Рис. 6.8. Соединение ПК посредством прямого кабеля (серого цвета)

Хаб имеет ограниченное количество портов для подключения компьютеров, поэтому в программе реализована возможность создания сети с использованием нескольких хабов ([рис. 6.9](#)).

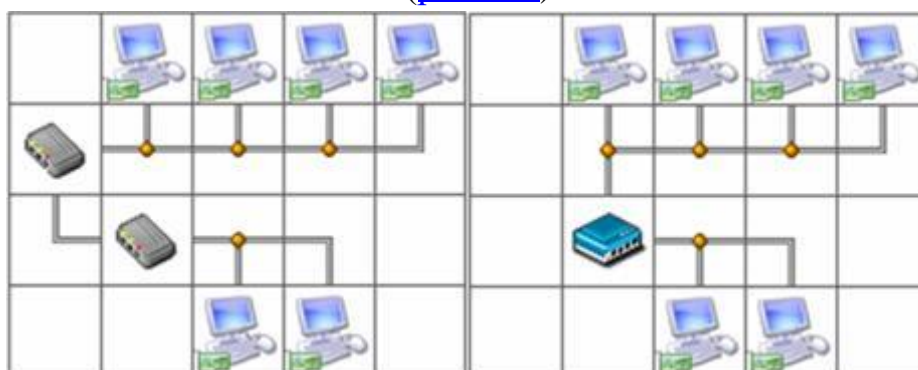


Рис. 6.9. Соединение двух хабов между собой (слева) и решение той же задачи посредством восьмипортового свитча

Между собой хабы соединяются либо **перевернутым кабелем** через обычный *порт*, либо **прямым кабелем** через порт **Up-Link** (в программе применяется именно такой принцип). Соединение хабов между собой T-образным кабелем в программе недопустимо!

Up-link служит для соединения одного свича (хаба) с другим (в обычный *порт*) обычным кабелем (не кроссовером). *Uplink-порт* обычно имеет *разъем* типа *RJ-45*. Наличие нескольких высокоскоростных *uplink-портов* позволяет построить локальную *сеть* по более сложной топологии. Иначе говоря, *Uplink* служит для каскадного соединения свичей или хабов, у которых нет автоопределения варианта обжима кабеля.

Если соединению ПК кабелем мешают различные препятствия, то можно организовать беспроводную сеть (рис. 6.10). Соединение точки беспроводного доступа с хабом T-образным кабелем недопустимо.

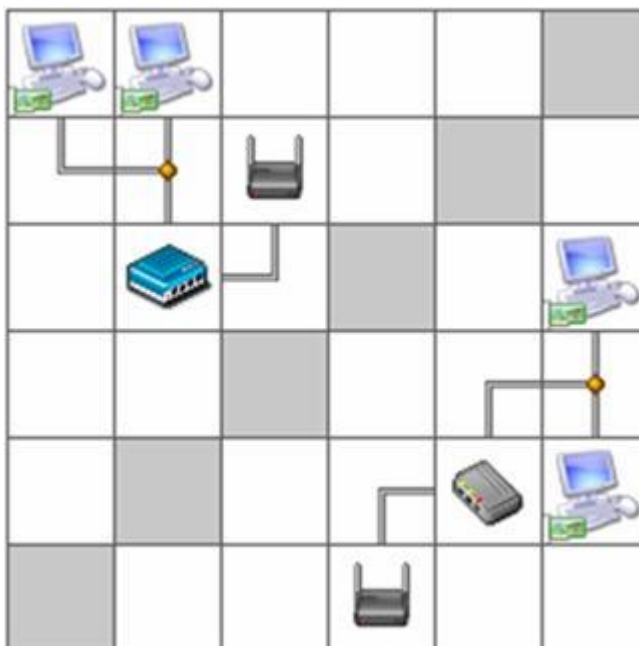


Рис. 6.10. Пример создания беспроводной сети

Сетевые решения оптимальные и неудачные

При старте программы вам предлагается выполнить, так называемый, стандартный тест (рис. 6.11).

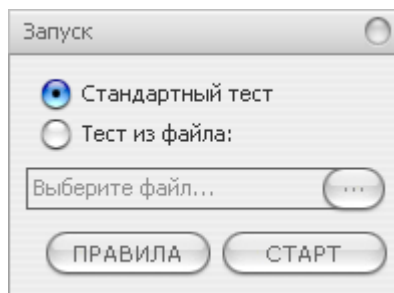


Рис. 6.11. Переключатель программы установлен на запуск стандартного теста

Пример правильного решения стандартного теста программы приведен на рис. 6.12. ниже:

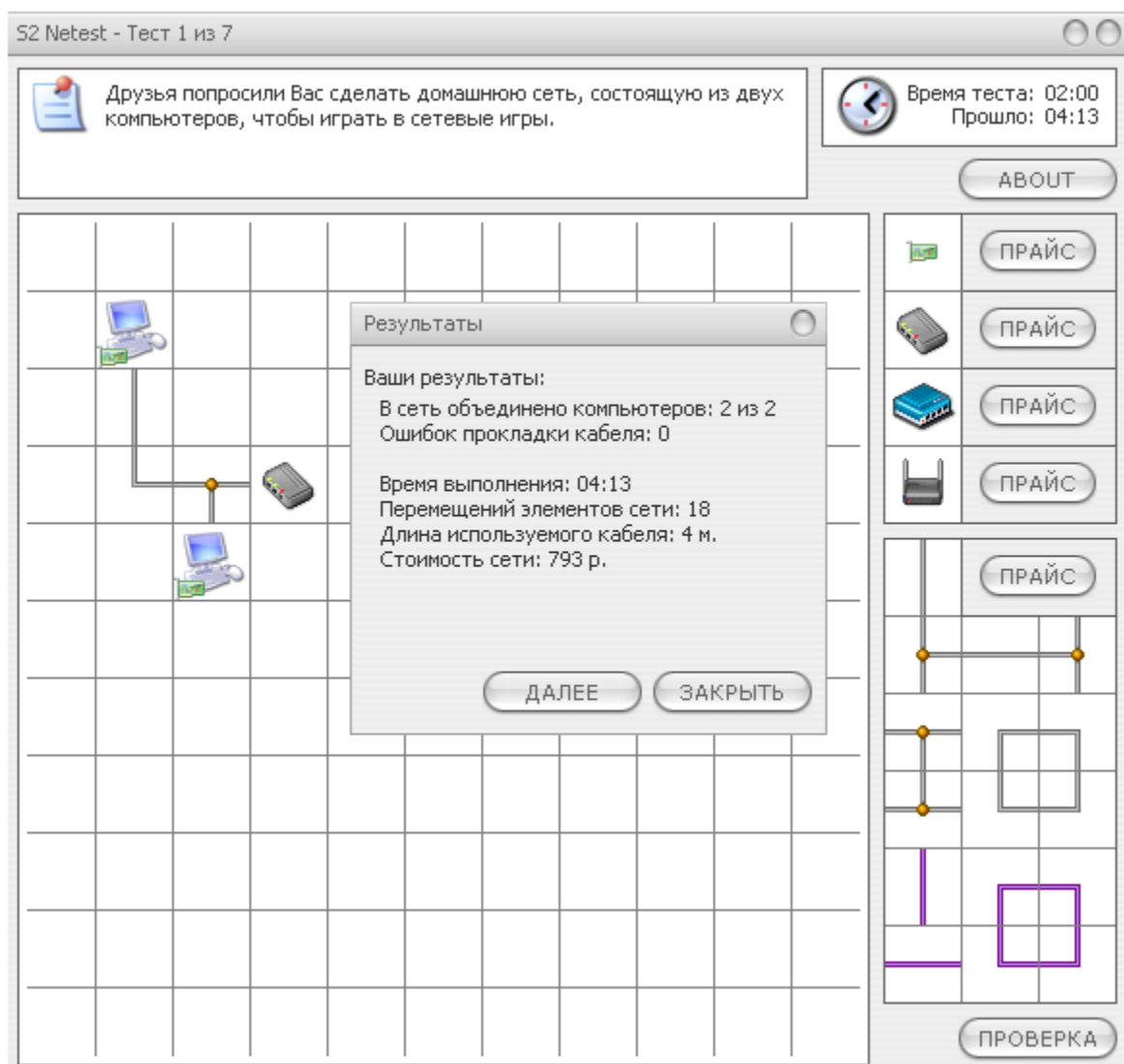


Рис. 6.12. Неоптимальное (неудачное) решение стандартного теста (793 руб)

Как видим ошибок в прокладке кабеля нет. Однако, оценку ОТЛИЧНО за такое, в принципе правильное решение получить нельзя. Поскольку существует более оптимальное решение этой задачи (рис. 6.13). Вариант в 793 руб. можно заменить более экономичным вариантом в 228 руб. с использованием не прямого, а перекрестного кабеля.

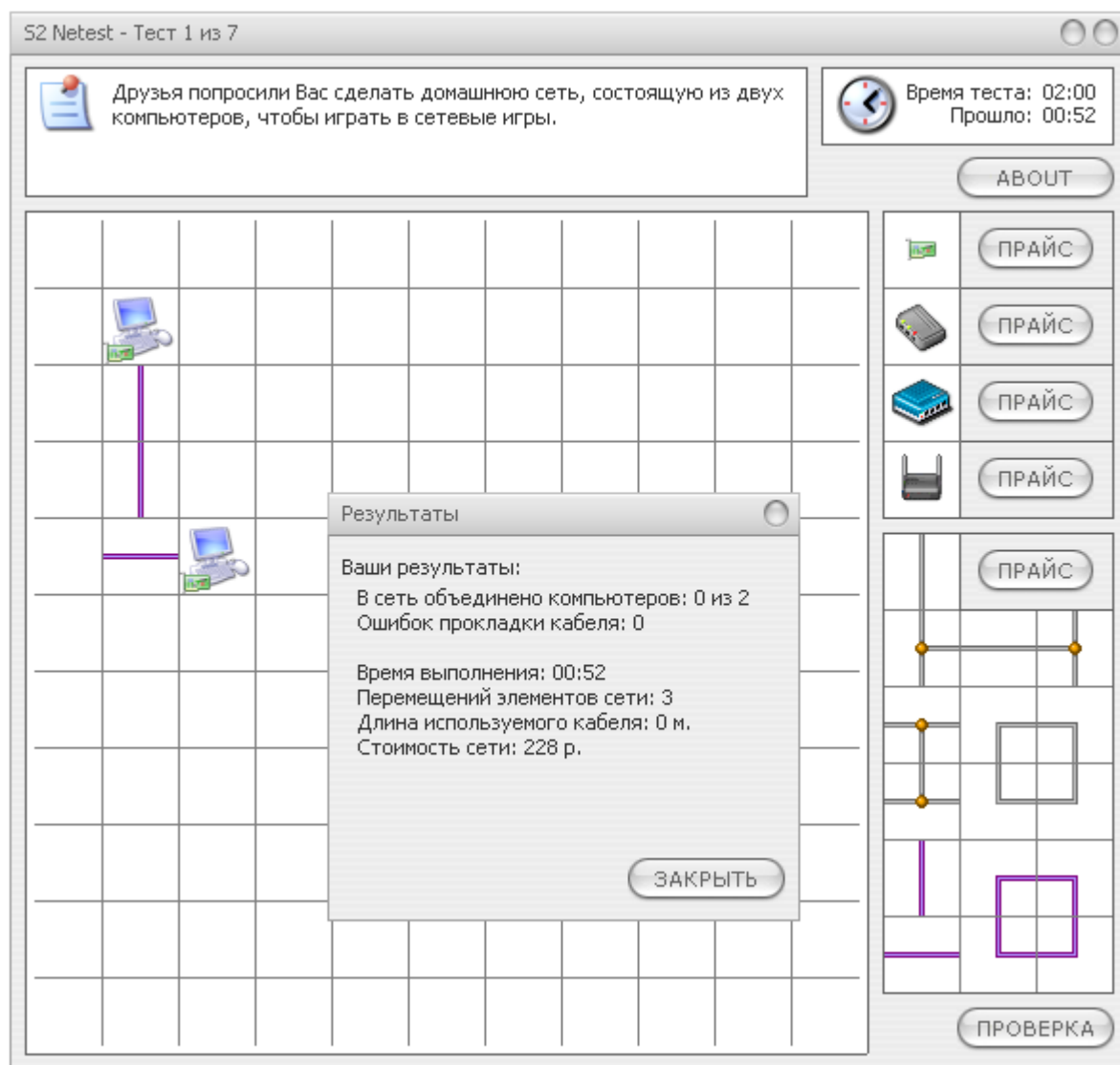


Рис. 6.13. Оптимальное решение стандартного теста (228 руб.)

Задание 1. Посмотрите на рис. 12.14 и назовите эти устройства и их характеристики.\



Рис. 6.14. Назовите эти устройства и их характеристики

Задание 2. Постройте следующую схему (рис. 12.15) и кнопкой Проверка убедитесь в том, что она работает верно.

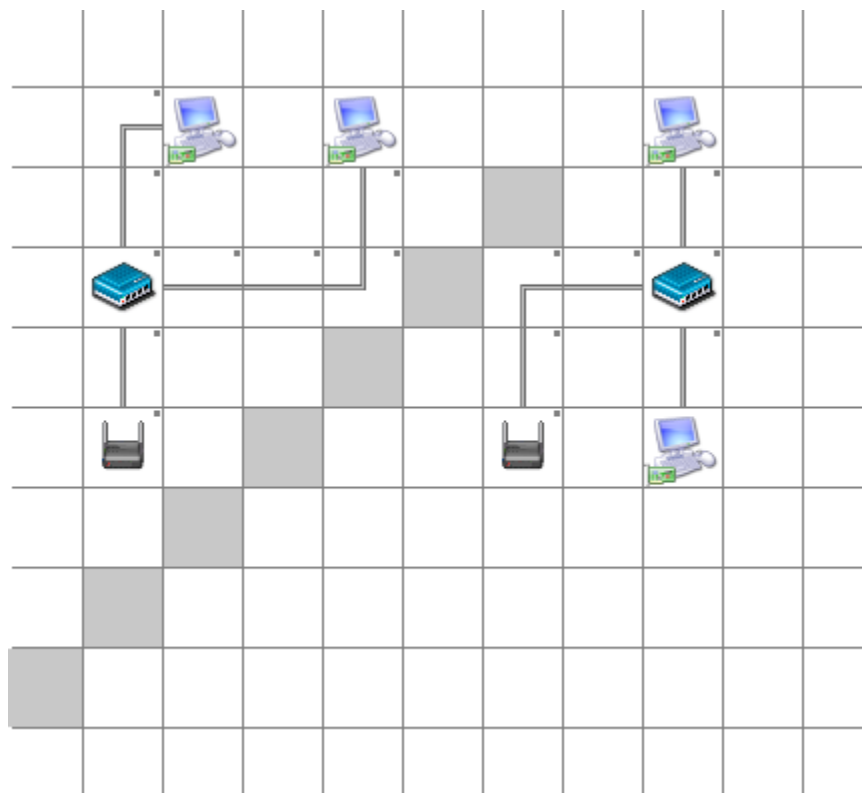


Рис. 6.15. Схемы беспроводной локальной сети

Задание 3. Проверка оптимальности построения сети

Постройте *сеть* следующего вида (рис. 6.16). Проверьте ее работоспособность.



Рис. 6.16. Сеть по топологии Звезда

Теперь *хаб* замените на свитч. *Сеть* также будет работать. Но какое из этих двух решений будет более оптимальным и почему?

Краткие итоги

В работе была эмулирована одноранговая *локальная сеть*, применены инструменты для ее создания в программе моделирования сетей S2 Netest, показаны примеры оптимальных и не удачных вариантов проектирования локальных сетей.

Лабораторная работа №5 Настройка связи между ПК в виртуальной сети

Настраиваем виртуальный ПК для работы в сети

Запускаем обе, ранее созданные нами виртуальные машины командой **VM -Питание Power On**. Для работы в сети настроим сначала первую машину. Для этого в **Панели управления** найдем **Сетевые подключения-Подключение по локальной сети-Свойства**, затем находим свойства **Протокола Интернет (TCP/IP)** и пишем **IP-адрес** и **Маску подсети** как на рис. 4.1.

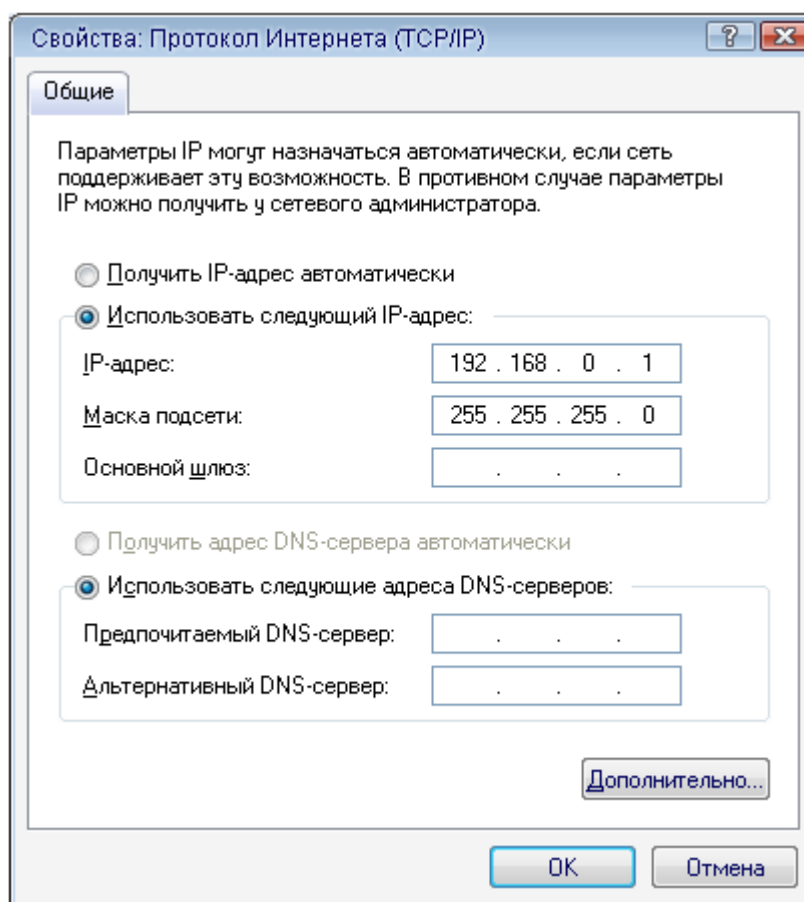
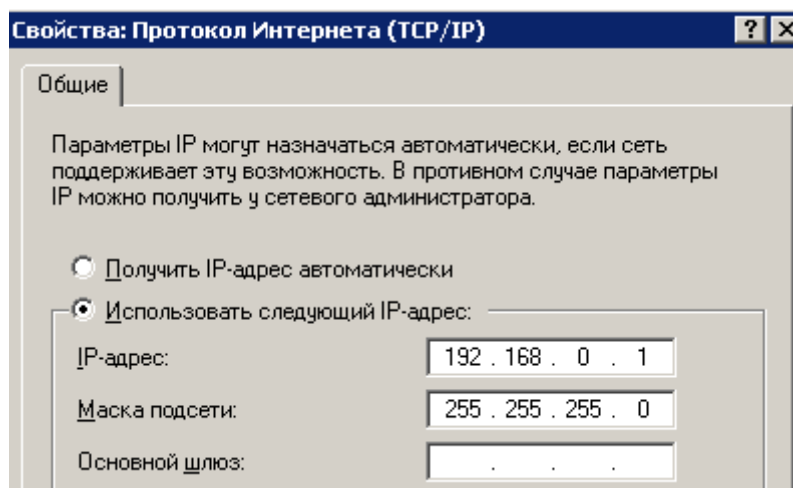
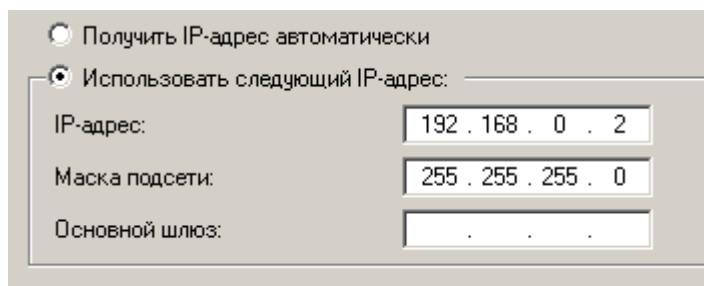


Рис. 4.1. Настраиваем VM-1

Совет

Вам придется периодически переходить от окна физического ПК к окну виртуального ПК. Для этого нажимайте на *сочетания* клавиш **Ctrl+Alt**. Аналогично включим и настроим вторую машину (рис. 4.2).



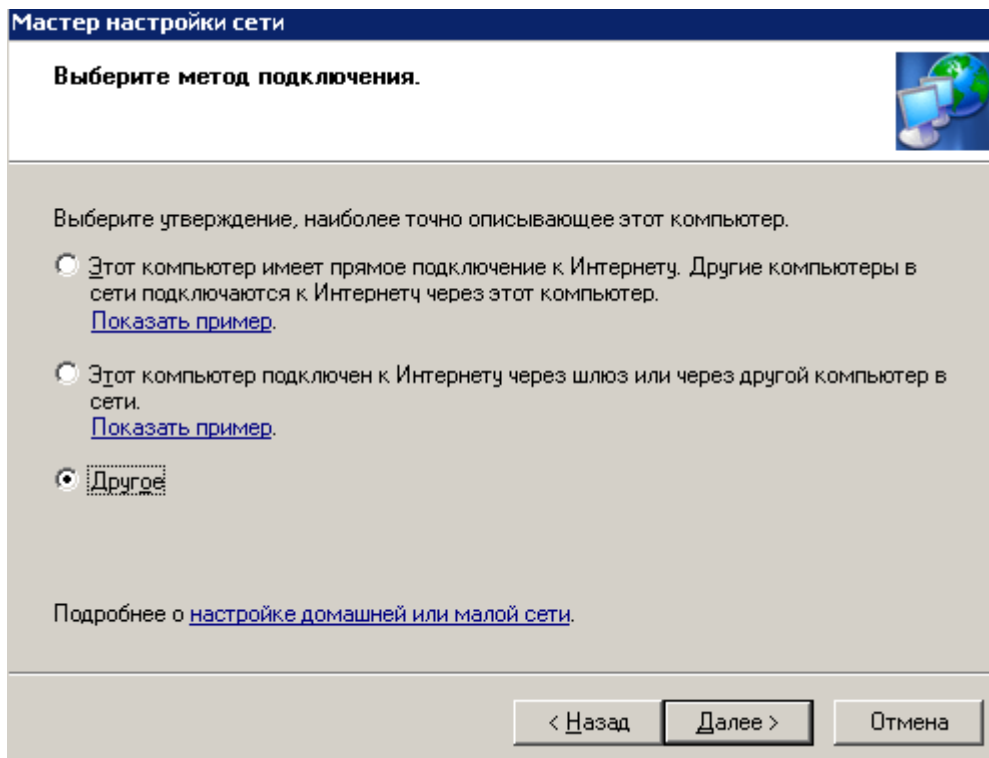
Получить IP-адрес автоматически
 Использовать следующий IP-адрес:

IP-адрес: 192 . 168 . 0 . 2
Маска подсети: 255 . 255 . 255 . 0
Основной шлюз: . . .

Рис. 4.2. Настраиваем VM-2

Настраиваем виртуальную сеть

Для настройки сети выполним команду **Сетевое окружение-Установить домашнюю или малую сеть** (рис. 4.3 и 4).



Мастер настройки сети

Выберите метод подключения.

Выберите утверждение, наиболее точно описывающее этот компьютер.

Этот компьютер имеет прямое подключение к Интернету. Другие компьютеры в сети подключаются к Интернету через этот компьютер.
[Показать пример.](#)

Этот компьютер подключен к Интернету через шлюз или через другой компьютер в сети.
[Показать пример.](#)

Другое

Подробнее о [настройке домашней или малой сети.](#)

< Назад Далее > Отмена

Рис. 4.3. Выбираем переключатель Другое

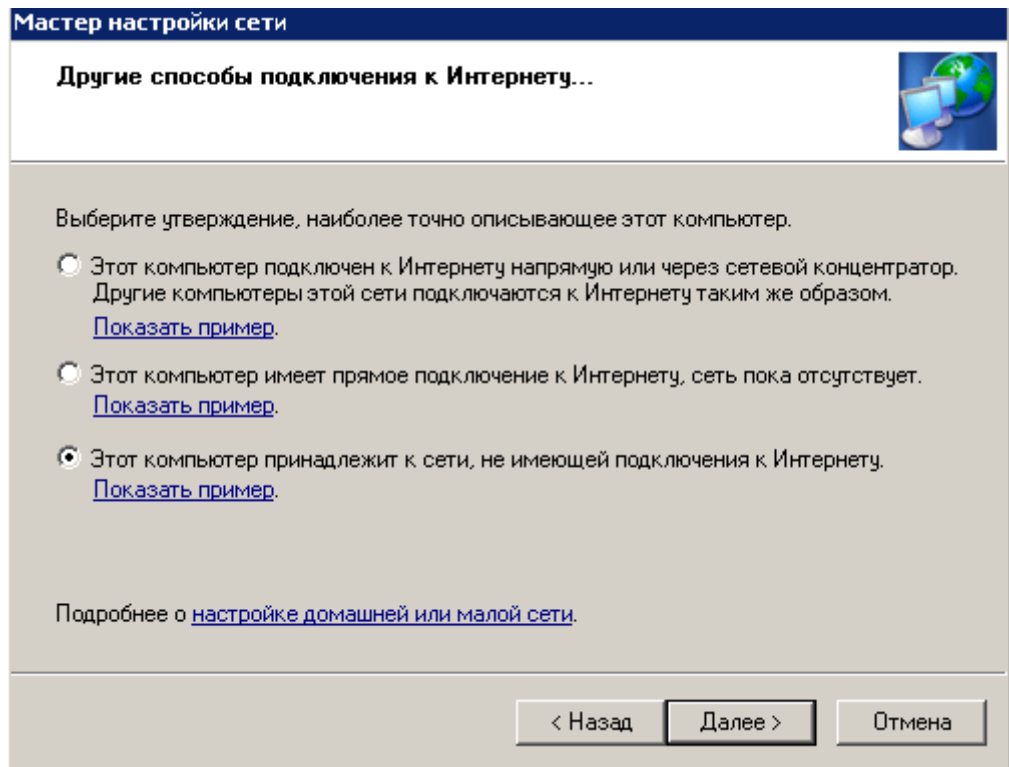


Рис. 4.4. Установим третий переключатель сверху
Присвоим машине сетевое имя (рис. 4.5).

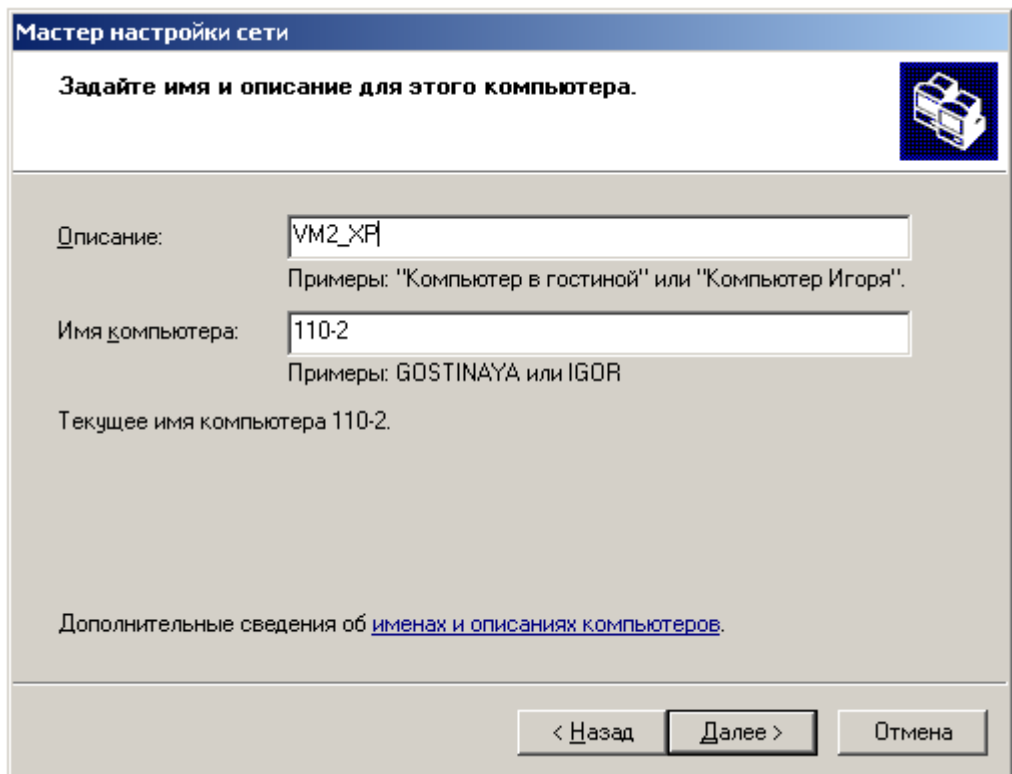


Рис. 4.5. Даем машине имя и описание
На следующем шаге (нажав **Далее**) создадим рабочую группу 110 (рис. 4.6).

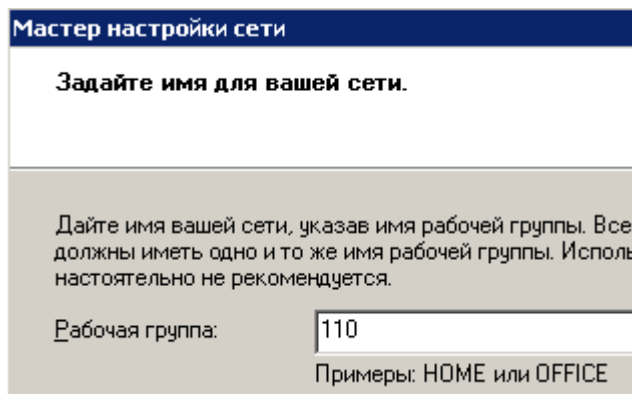


Рис. 4.6. Задаем имя для локальной сети
Снова **Далее** и включим общий *доступ* (рис. 4.7).

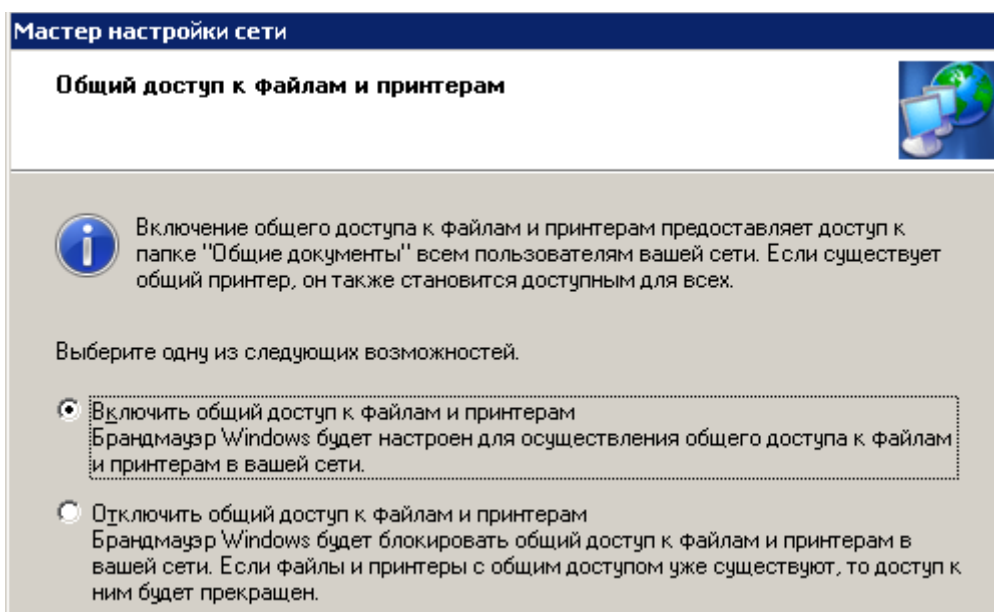


Рис. 4.7. Установим верхний переключатель

Работу мастера завершаем (рис. 4.8).

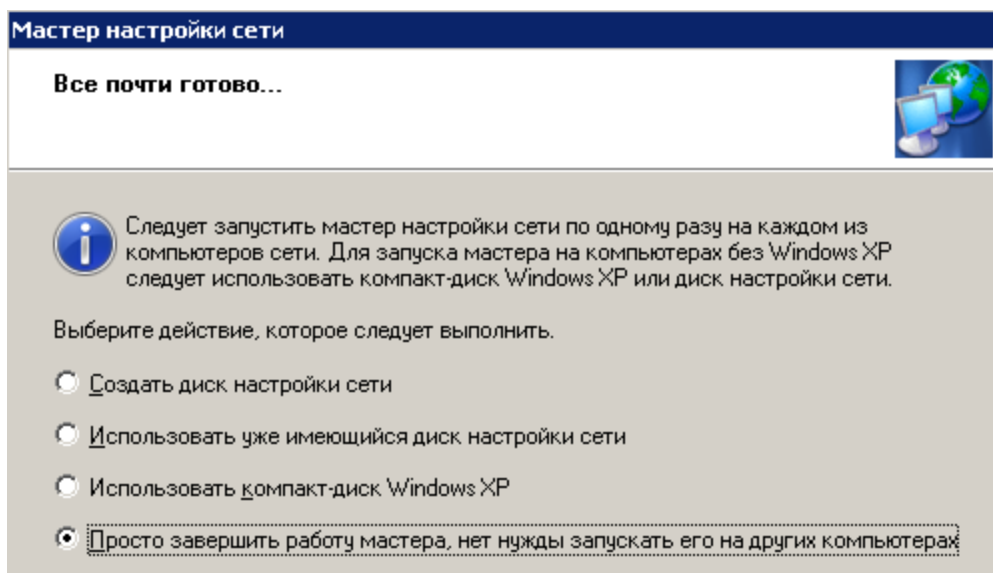


Рис. 4.8. Устанавливаем первый переключатель снизу

Эта машина настроена для работы в сети, перезагружаем ее и аналогично настроим другой виртуальный ПК, также включив его в рабочую группу 110. Перезагружаем. Сетевая настройка обеих машин завершена.

Проверяем работу виртуальных машин в сети

Попробуем зайти с первой машины на вторую и наоборот. Для этого войдем в **Сетевое окружение** и выполним команду **Отобразить компьютеры рабочей группы**. Если все нормально, то в рабочей группе 110 мы увидим машину 1 и машину 2 (рис. 4.9).

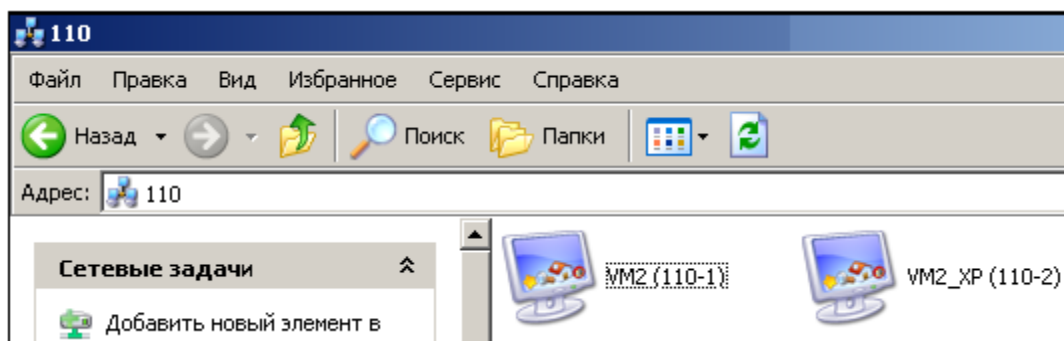


Рис. 4.9. Локальная виртуальная сеть настроена

Далее мы можем работать с такой сетью, как с обычной. Например, создать папки с общим доступом. Однако, может выйти и такое сообщение (рис. 4.10).

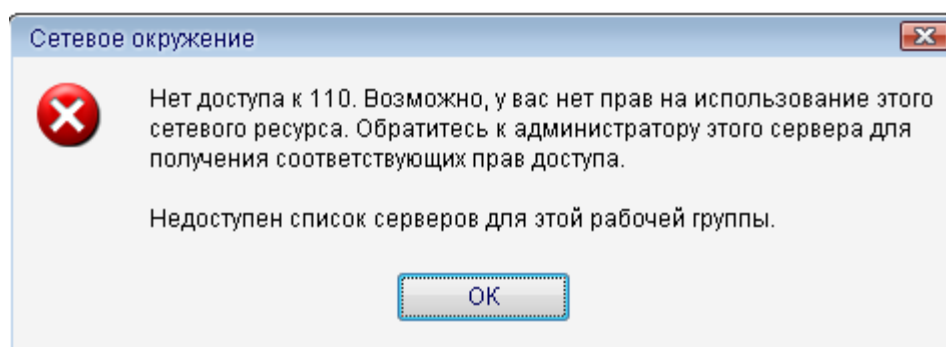


Рис. 4.10. Нет доступа к рабочей группе 110

Примечание

Одной из причин конфликтов в локальной сети (отсутствие общего доступа между ресурсами) может быть установка разного времени на рабочих станциях, т.е. для обеих машин таймер времени должен быть синхронным. Еще причина – использование нелегальной операционной системы. Другие причины мы изучим позднее.

Установка средств VMware

Чтобы получить *доступ* из виртуальной машины к файлам на физическом ПК потребуется команда **ВМ-Установка средств VMware** (рис. 4.11).

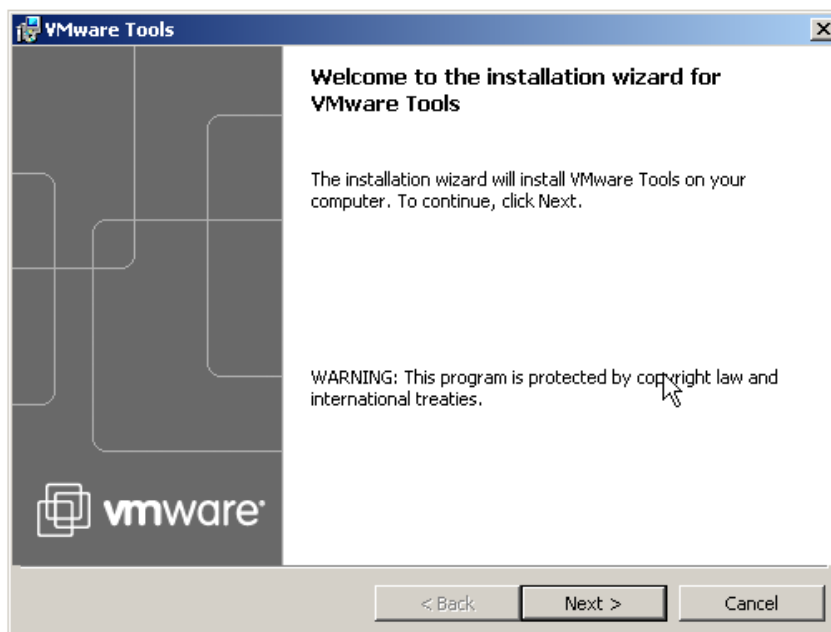


Рис. 4.11. Окно начала установки средств VMware

После инсталляции средств и перезагрузки виртуальной машины на рабочем столе *Windows XP* появится соответствующий значок. Далее выполним команду **ВМ-Настройки** и откроем вкладку **Опции (Options)** и встанем курсором на строчку папок с общим доступом (рис. 4.12).

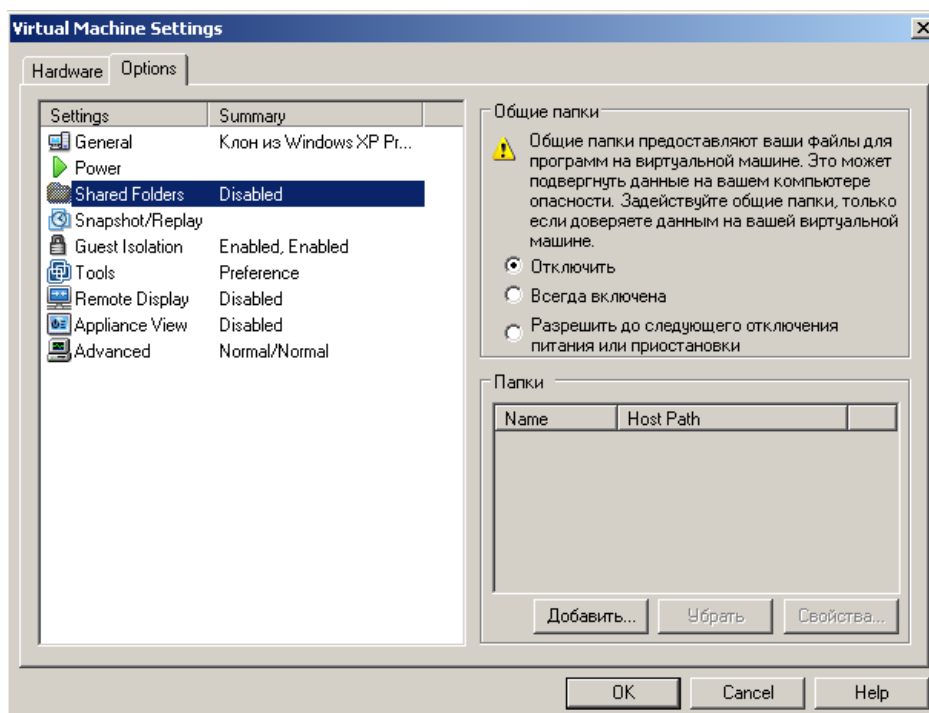


Рис. 4.12. Папки с общим доступом пока недоступны и пусты

Нажимаем на кнопку **Добавить**, дадим этой папке имя и укажем *диск* для нее (рис. 4.13).

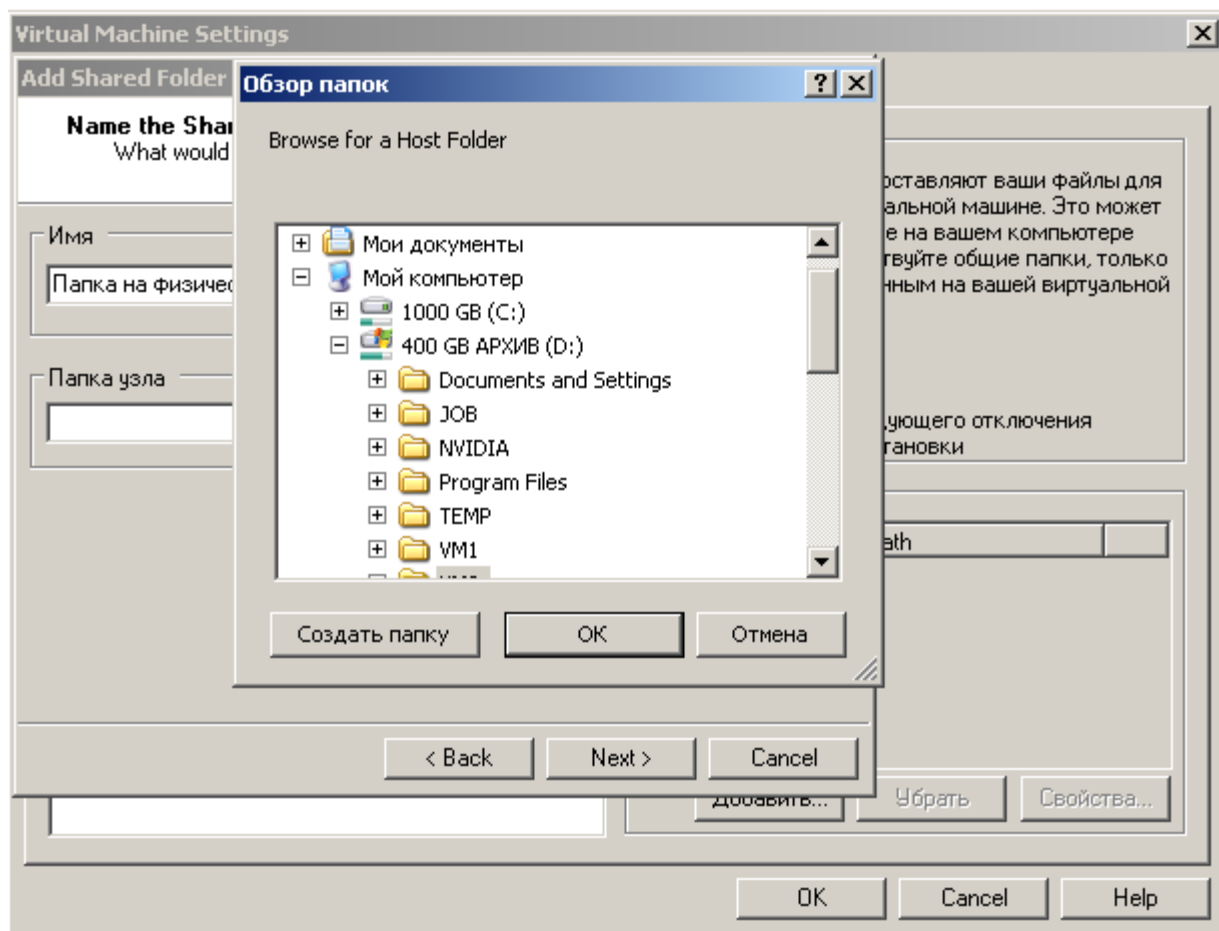


Рис. 4.13. Задаем параметры для папки с общим доступом

Далее активируем атрибуты папки (рис. 4.14).

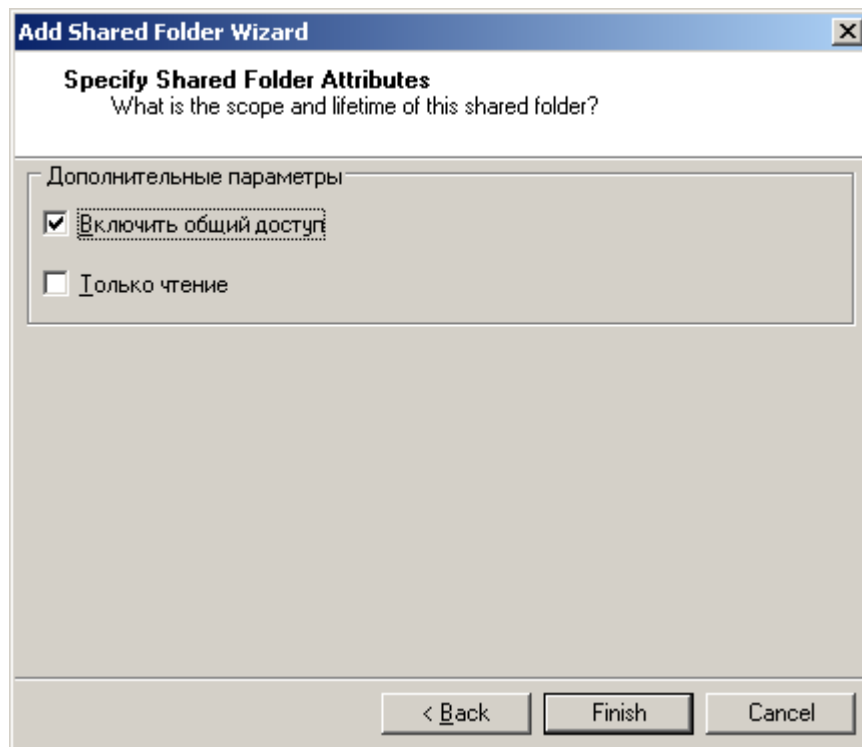


Рис. 4.14. В этом окне нам нужны оба флажка
 В следующем окне устанавливаем переключатель **Всегда включена** (рис. 4.15).

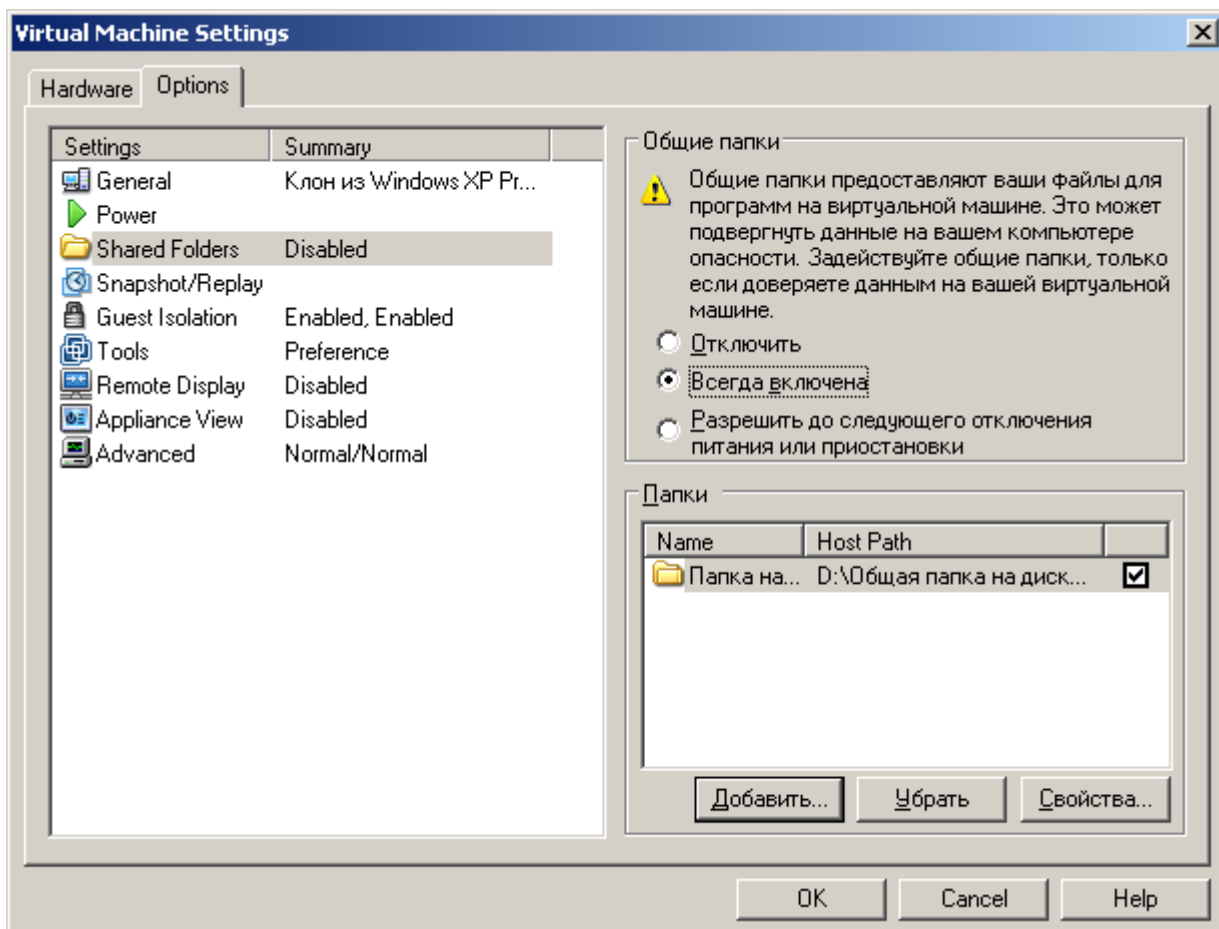


Рис. 4.15. Активируем этот переключатель

Теперь при просмотре всей сети мы увидим папку на нашем физическом ПК, т.е. через значок **VMware Shared Folders** у нас есть *связь* физического ПК с виртуальными машинами (рис. 4.16).

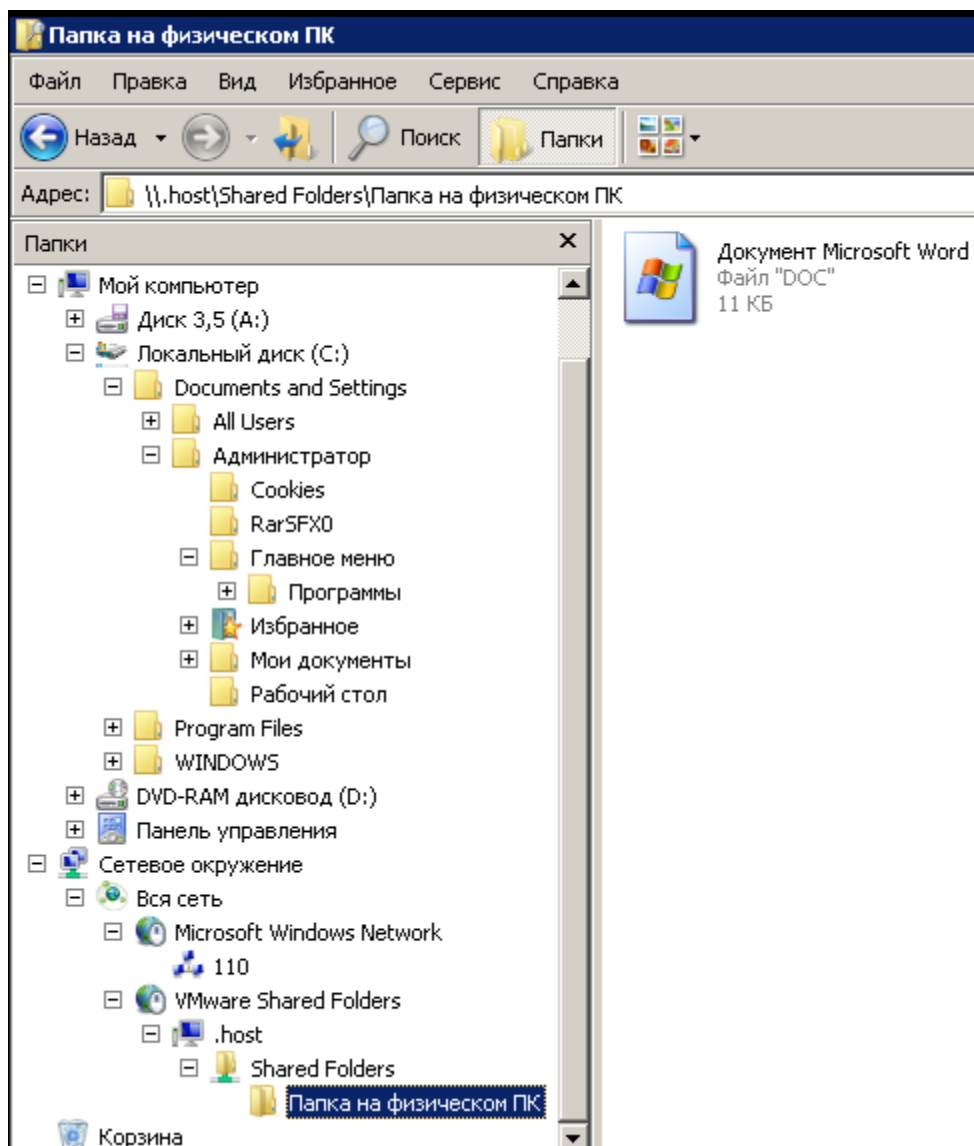
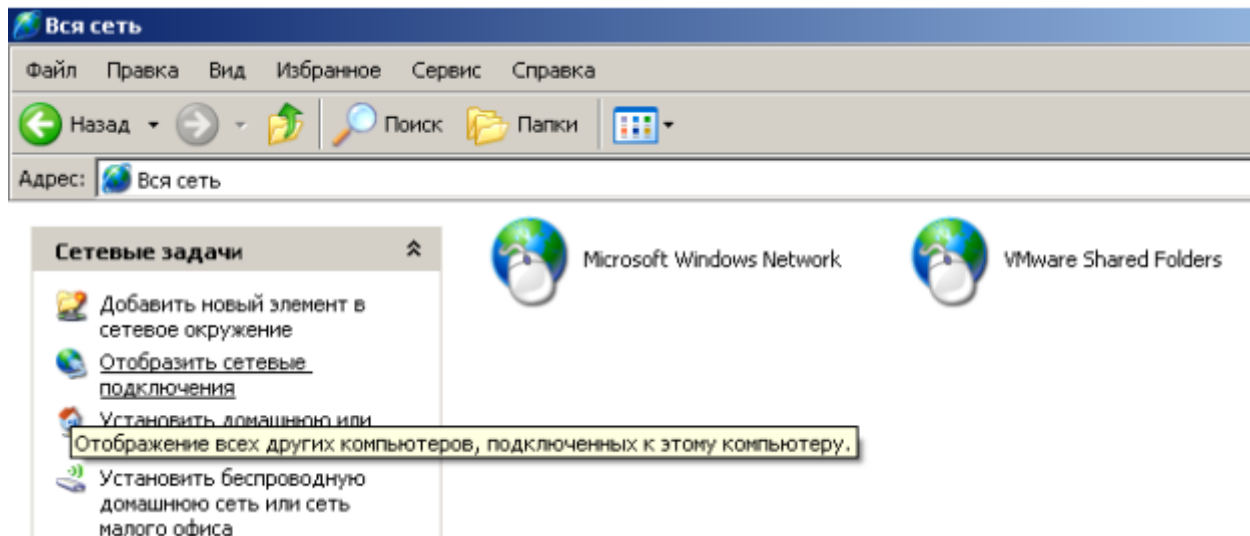


Рис. 4.16. Связь физической машины с виртуальной установлена
Теперь на виртуальную машину мы можем устанавливать любой *soft*.

Примечание

Обратите внимание на то, что установка средств Wmware решает сразу и другие проблемы, например, настройку драйверов устройств на виртуальном ПК.

Совет

Если общая папка на физическом ПК не видна, то в Сетевом окружении вы ее можете добавить, используя команду **Добавить новый элемент в сетевом окружении** (рис. 4.17).

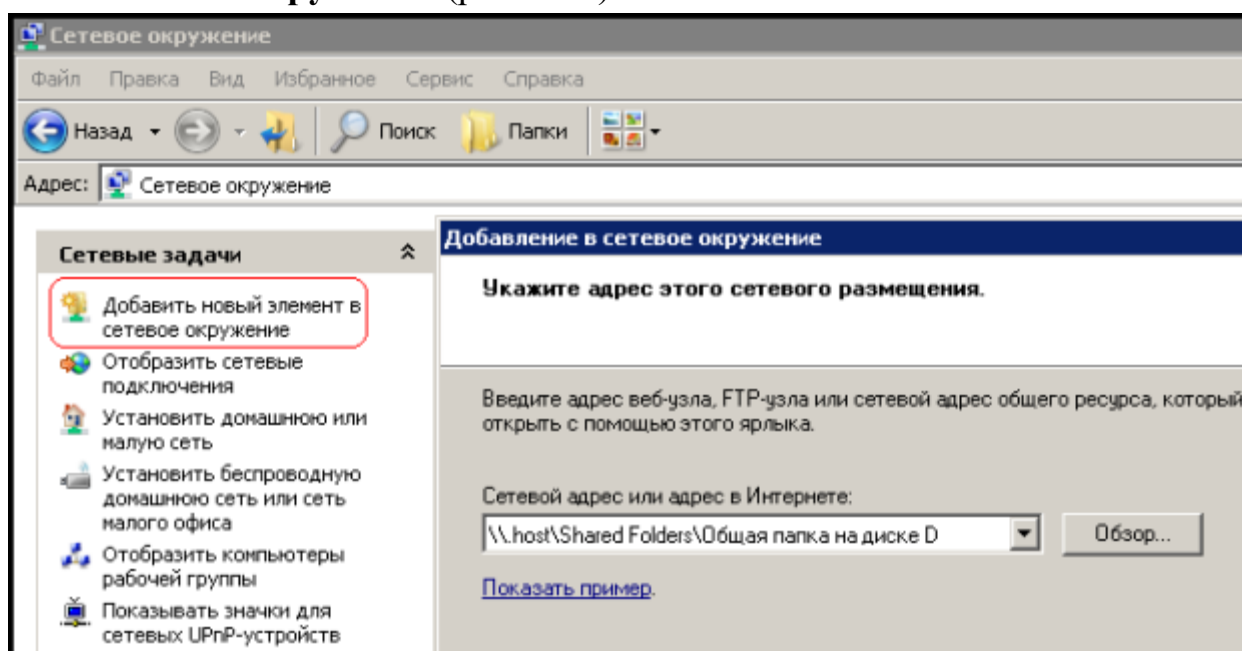


Рис. 4.17. Красным отмечена команда **Добавить новый элемент в сетевом окружении**

Краткие итоги

В этой работе мы научились настраивать связи между ПК в виртуальной сети, а также производить установку на VM средств Wmware. По лабораторной работе имеется скринкаст.

Порядок выполнения:

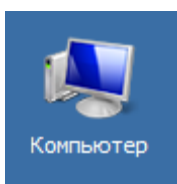
После настройки сделать 2 скриншота:

- 1 скриншот - сетевого окружения(должно быть 2 и более машин)
- 2 скриншот – наличие общей (расшаренной) папки

Лабораторная работа №6 Диагностика IP протокола

Применение команды Ping для проверки наличия связи компьютеров в сети

Наиболее быстрым способом проверки работоспособности локальной можно назвать системную команду *PING*, которая посылает сетевой *запрос* на заданный *IP-адрес* компьютера, получает ответ и выводит отчет на экран. Если посланный *запрос* получен обратно - *связь* физически существует, то ваша *сеть* настроена и работает корректно. Если же на экране вы увидите надпись "Превышен *интервал* ожидания *запрос*" - вы допустили ошибку либо в настройках, либо в подключении компьютеров. Перед запуском команды *Ping* необходимо посмотреть доступные компьютеры в



сети. Заходим в **Компьютер** и видим, что в нашей рабочей группе 110 имеется четыре ПК (рис. 8.1).

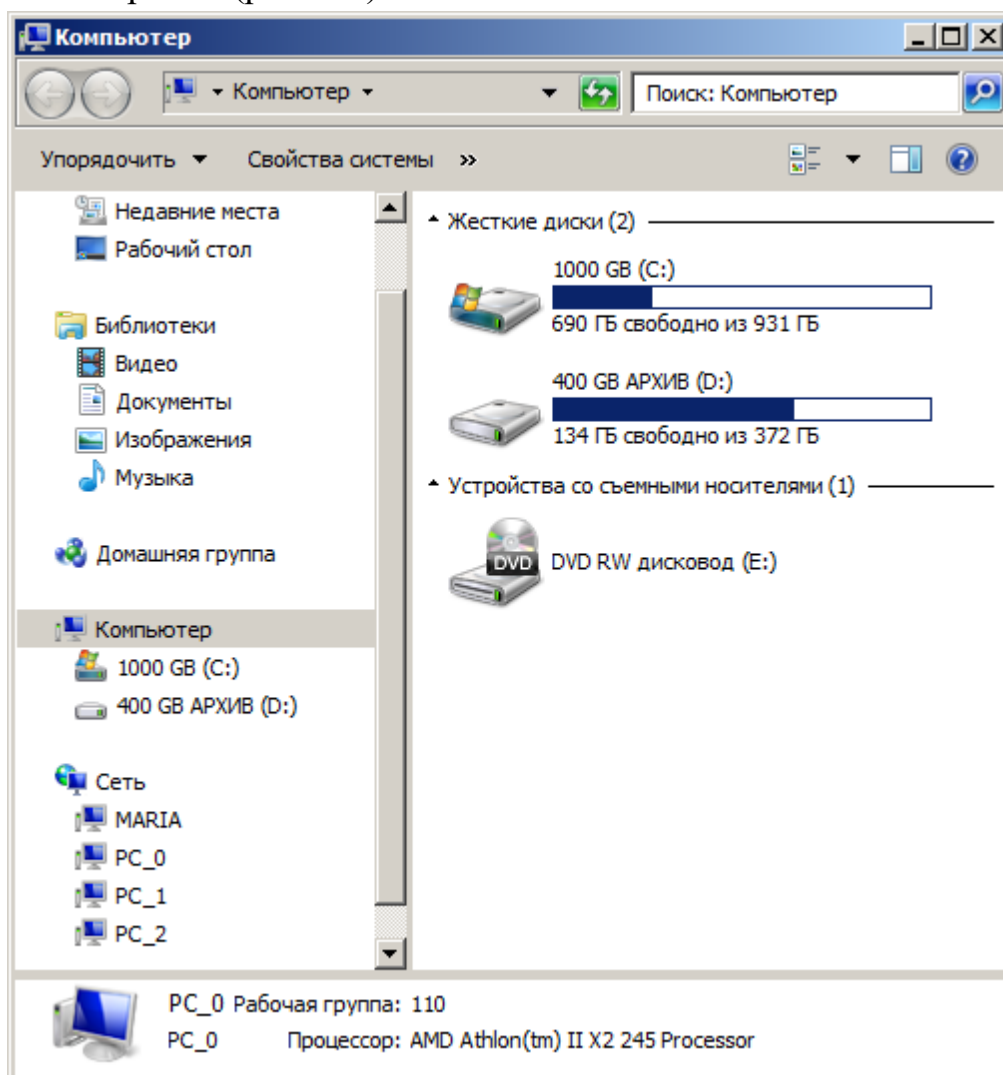
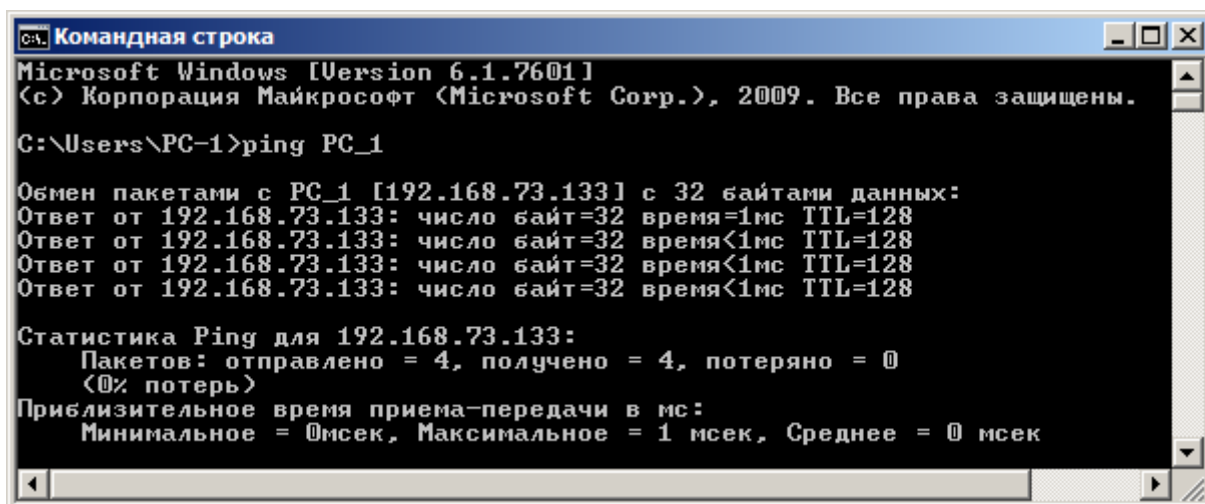


Рис. 8.1. В рабочей группе 110 мы видим 4 ПК

Для того чтобы воспользоваться командой *ping*, откройте окно командной строки командой **Пуск-Все программы-Стандартные-Командная строка** и введите там команду *ping*, укажите имя или *IP-адрес* удаленного компьютера (или его ИМЯ"/>) (рис. 8.2). По умолчанию утилита *ping* отправляет 4 пакета и ожидает каждый ответ в течение четырех секунд. По умолчанию команда посылает пакет 32 байта. За размером тестового пакета отображается время отклика удаленной системы (в нашем случае — меньше 1 миллисекунды"/>).



```
С:\>Командная строка
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\PC-1>ping PC_1

Обмен пакетами с PC_1 [192.168.73.133] с 32 байтами данных:
Ответ от 192.168.73.133: число байт=32 время=1мс TTL=128
Ответ от 192.168.73.133: число байт=32 время<1мс TTL=128
Ответ от 192.168.73.133: число байт=32 время<1мс TTL=128
Ответ от 192.168.73.133: число байт=32 время<1мс TTL=128

Статистика Ping для 192.168.73.133:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (<0% потерь)
    Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 1 мсек, Среднее = 0 мсек
```

Рис. 8.2. Пингование машины PC_1 с IP-адресом 192.168.73.133

При необходимости для этой команды вы можете использовать следующие параметры:

- t. Данный *параметр* указывает на то, что производится проверка связи с указанным узлом до прекращения вручную;
- n. Текущий *параметр* определяет количество отправляемых Echo-запросов;
- f. Этот *параметр* устанавливает бит "не фрагментировать" на *ping*-пакете. По умолчанию фрагментация разрешается;
- w. Данный *параметр* позволяет настроить тайм-аут для каждого пакета в миллисекундах (по умолчанию установлено значение 4000"/>);
- a. Текущий *параметр* определяет имена узлов по адресам;
- l. При помощи этого параметра вы можете указать размер буфера отправки;
- i. Использование данного параметра позволяет вам задать срок жизни пакета;
- v. Этот *параметр* задает тип службы для IPv4 и не влияет на поле *TOS* в *IP*-заголовке;

-r. Текущий *параметр* записывает *маршрут* для указанного числа прыжков;

-s. Данный *параметр* позволяет отмечать время для указанного числа прыжков;

-j. Используя этот *параметр*, вы можете указать свободный выбор маршрута *по* списку узлов;

-k. При помощи данного параметра вы можете определить жесткий выбор маршрута *по* списку узлов;

-R. Текущий *параметр* позволяет использовать заголовок для проверки также и обратного маршрута только для IPv6;

-S. Данный *параметр* указывает используемый *адрес* источника;

-4. *Параметр* определяет принудительное использование протокола *IP* версии 4;

-6. *Параметр* определяет принудительное использование протокола *IP* версии 6.

Итак, выше было показано, что *утилита Ping* используется в том случае, когда необходимо проверить, может ли *компьютер* подключиться к сети *TCP/IP* или сетевым ресурсам. Иначе говоря, мы пингуем для того, чтобы проверить, что отправляемые пакеты доходят до получателя. ПК-отправитель отправляет *Echo-запрос*, а ПК-получатель, в ответ должен отправить *ICMP-сообщение* с ответом. Если удаленный *компьютер* реагирует на *запрос ping*, то подключение к удаленному компьютеру работает. Также, *утилита ping* ведет статистику, из которой понятно, сколько пакетов получено, а сколько потеряно. Но, это еще не все.

Применение команды Ping для анализа качества связи ПК в сети

Для тестирования качества связи запустите *Ping* со следующими параметрами: **ping.exe -l 16384 -w 500 -n 100 192.168.73.133**. Это обеспечит отправку 100 запросов (n) пакетами *по 16 килобайт* (l) на заданный *IP адрес* с интервалом ожидания ответа в 0,5 секунды (w). То есть:

L – размер буфера отправки.

N – число отправляемых запросов,

W – *время ожидания* ответа на *запрос* в миллисекундах,

Подождите, пока пройдут все 100 пакетов. Ответ должен будет быть приблизительно такой (рис. 8.3).

```
Командная строка
Ответ от 192.168.73.133: число байт=16384 время=1мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=3мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=5мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=1мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128

Статистика Ping для 192.168.73.133:
  Пакетов: отправлено = 100, получено = 100, потеряно = 0
  (0% потерь)
Приблизительное время приема-передачи в мс:
  Минимальное = 0мсек, Максимальное = 17 мсек, Среднее = 2 мсек
```

Рис. 8.3. Ответ на команду ping.exe с ключами

Проанализируем результат выполнения команды:

- 0% потерь – сеть работает отлично.
- Если потери информации составили не более 3%, то сеть работает хорошо.

• При потерях 3-10% дошли не все пакеты, но сеть, благодаря алгоритмам коррекции ошибок, работает удовлетворительно. Из-за необходимости повторной доставки потерянной информации снижается эффективная скорости работы сети – сеть тормозит.

• Если число потерянных пакетов превышает 10-15%, то необходимо принять меры по устранению неисправности. Качество связи ПК неудовлетворительное.

Далее: как видим, время отклика удаленной системы среднее 2 мсек, а максимальное 17 мсек. Анализируя отклик *по* миллисекундам, надо иметь ввиду следующее. *По* стандарту, нормальное время отклика 16-килобайтного пакета для 100-мегабитной сети - 3-8 мс. Для 10-мегабитной - 30-80 мс. Получается, что у нас *сеть* работает на скорости порядка 100 мбит/сек.

Использование утилиты PathPing

Pathping это *утилита*, которая позволяет обнаружить потери пакетов на маршруте между вашим компьютером и заданным адресом *IP*. Потери пакетов могут сильно повлиять на работу сети, например, когда вы играете в видеоигру. Иначе говоря, *утилита* PathPing отправляет многочисленные сообщения с Echo-запросом каждому маршрутизатору, который находится между исходным пунктом и пунктом назначения, после чего, на основании пакетов, полученных от каждого из них, вычисляет процентное соотношение пакетов, возвращаемых в каждом прыжке. Поскольку *утилита* PathPing показывает степень потери пакетов на каждом маршрутизаторе или узле, то с

ее помощью вы можете точно определить маршрутизаторы и узлы, на которых возникают *сетевые проблемы*. Пример использования данной команды приведен на рис. 4.

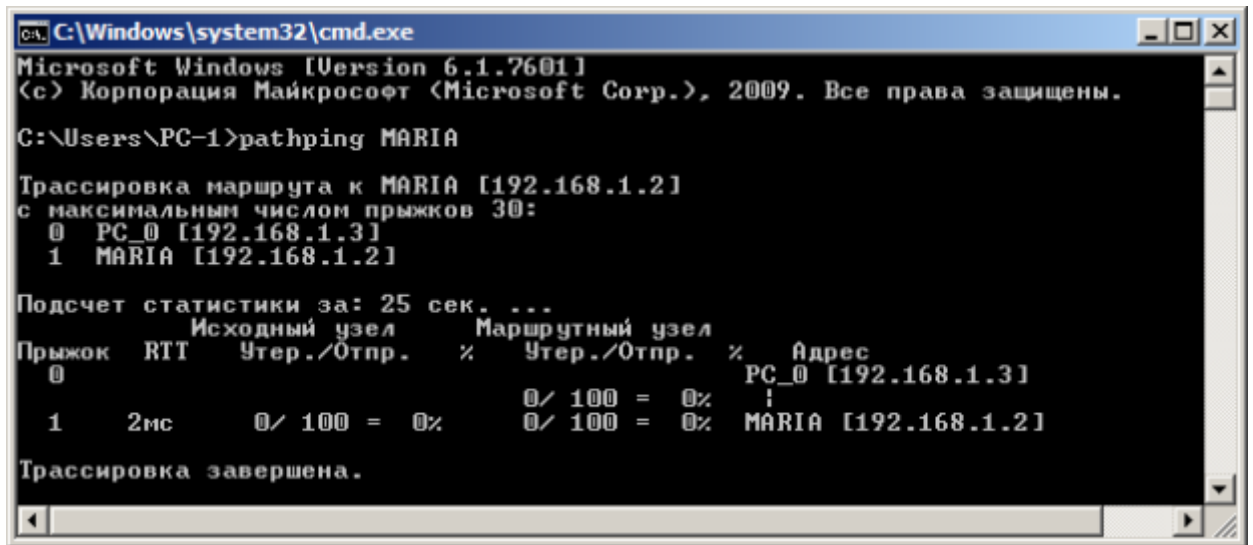


Рис. 8.4. Поиск потерь пакетов на маршруте от ПК PC_0 до ПК MARIA. Итак, в строке поиска наберем **CMD**, чтобы вызвать командную строку (рис. 8.5).

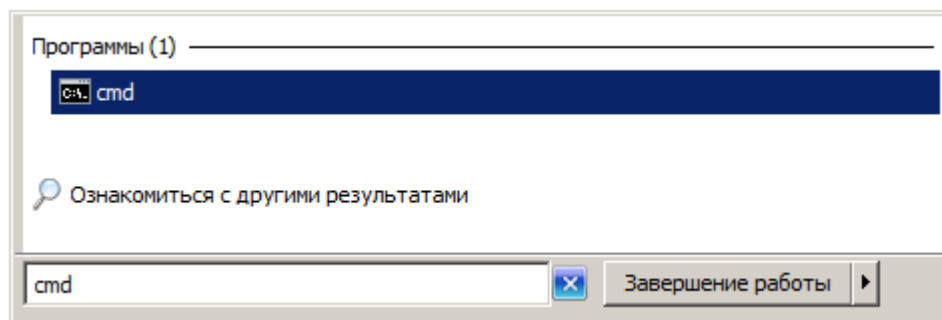


Рис. 8.5. Один из способов вызова командной строки в ОС Windows 7. Далее произведет трассировку маршрута от нашего ПК до поискового сервера Яндекс (рис. 8.6).

```

C:\Windows\system32\cmd.exe
C:\Users\PC-1>pathping yandex.ru

Трассировка маршрута к yandex.ru [213.180.204.11]
с максимальным числом прыжков 30:
 0 PC_0 [192.168.1.3]
 1 192.168.1.1
 2 lo0-at66-2.natm.ru [213.148.173.214]
 3 at66-ats66-L3-giga-core.natm.ru [213.148.163.81]
 4 ATS3-TGE1-8-TTS-TGE1-4.natm.ru [78.81.0.37]
 5 GWay-TGE0-2.natm.ru [78.81.0.254]
 6 ge-0-1-0-v1988-10g.M320-1-NOUG.nwtelecom.ru [212.48.214.53]
 7 ae1-30g.MX960-1-MMI.nwtelecom.ru [212.48.198.246]
 8 as13238-yandex.gateway.nwtelecom.ru [212.48.214.102]
 9 * * * * *
Подсчет статистики за: 200 сек. ...
Прыжок RTT Исходный узел Маршрутный узел Адрес
0 --- PC_0 [192.168.1.3]
1 1мс 0/100 = 0% 0/100 = 0% 192.168.1.1
2 1мс 0/100 = 0% 0/100 = 0% lo0-at66-2.natm.ru [213.148.1
4]
3 2мс 0/100 = 0% 0/100 = 0% at66-ats66-L3-giga-core.natm.
13.148.163.81]
4 1мс 0/100 = 0% 0/100 = 0% ATS3-TGE1-8-TTS-TGE1-4.natm.r
.81.0.37]
5 3мс 0/100 = 0% 0/100 = 0% GWay-TGE0-2.natm.ru [78.81.0.
6 2мс 0/100 = 0% 0/100 = 0% ge-0-1-0-v1988-10g.M320-1-NOU
elecom.ru [212.48.214.53]
7 11мс 0/100 = 0% 0/100 = 0% ae1-30g.MX960-1-MMI.nwtelecom
212.48.198.246]
8 --- 100/100 =100% 0/100 = 0% as13238-yandex.gateway.nwtele
u [212.48.214.102]

Трассировка завершена.

```

Рис. 8.6. Пример использования утилиты Pathping

Проанализируем результат:

- Первый блок информации представляет собой трассировку. Вы можете пропустить его и перейти ко второму блоку информации, в котором будет указано процентное отношение потерь пакетов.

- Если пакеты не терялись на данном маршруте подключения, то вы увидите 0% потерь пакетов. Если вы увидите значения, отличающиеся от 0%, это означает, что на пути к нашим серверам были потери пакетов. Потери выше 1% начиная с первого шага, могут указывать на некорректную работу узлов сети или маршрутизаторов. Если эти устройства вам доступны, то нужно попробовать обновить их программное обеспечение или полностью заменить их. Иначе, о потерях, возникших после первого шага и до последнего шага, следует сообщить вашему Интернет провайдеру.

Примечание

Если последние строки указывают на 100% потерь, то это не является показателем проблемы, а происходит потому, что сервера защищены от нежелательного трафика и атак.

С данной командой вы можете использовать следующие параметры:

-g. Данный *параметр* определяет использование параметра свободной маршрутизации в *IP*-заголовке с набором промежуточных мест назначения для сообщений с Echo-запросом, который указывается в списке компьютеров.

-h. Данный *параметр* задает максимальное количество переходов на пути при поиске конечного объекта;

-i. Этот *параметр* указывает *IP-адрес* источника;

-n. Текущий *параметр* предотвращает попытки сопоставления *IP*-адресов промежуточных маршрутизаторов с их именами, что существенно ускоряет вывод результатов;

-r. Используя данный *параметр*, вы можете задать *время ожидания* между последовательными проверками связи, где значением по умолчанию указано 250 миллисекунд;

-q. При помощи текущего параметра вы можете указать количество сообщений с Echo-запросом, отправленных каждому маршрутизатору пути (по умолчанию - 100);

-w. Данный *параметр* определяет *время ожидания* для получения Echo-ответов протокола *ICMP* или *ICMP*-сообщений об истечении времени в миллисекундах, которые соответствуют данному сообщению Echo-запроса. Значение по умолчанию 4 секунды;

-4. *Параметр* определяет принудительное использование протокола *IP* версии 4;

-6. *Параметр* определяет принудительное использование протокола *IP* версии 6.

Другие команды командной строки. Отображение параметров TCP/IP-протокола командой Ipconfig

Команда **IPCONFIG** используется для отображения текущих настроек протокола *TCP/IP* и для обновления некоторых параметров, задаваемых при автоматическом конфигурировании сетевых интерфейсов при использовании протокола *DHCP*. Предположим, что у нас имеется *сеть*, изображенная на рис. 8.7.

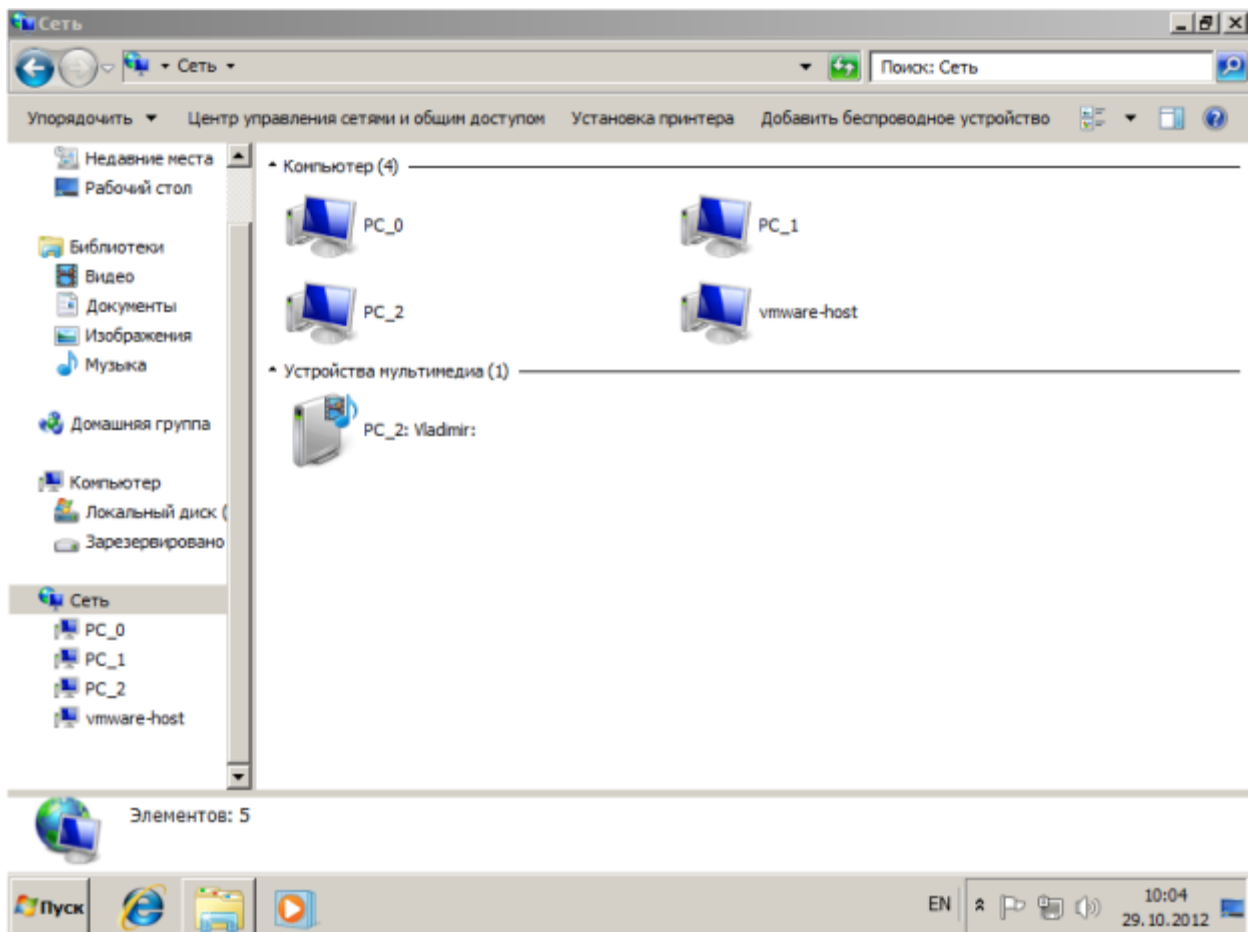


Рис. 8.7. Небольшая локальная сеть
Выполним команду командой Ipconfig на PC_2 (рис. 8.8).

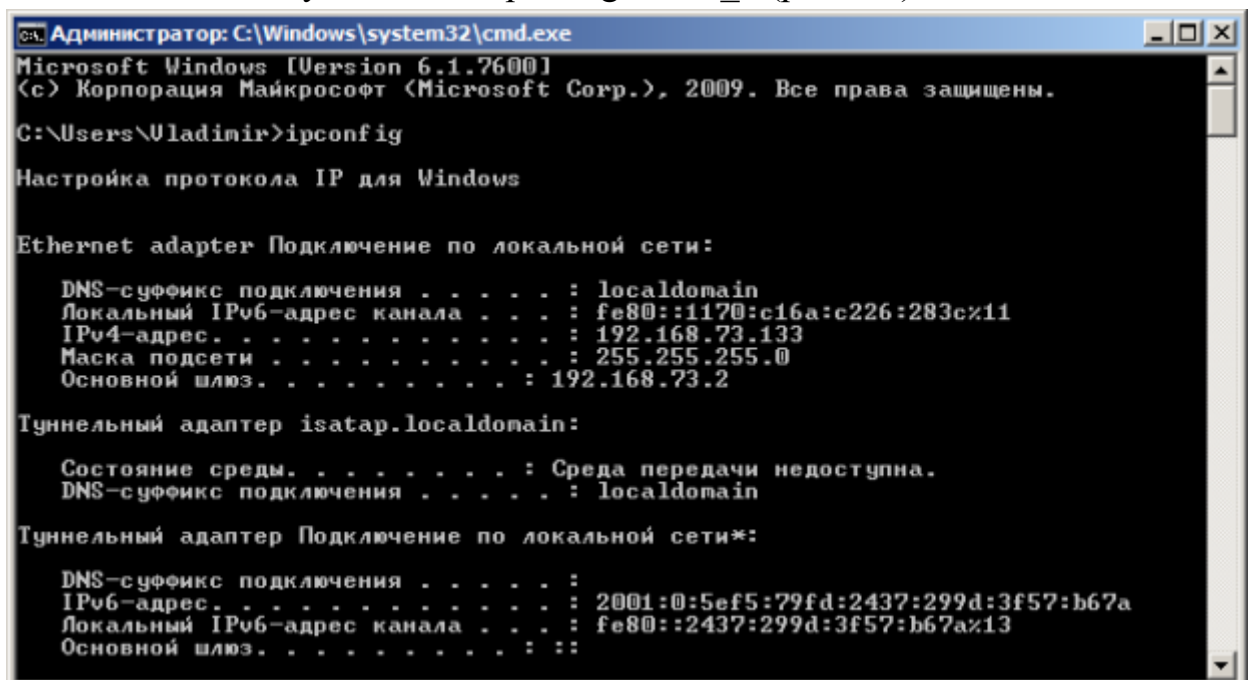


Рис. 8.8. Отображение параметров TCP/IP-протокола командой Ipconfig
Из отчета мы видим такую информацию:

- DNS-суффикс подключения - localdomain (из настроек сетевого подключения)

- Локальный IPv6-адрес канала - локальный IPv6 адрес, если используется адресация IPv6
- IPv4-адрес - используемый для данного адаптера IPv4 – адрес
- Маска подсети - 255.255.255.0
- Основной шлюз - IP-адрес маршрутизатора, используемого в качестве шлюза по умолчанию.

Примечание

Туннельный адаптер isatap.localdomain это эмуляция IPv6 в сетях IPv4. ISATAP (Intra-Site Automatic Tunnel Addressing Protocol) — Протокол автоматической внутрисайтовой адресации туннелей, позволяющий передавать между сетями IPv6 пакеты через сети IPv4

Ключи команды:

/all *Отображение* полной информации *по* всем адаптерам.

/release [адаптер] *Отправка* сообщения DHCPRELEASE серверу DHCP для освобождения текущей конфигурации DHCP и удаления конфигурации IP-адресов для всех адаптеров (если *адаптер* не задан) или для заданного адаптера. Этот *ключ* отключает протокол TCP/IP для адаптеров, настроенных для автоматического получения IP-адресов.

/renew [адаптер] Обновление IP-адреса для определённого адаптера или если *адаптер* не задан, то для всех. Доступно только при настроенном автоматическом получении IP-адресов.

/flushdns Очищение DNS кэша.

/registerdns Обновление всех зарезервированных адресов DHCP и перерегистрация имен DNS.

/displaydns *Отображение* содержимого кэша DNS.

/showclassid *адаптер* *Отображение* кода класса DHCP для указанного адаптера. Доступно только при настроенном автоматическим получением IP-адресов.

/setclassid *адаптер* [код_класса] Изменение кода класса DHCP. Доступно только при настроенном автоматическим получением IP-адресов.

/? Справка. TCP/IP: значения IP адреса, маски и шлюза.

Команда вывода списка компьютеров рабочей группы Net view

В командной строке введите команду **net view**, и вы увидите *список* компьютеров своей рабочей группы (рис. 8.9).


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

C:\Users\PC-1>net view
Имя сервера          Заметки
-----
\\MARIA                MARIA
\\PC_0                 PC_0
\\PC_1                 Uladimir
\\PC_2
Команда выполнена успешно.

C:\Users\PC-1>_
```

Рис. 8.9. В рабочей группе имеется 4 ПК

Трассировка

Tracert — это служебная компьютерная программа, предназначенная для определения маршрутов следования данных в сетях *TCP/IP*. Программа *tracert* выполняет отправку данных указанному узлу сети, при этом отображая сведения о всех промежуточных маршрутизаторах, через которые прошли данные на пути к целевому узлу. В случае проблем при доставке данных до какого-либо узла программа позволяет определить, на каком именно участке сети возникли неполадки.

Запуск программы производится из командной строки. Для этого вы должны войти в неё. Для операционной системы *Windows 7* существует несколько способов запуска командной строки:

1. Сочетание клавиш Win (кнопка с логотипом Windows) + R (должны быть нажаты одновременно) — В графе "Открыть" написать "cmd" и нажать Ок.
2. Пуск — Все программы — Стандартные — Командная строка.

В открывшемся окне мы напишем **tracert ya.ru**. Принцип действия этой программы схож с принципом действия программы *ping*. Команда отправляет на сервер данные и при этом фиксирует все промежуточные маршрутизаторы, через которые проходят эти данные на пути к серверу (целевому узлу). Если при доставке данных до одного из узлов происходит проблема, программа определяет участок сети, на котором возникли неполадки. Время отклика показывает загруженность канала. А вот если вместо времени отклика вы видите надпись "**Превышен интервал ожидания для запроса**", это значит, что на данном узле связи происходит потеря данных, а значит, проблема именно в нем – рис. 8.10.

```

C:\Windows\system32\cmd.exe
C:\Users\PC-1>tracert ya.ru

Трассировка маршрута к ya.ru [87.250.250.3]
с максимальным числом прыжков 30:

  1  <1 ms    <1 ms    <1 ms    192.168.1.1
  2   4 ms     1 ms     1 ms     lo0-at66.natm.ru [213.148.173.223]
  3   4 ms     2 ms     2 ms     at66-ats66-L3-giga-core.natm.ru [213.148.163.81]

  4   4 ms     14 ms    6 ms     AT53-TGE1-8-TTS-TGE1-4.natm.ru [78.81.0.37]
  5   3 ms     3 ms     4 ms     GWay-TGE0-2.natm.ru [78.81.0.254]
  6   4 ms     1 ms     1 ms     ge-0-1-0-v1988-10g.M320-1-NOUG.nwtelecom.ru [212
.48.214.53]
  7  10 ms     9 ms     9 ms     ae1-30g.MX960-1-MMI.nwtelecom.ru [212.48.198.246
]
  8  10 ms     9 ms     9 ms     as13238-yandex.gateway.nwtelecom.ru [212.48.214.
102]
  9   *       *       *       Превышен интервал ожидания для запроса.
 10  16 ms    16 ms    16 ms    s600-61.yandex.net [87.250.239.36]
 11  17 ms    18 ms    18 ms    13-s3600-s600.yandex.net [213.180.213.53]
 12  18 ms    17 ms    17 ms    www.yandex.ru [87.250.250.3]

Трассировка завершена.
C:\Users\PC-1>

```

Рис. 8.10. Пример трассировки домена ya.ru

Параметры команды tracert:

- d не определять доменные имена маршрутизаторов
- h <значение> установить максимальное количество переходов
- w <значение> установить максимальное время ожидания ответа (в миллисекундах)

Итак, трассировка маршрута помогает определить проблемный узел. Если данные проходят нормально и "стопорятся" на самом пункте назначения, то проблема действительно с сайтом. Если трассировка маршрута прекращается на середине пути, то проблема в одном из промежуточных маршрутизаторов. Если прохождение пакетов прекращается в пределах сети вашего провайдера — то и проблему нужно решать "на местном уровне". Попутно хочется отметить, что программа работает только в направлении от источника пакетов и является весьма грубым инструментом для выявления неполадок в сети.

Краткие итоги

В лабораторной работе мы рассмотрели применение команды *Ping* для проверки наличия связи компьютеров в сети и для анализа качества связи ПК, научились пользоваться командами PathPing, Ipconfig, Net view и Tracert. Работу дополняет скринкаст.

В готовой лабораторной работе, оформить и показать скриншоты команд: Ping, PathPing, Ipconfig, Net view и Tracert.

Лабораторная работа №7 Настройка беспроводной сети в ОС Windows 7

Пример 1. Легкая (полуавтоматическая) настройка беспроводного маршрутизатора TL-WR1043ND

Мы подключим к WI-FI маршрутизатор TP-LINK, точнее – модель TL-WR1043ND (рис.18.1-18.3). Это современное устройство, у которого максимальная скорость беспроводного соединения: 300 Мбит/сек, а скорость портов 1000 Мбит/сек.



Рис. 9.1. Wi-Fi-точка доступа (роутер) TL-WR1043ND



Рис. 9.2. Передняя панель беспроводного маршрутизатора TL-WR1043ND

Светодиодные индикаторы и кнопка-индикатор QSS (быстрая настройка параметров безопасности):

- **PWR** – питание. Индикатор выкл - питание отключено, вкл - питание включено.
- **SYS** – система. Вкл. - загрузка исходных параметров или системная ошибка. Мигает - устройство работает в нормальном режиме. Выкл. - системная ошибка.
- **WLAN** – беспроводная сеть. Выкл. - функция беспроводной передачи данных отключена. Мигает - функция беспроводной передачи данных включена.

- **WAN** (Интернет), **LAN** (Локальная сеть) 1-4. Выкл. - у порта нет подключенных устройств. Вкл. - к порту подключено устройство, но оно неактивно. Мигает - к порту подключено устройство и оно активно.

- **QSS** - быстрая настройка параметров безопасности. Медленно мигает - беспроводное устройство производит подключение к сети через функцию QSS. Этот процесс занимает примерно две минуты. Вкл. - беспроводное устройство было успешно подключено к сети посредством функции QSS. Быстро мигает - не удалось подключить беспроводное устройство к сети посредством функции QSS.



Рис. 9.3. Задняя панель беспроводного маршрутизатора TL-WR1043ND

На задней панели расположены следующие элементы:

- **POWER** - разъем для подключения питания от адаптера питания, входящего в комплект поставки беспроводного маршрутизатора TL-WR1043ND

- **RESET** – кнопка сброса конфигурации роутера для его возврата к заводским настройкам. При помощи иголки нажмите и удерживайте кнопку Reset 5 секунд, затем подождите, пока маршрутизатор выполнит перезагрузку.

- **USB** - разъем для подключения устройства хранения данных или, например, принтера.

- **WAN** синяя розетка RJ-45 для подключения DSL/кабельного модема или сети Интернет (порт для подключения Сети от провайдера).

- **Антенна Wi-Fi** черного цвета служит для беспроводного получения и передачи данных.

- **1,2,3,4 (LAN)** – розетки RJ-45 желтого цвета для подключения маршрутизатора к компьютерам локальной сети.

Итак, наш беспроводный роутер подключен к электросети, от него идет *витая пара* на стационарный ПК (патчкорд входит в комплект поставки), а Wi-Fi мы будем использовать, чтобы подключить ноутбук. Настройку роутера можно производить как на стационарном ПК (десктопе), так и со стороны ноутбука. Или там, или там нужно выполнить команду **Панель Управления – Центр управления сетями и общим доступом – Настройка нового подключения или сети-Создание и настройка новой сети** (рис. 9.4).

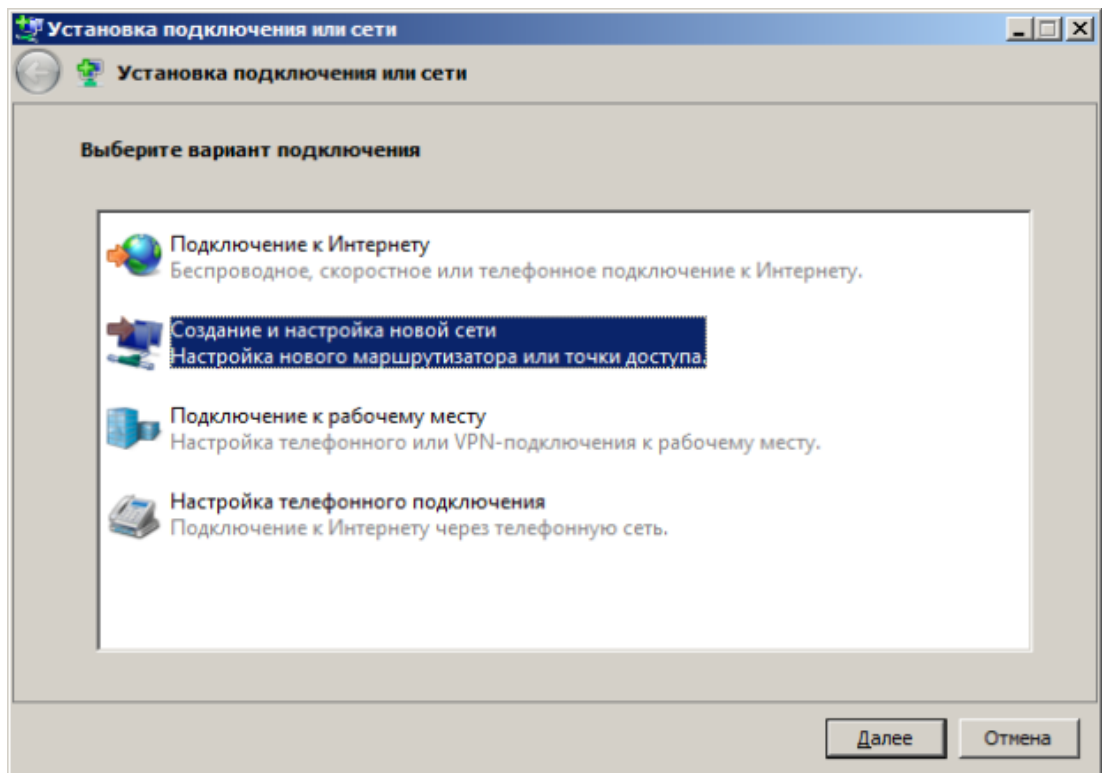


Рис. 9.4. Окно Установка подключения или сети

Нажимаем на кнопку **Далее**, видим наше беспроводное устройство (рис. 9.5).

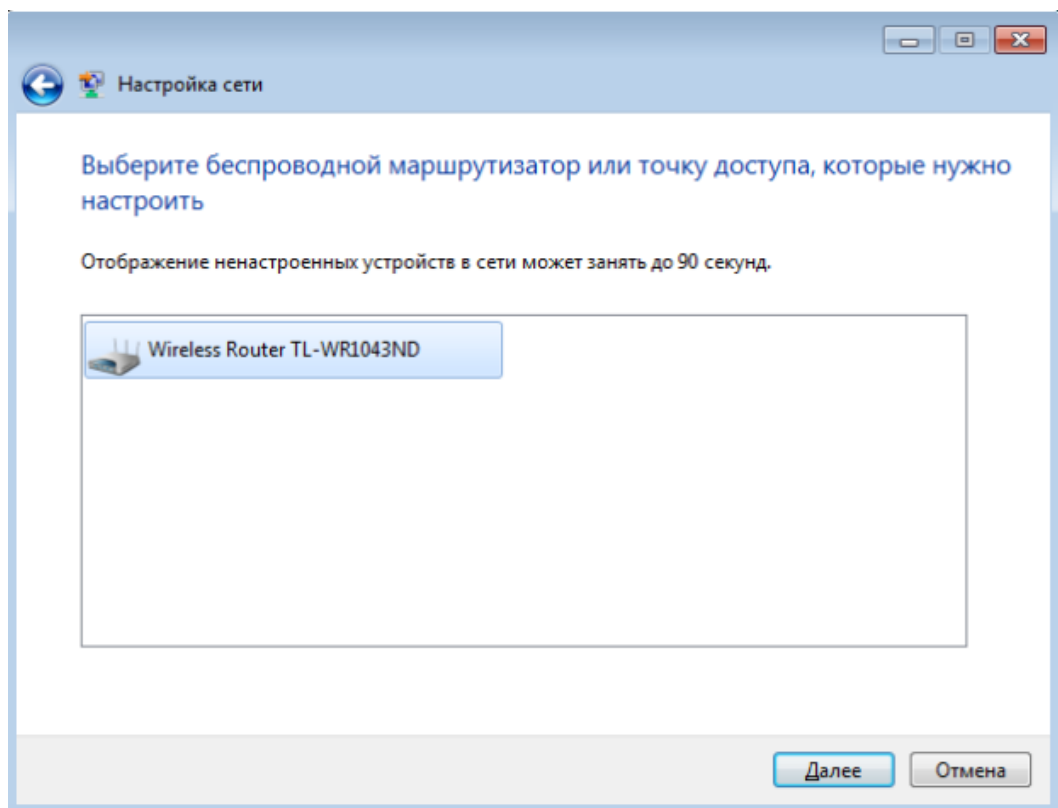


Рис. 9.5. Обнаружение точки доступа прошло нормально

Следующим этапом необходимо вести PIN-код с этикетки на маршрутизаторе (рис. 9.6 и рис. 9.7).



Рис. 9.6. На этикетке маршрутизатора читаем PIN-код

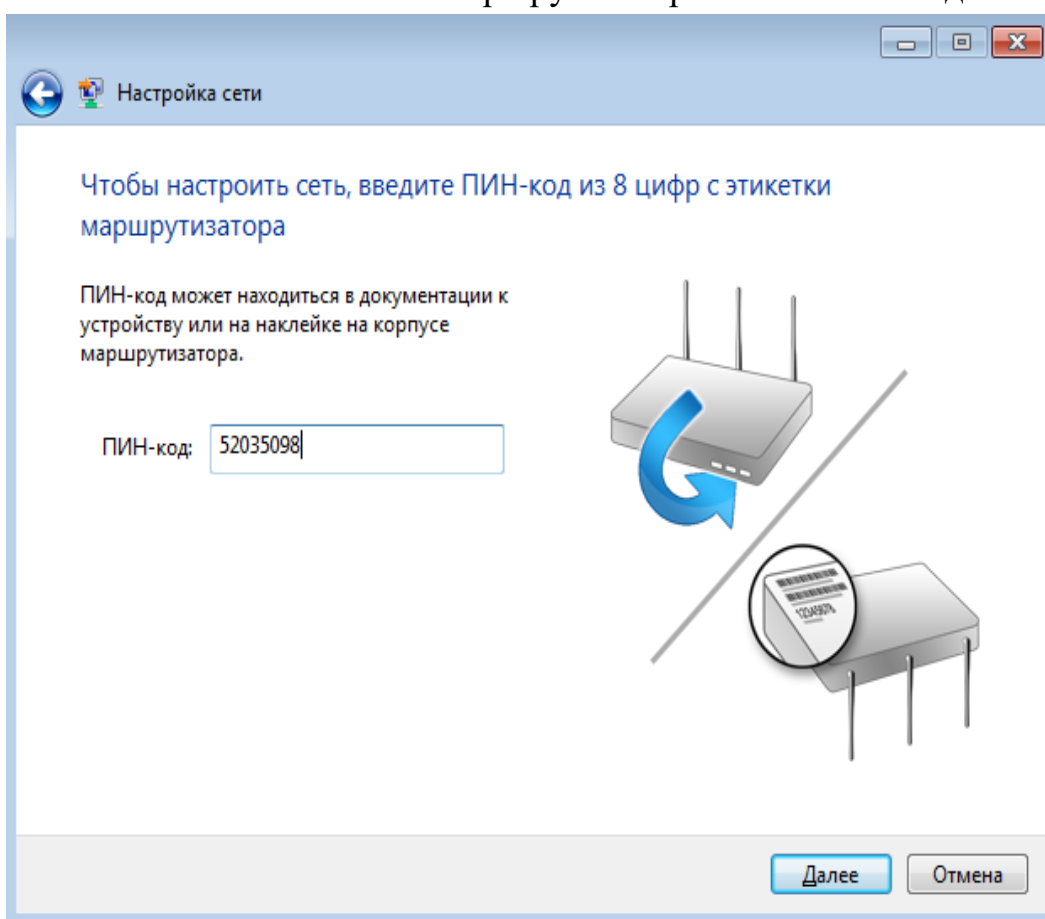


Рис. 9.7. Вводим PIN-код в окно Настройка сети

После нажатия на кнопку **Далее** следует согласиться с рекомендуемыми настройками точки доступа или задать свои (имя беспроводной сети, *пароль* для доступа к сети, уровень безопасности и тип шифрования) – рис. 9.8.

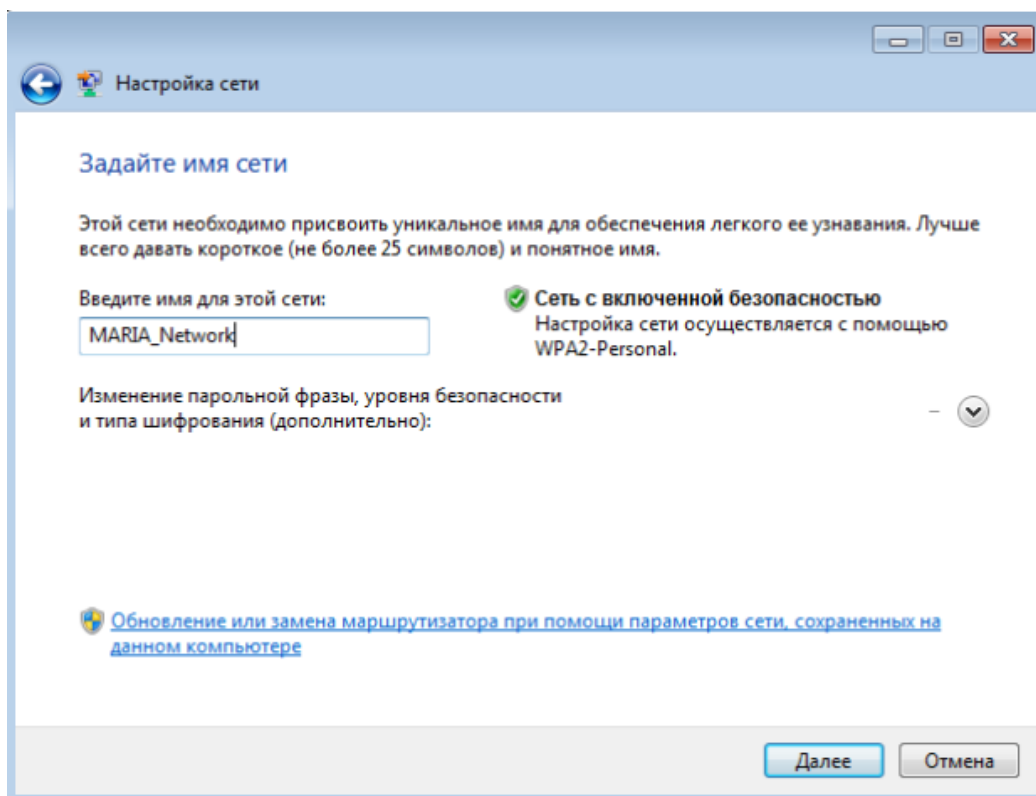


Рис. 9.8. Вводим имя сети (его придумываем сами)

После нажатия кнопки **Далее** произойдет настройка точки доступа (беспроводного маршрутизатора), генерация ключа безопасности и подключение нашего ноутбука к беспроводной сети (рис. 9.9 и рис. 9.10).

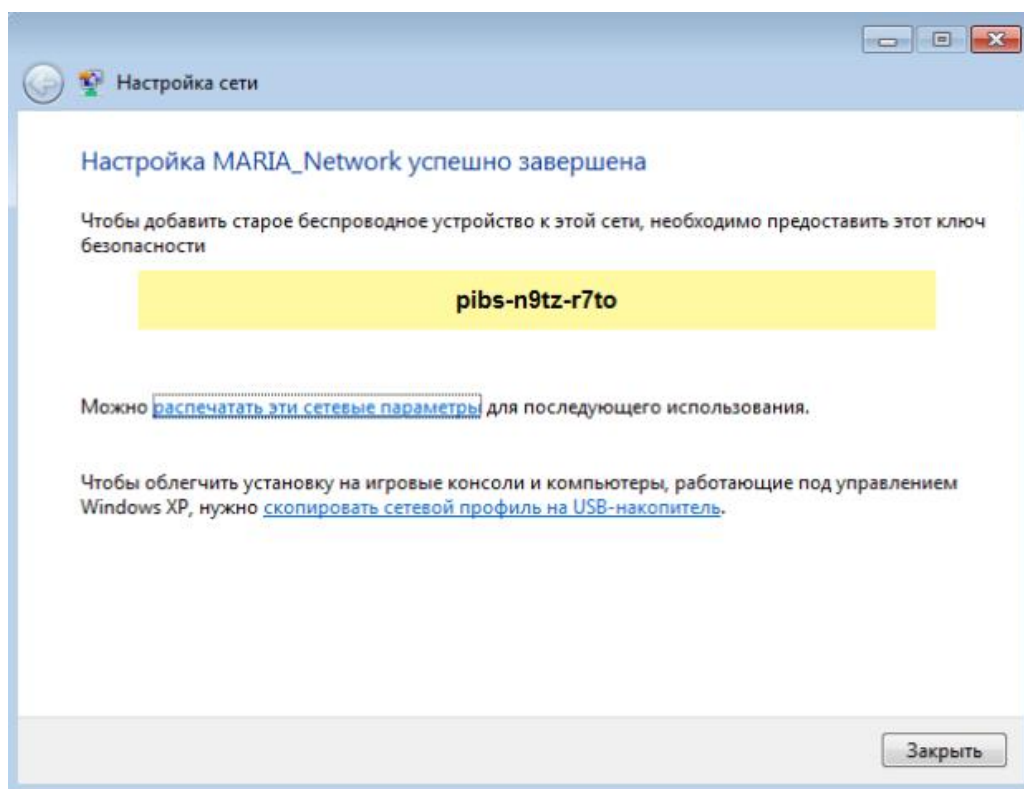


Рис. 9.9. Создание ключа безопасности

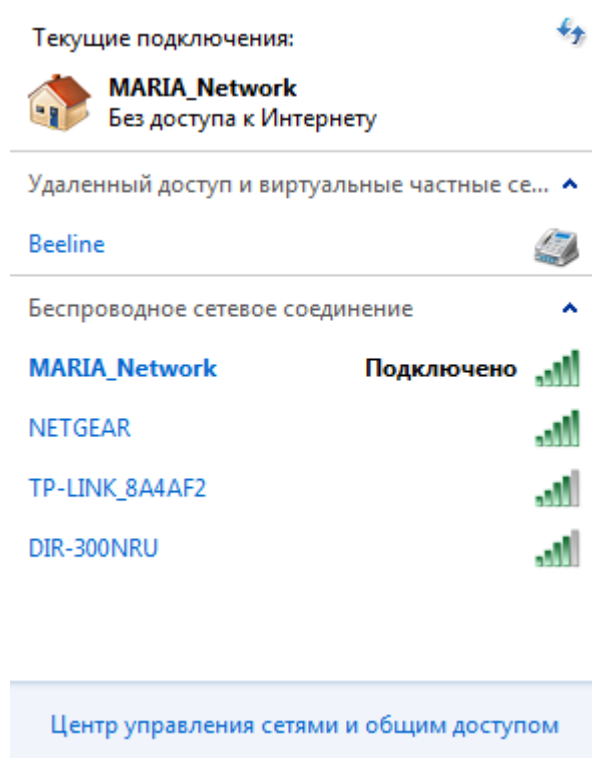


Рис. 9.10. Беспроводное соединение подключено

Примечание

Модель TL-WR1043ND имеет кнопку быстрой настройки защиты (QSS) для автоматической передачи ключа шифрования клиентскому устройству с такой же функцией. Поэтому, при подключении к нашей беспроводной сети нового компьютера под управлением Windows 7 (их может быть до 20 шт.), можно не вводить ключ безопасности, а просто нажать на эту кнопку на маршрутизаторе. Подключение к беспроводной сети произойдет автоматически (рис. 9.11).

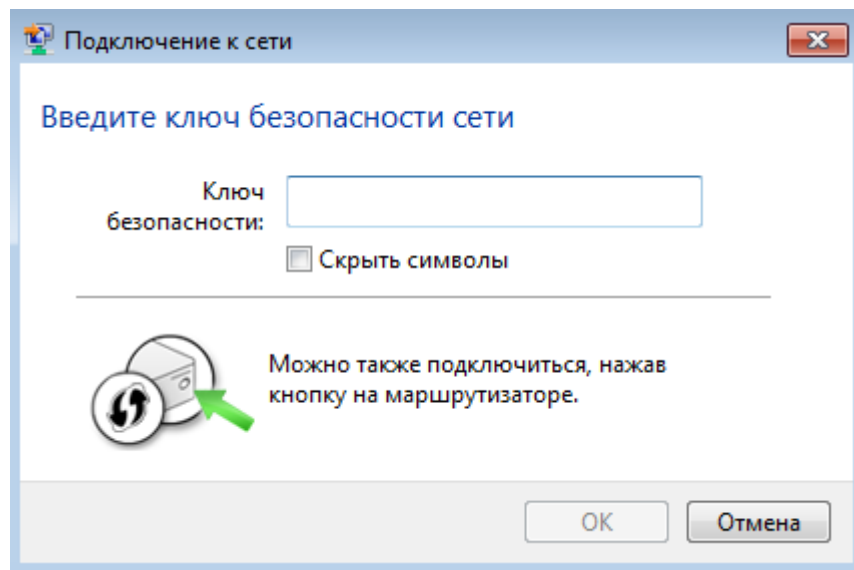


Рис. 9.11. Окно ввода ключа безопасности

Пример 2. Настройка на работу в Интернет Wi-Fi роутера Net Gear JWNR2000 в ручном режиме

В этой работе мы изучим, как можно с помощью Wi-Fi роутера подключить к *Интернет* два ПК: стационарный и ноутбук. Порты и индикаторы роутера приведены на рис. 9.12.

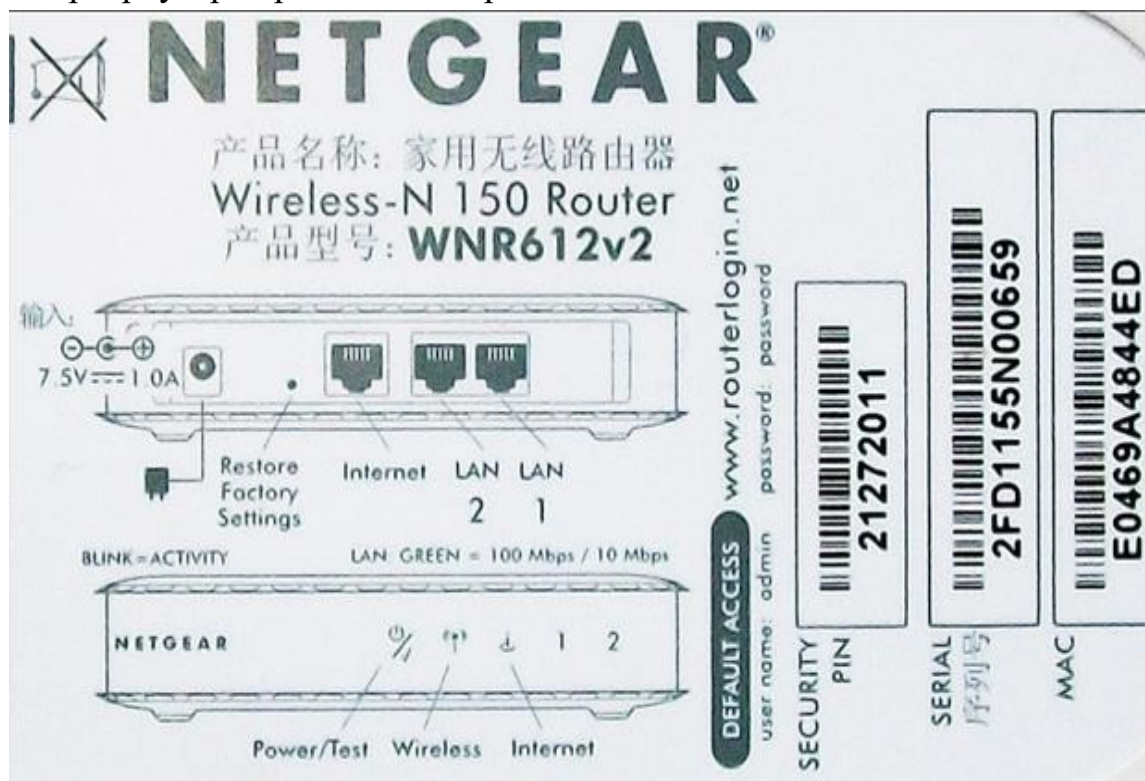


Рис. 9.12. Обозначение портов и индикаторов роутера Net Gear JWNR2000

Характеристики этой модели маршрутизатора для выделенной линии таковы:

- Частота - 2,4 ГГц
- Режимы - Infrastructure, WDS-Bridge
- Кнопки - Reset, WPS

Примечание

Кнопка *WPS* нужна для упрощения процесса настройки беспроводной сети. Нажатие *WPS* автоматически обозначает имя сети и задает *шифрование*, для защиты от несанкционированного доступа в *сеть*, при этом нет необходимости вручную задавать все параметры.

- Индикаторы - LAN, Power, WLAN, WPS
- Порты Fast Ethernet - 4 порта 10/100 Мбит/сек
- Порты WAN - 1 порт RJ-45
- Управление - Веб-интерфейс, GUI, SNMP

- Firewall - фильтрация по MAC-адресу, фильтрация пакетов, защита от DoS-атак

- Поддержка схем обеспечения безопасности беспроводной передачи WPA2-PSK; WPA-PSK; TKIP; AES; WEP-кодирование с 64- или 128-битным ключом

- Защищенные VPN-протоколы - PPTP, PPPoE

- Получение IP-адреса - Static IP, Dynamic IP

- QoS - Поддерживается

- Поддержка WMM (Wi-Fi Multimedia) - Есть

- DMZ - Поддерживается

- NAT - Поддерживается

- DHCP-сервер - Есть

- Максимальная скорость беспроводной передачи данных - 300 Мбит/сек

- Стандарты беспроводной связи - IEEE 802.11n, IEEE 802.11g, IEEE 802.11b

Шаг 1 – Настройка стационарного ПК для ОС Windows XP

Подключаем роутер согласно схеме на рис. 9.13.

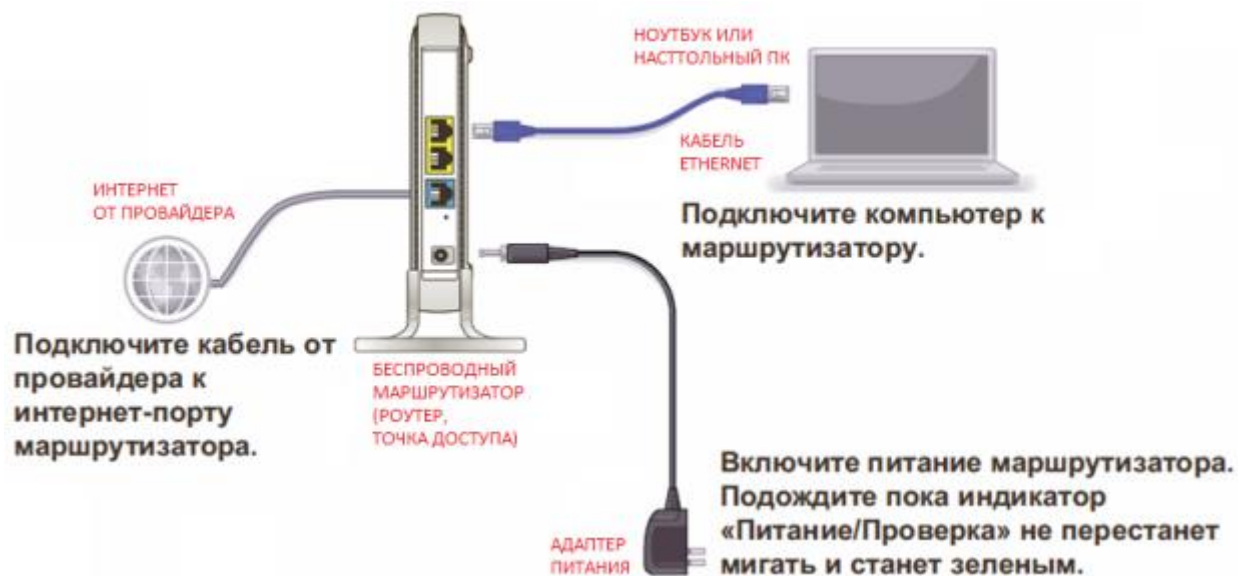


Рис. 9.13. Схема подключения устройств беспроводной сети к точке доступа

Далее нужно настроить протокол *TCP/IP* как на рис. 9.14.

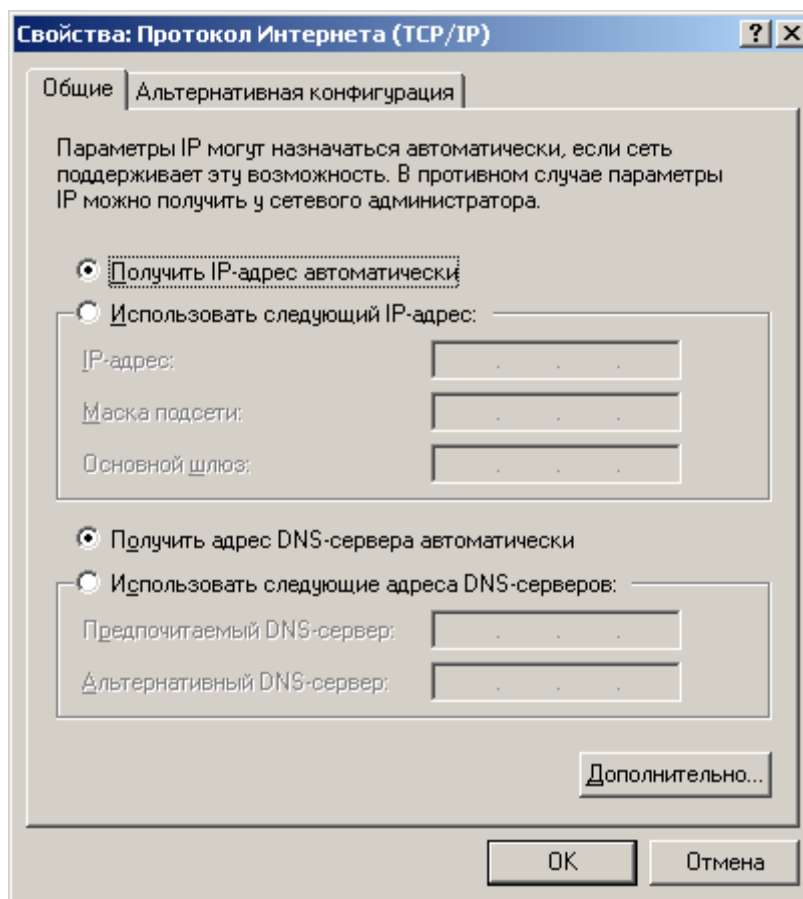


Рис. 9.14. Настройка протокола Интернет

Затем введите в браузере 192.168.1.1 и получите следующее окно (рис. 9.15). Вводим сюда *Имя пользователя* и *Пароль* (они написаны на этикетке роутера – см. рис. выше).

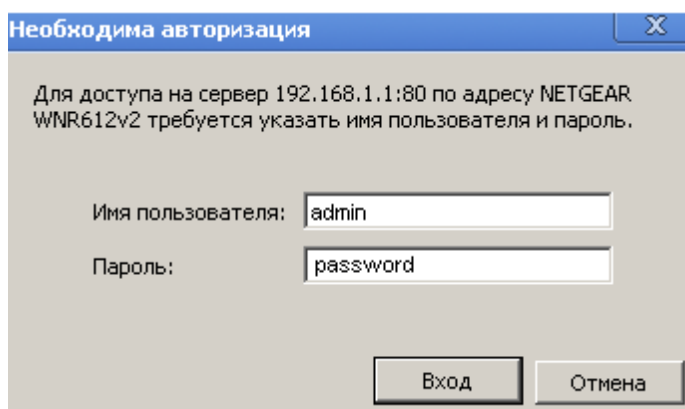


Рис. 9.15. Окно входа на сервер 192.168.1.1

После нажатия на кнопку **Вход** откроется окно **Основные настройки**. В программе имеется Мастер установки, но он здесь не очень хорош, поэтому лучше воспользоваться ручной настройкой роутера (рис. 9.16).

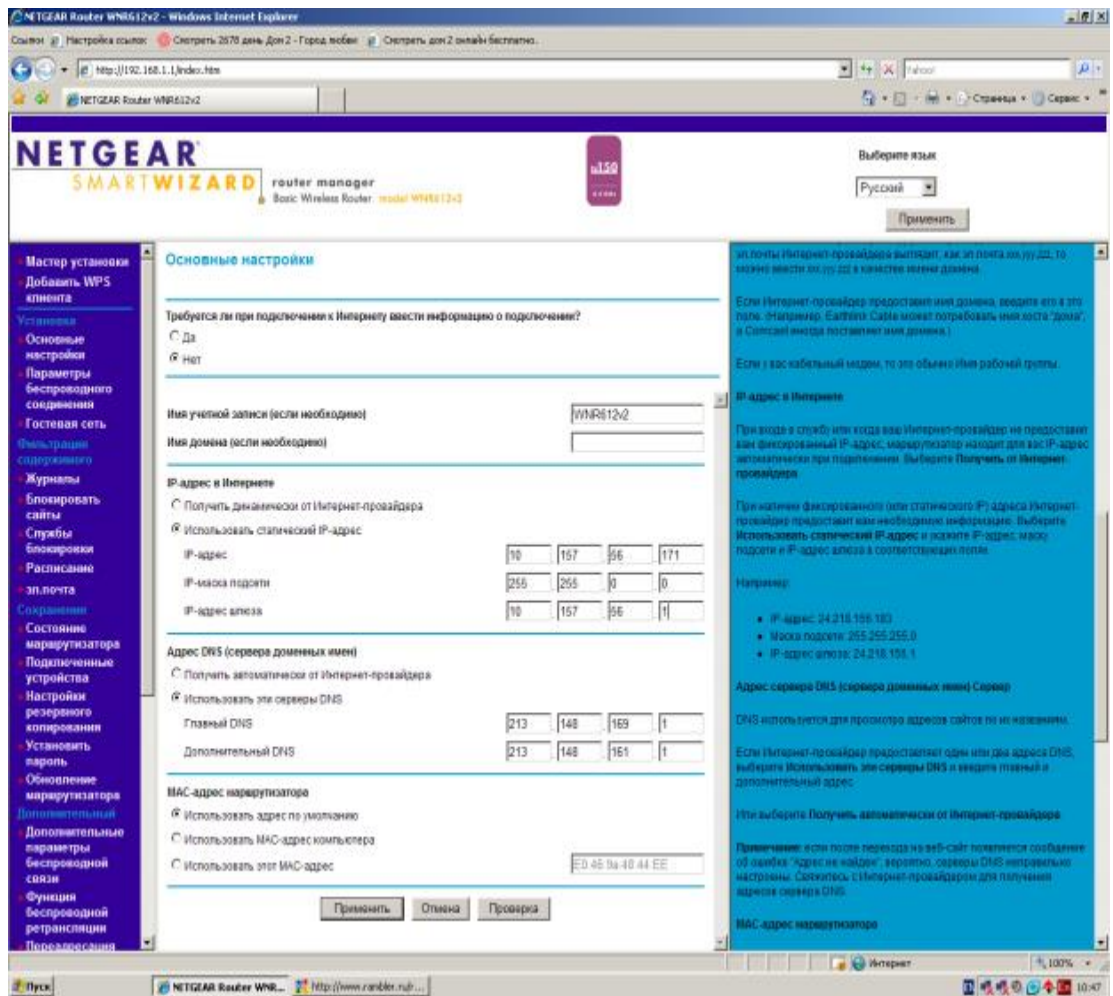


Рис. 9.16. Эти данные вводите в соответствии с договором провайдера Интернет

В данное окно вводим IP-адрес, IP-маску подсети и IP-адрес шлюза из договора с провайдером. Нажимаем на кнопку **Применить** – появляется другое окно **Основные настройки** (рис. 9.17).

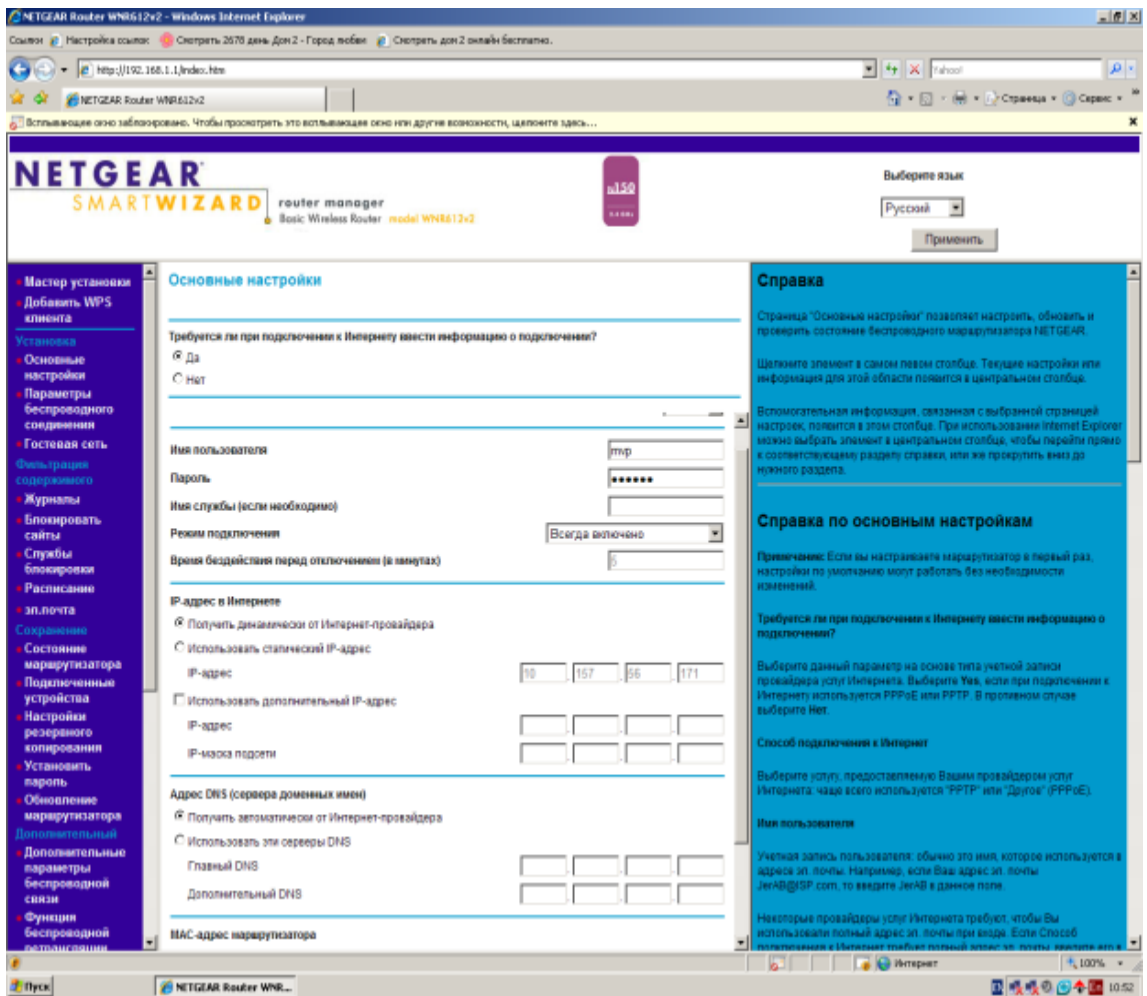


Рис. 9.17. Окно основные настройки

Здесь в соответствии с договором провайдера *Интернет* вводим **Имя пользователя** и **Пароль**. В этом окне же следует в списке **Поставщик услуг Интернета** выбрать протокол **PPPoE** (рис. 9.18). Нажимаем **Применить**.



Рис. 9.18. Из протоколов доступа выбираем протокол PPPoE

После обновления параметров роутера в *поле Сохранение* найдите опцию **Установить пароль** и замените *пароль* по умолчанию, т.е. **password** на какой-либо свой, например, **quthor**. Далее настройте окно **Параметры беспроводного соединения** – рис. 9.19.

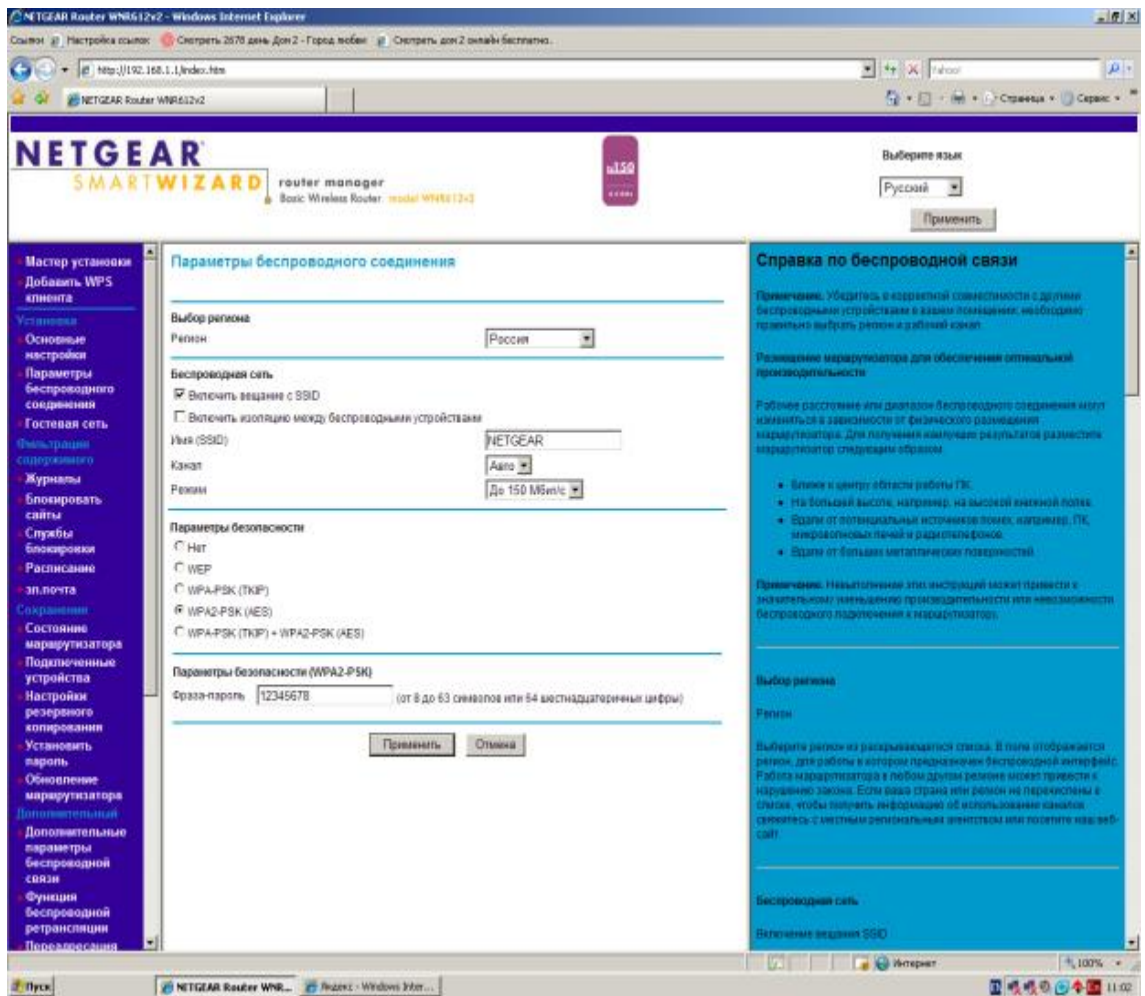


Рис. 9.19. Окно Параметры беспроводного соединения

Примечание

SSID – название беспроводной сети

Фраза – *пароль* здесь задана 12345678, но лучше ввести что-либо более сложное.

Совет

Для замены пароля наберите 192.168.1.1., введите *admin* и *quthor*, выберите команду **Параметры беспроводного соединения** и введите новый *пароль*, например, *masha+vova=love*

ШАГ 2 – Настройка Wi-Fi сети на ноутбуке для ОС Windows 7

Настроим работу Wi-Fi адаптера на ноутбуке, чтобы он смог получить *Интернет* от роутера NetGear. Выполните на ноутбуке команду **Панель управления-Сеть и Интернет-Подключение к сети** (рис. 9.20).

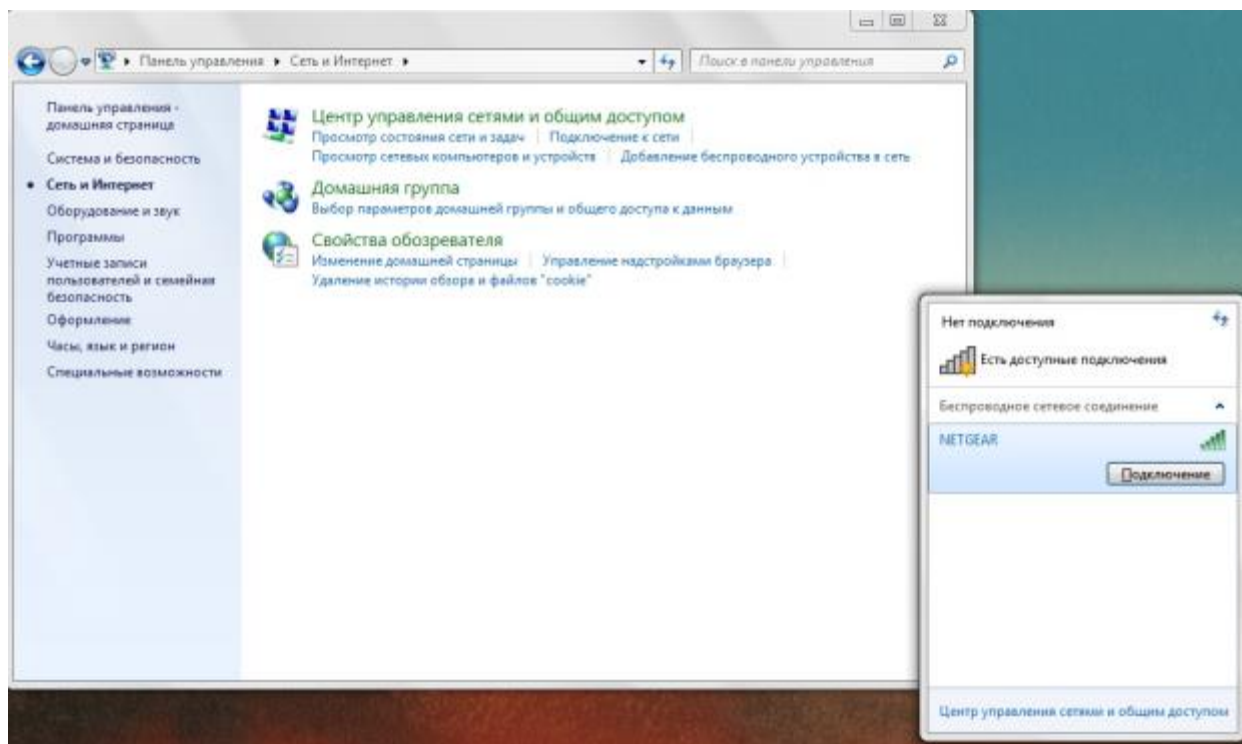


Рис. 9.20. Беспроводное сетевое соединение (роутер) ноутбук обнаружил

После нажатия на кнопку **Подключение** необходимо ввести фразу-*пароль* ключа безопасности (в нашем случае 12345678 или, если вы этот *пароль* изменили, то masha+vova=love) и *Интернет* на ноутбук будет подключен. *Интернет* запускается через любой *браузер* как при включенном стационарном ПК, так и без него. Лишь бы роутер был включен.

Настольная всенаправленная 8дБи антенна TL-ANT2408C

Беспроводная настольная антенна TL-ANT2408C работает на частоте 2.4-2.5ГГц с коэффициентом усиления 8дБи и позволяет существенно увеличить дальность беспроводного сигнала и повысить качество соединения. Антенна оснащена RP-SMA штекером, что обеспечивает совместимость с большинством беспроводных устройств – рис. 9.21.



Рис. 9.21. Беспроводная антенна для роутера модели TL-ANT2408C

Замена стационарной антенны маршрутизатора (точки доступа) на TL-ANT2408C заметно увеличит силу и дальность беспроводного сигнала. Поскольку данная антенна является всенаправленной, то нет необходимости поворачивать ее в ту или иную сторону для получения более четкого сигнала – антенна получает и отправляет сигналы во всех направлениях.

Примечание

дБи (dBi) - это децибелл по сравнению с "i", то есть по отношению к изотропному излучателю - идеальной антенне, диаграмма направленности которой представляет собой сферу, коэффициент усиления которой равен единице и КПД которой равен 100%. дБи (dBi) характеризует коэффициент усиления антенны и ее направленные свойства по сравнению с изотропным излучателем. Строго говоря, если говорят, что данная антенна имеет коэффициент усиления, например, 8 дБ, то на самом деле имеется ввиду 8 дБи.

Wi-Fi-адаптер TP-LINK TL-WN725N

Недорогой и компактный Wi-Fi-адаптер TP-LINK TL-WN725N показан на рис. 9.22.



Рис. 9.22. Wi-Fi адаптер TP-LINK TL-WN725N

Характеристики:

- Стандарт беспроводной связи - 802.11n, частота 2.4 ГГц
- Макс. скорость беспроводного соединения - 150 Мбит/с
- Интерфейс подключения - USB 2.0
- Защита информации (режим шифрования данных) - WEP, WPA, WPA2, 802.1x
- Мощность передатчика 20 dBm

Примечание

дБм (dBm) - это децибелл по сравнению с "m", в данном случае по отношению к милливатту. Иначе говоря, это значение того, на сколько децибелл данная мощность больше (или меньше) чем 1 мВт.

Беспроводной мини сетевой USB-адаптер (USB WLAN) TL-WN823N

Беспроводной мини сетевой *USB-адаптер* TL-WN823N предназначен для подключения ноутбука или настольного компьютера к беспроводной сети. Скорость беспроводного соединения до 300 Мбит/с. – рис. 9.23.



Рис. 9.23. Адаптер TL-WN823N с кнопкой WPS

Адаптер имеет функцию программной точки доступа. Включив этот режим, пользователи получают совместный *доступ* к прочим Wi-Fi устройствам (ноутбуки, смартфоны или планшетики) со своих ноутбуков

или персональных компьютеров с проводным подключением. *Адаптер* позволяет произвести настройку безопасности одним нажатием кнопки *WPS* (*Wi-Fi Protected Setup*). Пользователи могут моментально настроить защиту сети одним нажатием на маршрутизаторе кнопки *WPS*, после чего автоматически устанавливается соединение, защищенное режимом шифрования *WPA2*, который считается более надежным по сравнению с шифрованием *WEP*. Это не только быстрее обычной процедуры настройки безопасности, но и более удобно, так как пользователям даже не придется запоминать *пароль*.

USB-адаптер Wi-Fi TP-Link TL-WN822N

Модель TP-Link TL-WN822N хорошо справляется с поиском слабого сигнала Wi-Fi, так как имеет две мощных всенаправленных антенны с коэффициентом усиления 3 дБи (это больше, чем у стандартного ноутбука). *Адаптер* имеет кнопку *QSS* для быстрой настройки защищенного беспроводного соединения, а также светодиодный *индикатор* активности. *Функция* *QSS* (*Quick Secure Setup*) необходима для быстрой настройки защищенного беспроводного соединения. Эта *функция* является аналогом технологии *WPS* (*Wi-Fi Protected Setup*), только называется по-другому. При включённом беспроводном модуле на маршрутизаторе устройства тут же обнаруживают друг друга и запрашивают у пользователя разрешение на соединение. Вы нажимаете кнопки *QSS* на маршрутизаторе и *USB-адаптере* и тем самым настраиваете защищенное с использованием алгоритмов шифрования *WPA2* соединение. Пользователю останется лишь задать желаемый *пароль* (рис. 9.24).



Рис. 9.24. USB-адаптер Wi-Fi TP-Link TL-WN822N

Характеристики адаптера:

- Стандарты - IEEE 802.11b/g/n (300 Мбит/с)
- Порты - 1 x Mini-USB
- Частотный диапазон, ГГц - 2,4–2,4835
- Антенны - 2 x внешняя складная всенаправленная, 3 дБи
- Безопасность - WEP, WPA/WPA2, WPA-PSK/WPA2-PSK
- Режимы работы - Ad-Нос/в инфраструктуре
- Дополнительные функции - поддержка Sony PSP, QSS

Пример 3. Настройка WI-FI адаптера TP-LINK

Вставляем *адаптер* в *USB порт* ПК и устанавливаем *драйвер* адаптера (рис. 9.25).

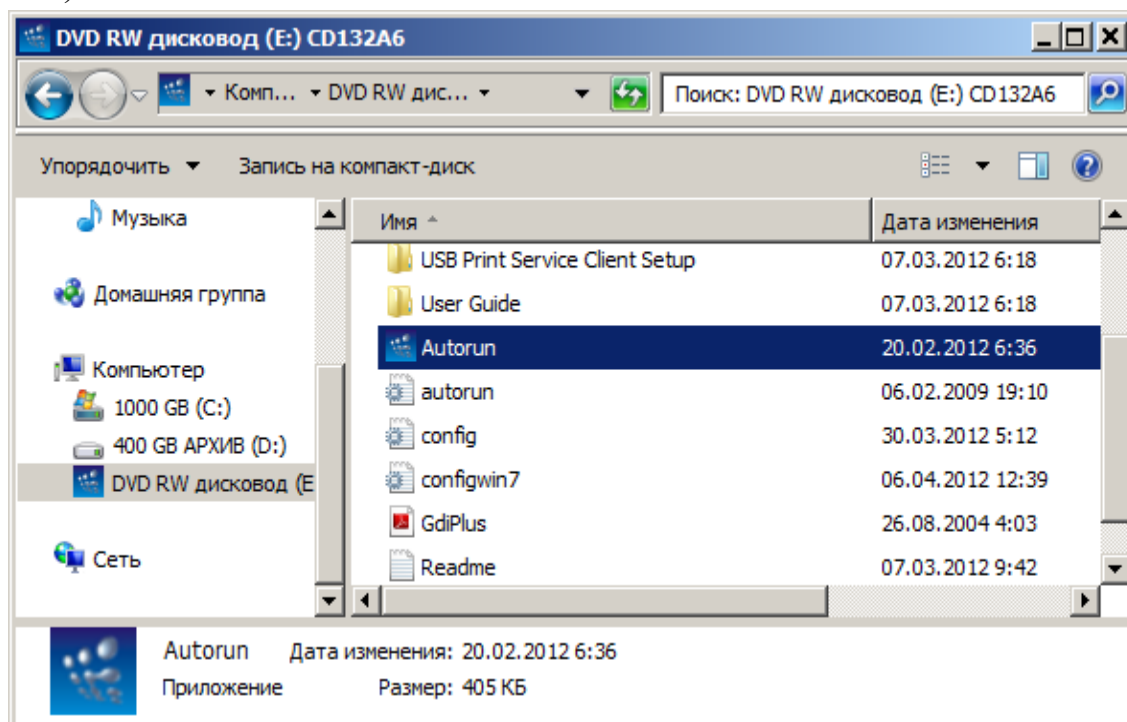


Рис. 9.25. Начинаем установку драйвера WI-FI адаптера

Указываем модель адаптера (рис. 9.26) и начинаем его настройку (рис. 9.27).



Рис. 9.26. Выбираем модель адаптера

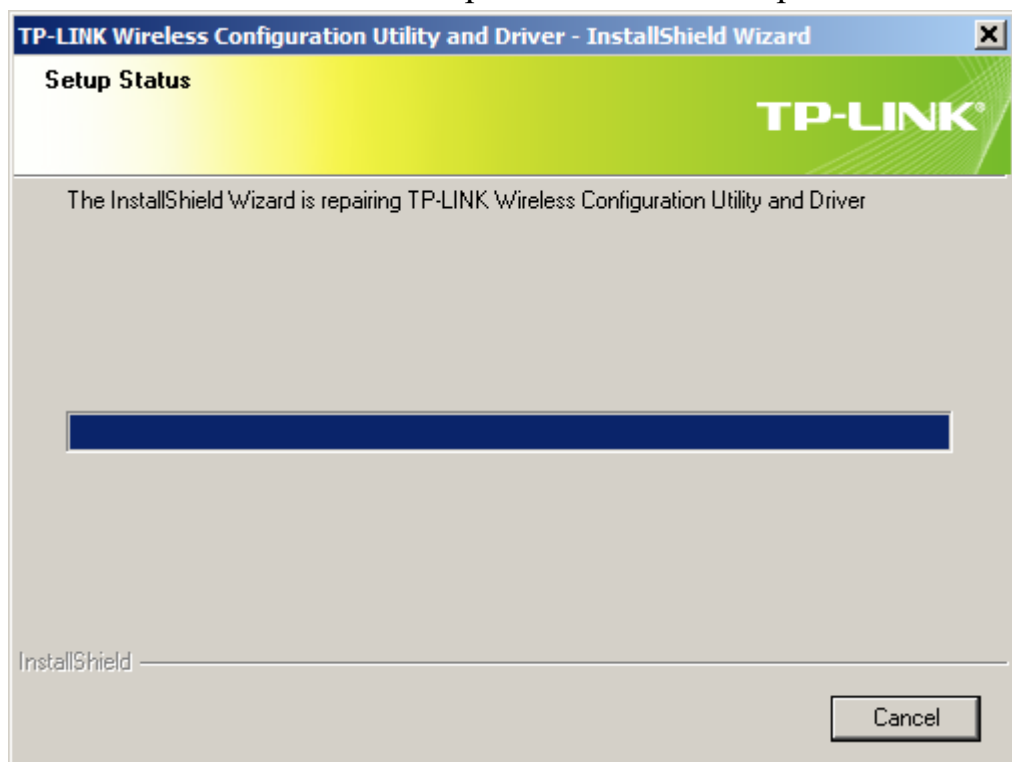


Рис. 9.27. Мастер начинает установку драйвера адаптера

После установки адаптер обнаружит ближайшие беспроводные сети, в том числе и наш роутер TP-Link (рис. 9.28).

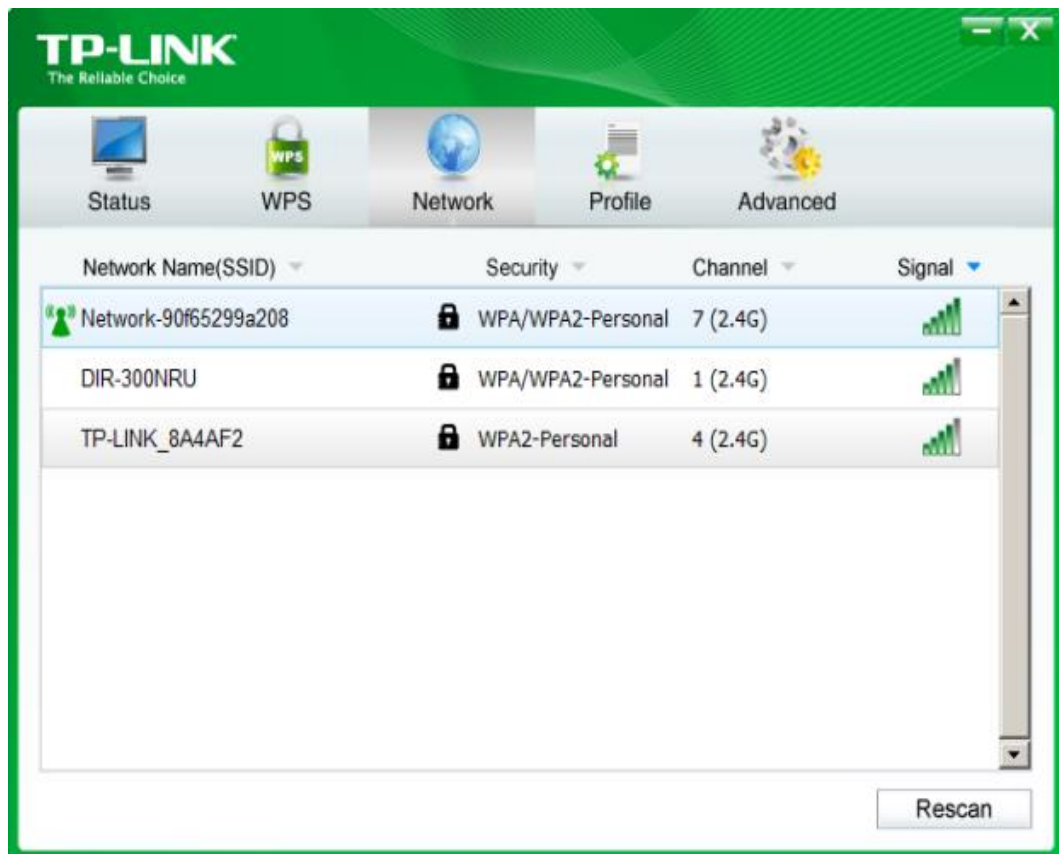


Рис. 9.28. Беспроводная сеть с точкой доступа TP-LINK обнаружена

Для подключения к сети мы должны или нажать на роутере кнопку WPS (рис. 9.29) или ввести, написанный на роутере PIN-код, у нас это число 52035098 – рис. 9.30 и рис. 9.31.



Рис. 9.29. Предложение о настройке безопасности сети путем нажатия на кнопку WPS



Рис. 9.30. Этикетка роутера с его PIN кодом



Рис. 9.31. Окно ввода PIN кода

Результат – беспроводная сеть настроена (рис. 9.32).



Рис. 9.32. ПК к сети Интернет подключен

Краткие итоги

В работе мы на практике рассмотрели работу с конкретными моделями беспроводных устройств, а именно: настройку беспроводного маршрутизатора TL-WR1043ND и Wi-Fi роутера *Net Gear JWNR2000*. Мы изучили настройку Wi-Fi сети на ноутбуке для ОС *Windows 7*. Познакомились с характеристиками настольной всенаправленной 8дБи антенны TL-ANT2408C, Wi-Fi-адаптера TP-LINK TL-WN725N, беспроводного мини сетевого *USB-адаптер (USB WLAN) TL-WN823N* и *USB-адаптером Wi-Fi TP-Link TL-WN822N*. К лабораторной работе прилагаются скринкасты и видеоролики.

Лабораторная работа №8

Программы для работы в сетях LAN и WAN

Пример 1. Winsent – бесплатная программа для общения в локальной сети

Чат – является программой для общения с другими пользователями сети (обмен текстовыми сообщениями).

Winsent Messenger это программа, предназначенная для быстрого обмена сообщениями и общения в локальной сети (рис. 10.1). Сайт программы <http://www.winsent.ru/>. Программа может использоваться как в домашней локальной сети, так и в локальной сети предприятия, офиса, корпоративной локальной сети. *Winsent Messenger* не требует использования выделенного сервера, и полностью совместим со службой сообщений (**net send**). Работает в любой версии *Windows* (7/Vista/XP/2000).

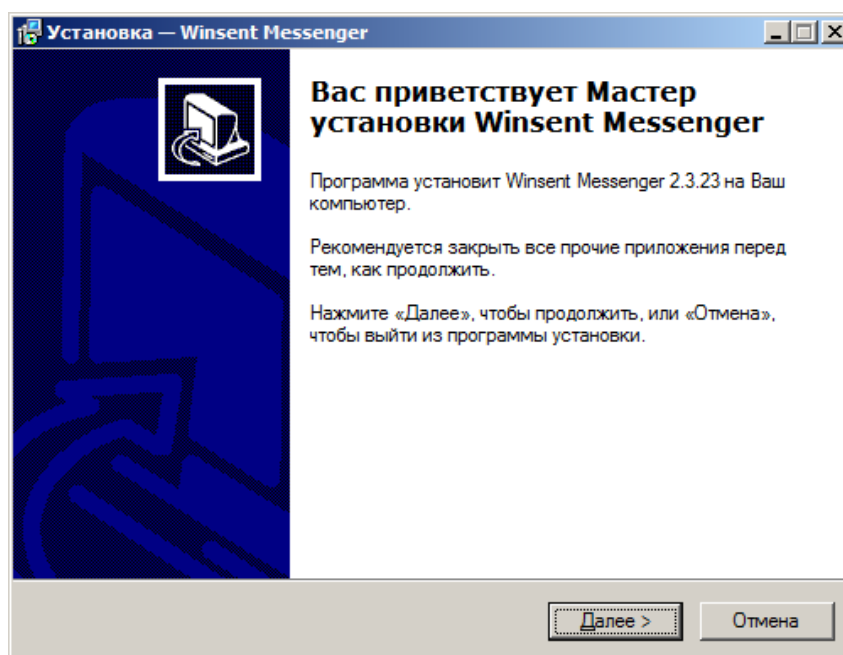


Рис. 10.1. Окно начальной установки программы Winsent Messenger

Если на вашем ПК программа запущена, то вы можете отправить сообщение любому ПК в сети, даже если на нем эта программа не стоит. Запустите программу – в тее появиться ее значок (рис. 10.2).



Рис. 10.2. Значок программы WinSent

Напишите имя рабочей группы или пользователя, затем пишите текст сообщения и отправляйте его (рис. 10.3)

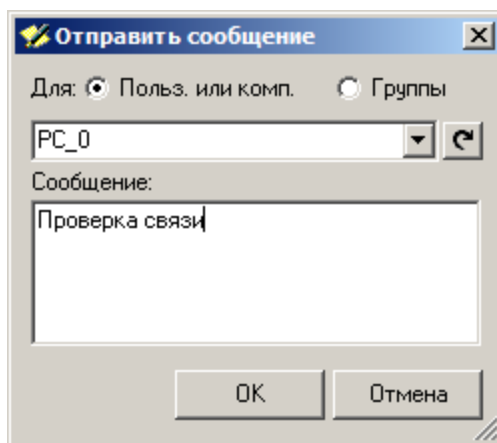


Рис. 10.3. Указываем получателя, пишем текст сообщения

Пример 2. Radmin - программа удаленного управление ПК по сети

Radmin - популярная программа для работы на удаленном компьютере в режиме реального времени. Она работает и в LAN, и WAN. Radmin включает в себя средство обмена файлами, текстовый и голосовой чат, и другие полезные функции. Во время работы с удаленным компьютером, вы можете не беспокоиться за безопасность своих данных: Radmin работает в режиме защиты данных, при котором все передаваемые данные защищены по стандарту AES – рис. 10.4.

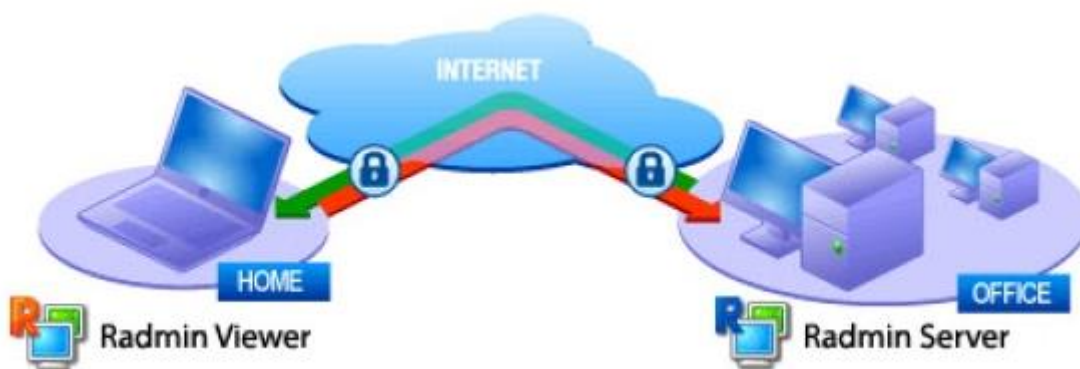


Рис. 10.4. Схематическое изображение работы с удаленным ПК в программе Radmin

Итак, перед началом работы оба компьютера должны иметь выход в Интернет или быть подсоединенными к общей локальной

сети (*LAN*). Предположим, что вы хотите установить *связь* вашего домашнего (у вас дома) и вашего офисного (у вас на работе) ПК. Установите на обоих ПК программы *Radmin Server* и *Radmin Viewer*.

Настройте *Radmin Server* на удаленном (ведущем, администраторском) компьютере

Запустите *Radmin Server*, например, на на PC_1. Щелкните правой кнопкой мыши на иконке *Radmin Server* в трее и выберите команду **Настройки Radmin Server-Права доступа** и установите *пароль* и *права доступа* к *Radmin Server* – рис. 10.5. Например, даем команду разрешить **Полный доступ** (ОК).

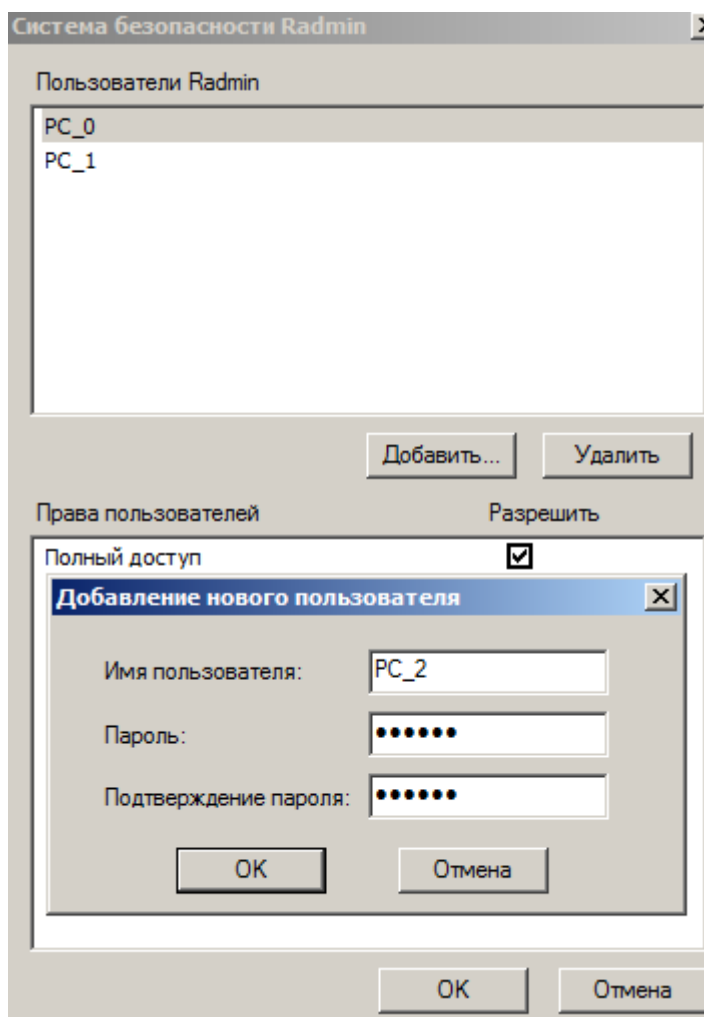


Рис. 10.5. Добавляем пользователей на Radmin Server

Запишите *IP адрес* Вашего компьютера, для этого наведите мышкой *курсор* на иконку *Radmin Server* (рис. 10.6).

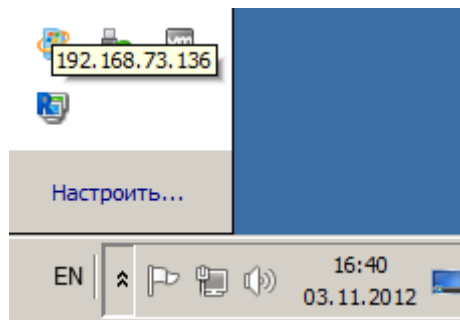


Рис. 10.6. Узнаем IP для нашего ПК (192.168.73.136)

Настройте Radmin Viewer на локальном (ведомом) компьютере (хосте)
Запустите Radmin Viewer на локальном компьютере, например, PC_2, и создайте новое подключение командой **Соединение-Соединиться** с (рис. 10.7).

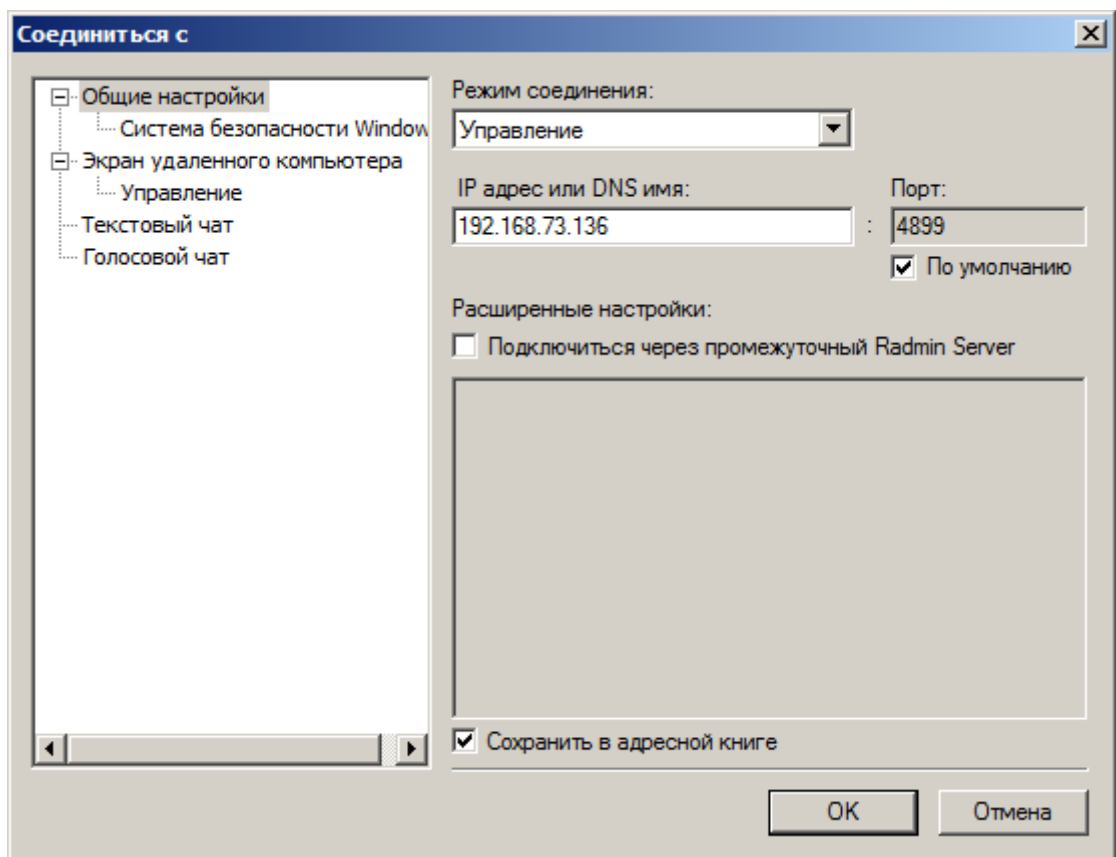


Рис. 10.7. Одно для соединения с ведущим ПК

Укажите *IP адрес* компьютера, на котором установлен и настроен Radmin Server. Затем выберите режим подключения и введите *имя пользователя* и *пароль*, заданные ранее в настройках Radmin Server на удаленном компьютере (рис. 10.8).

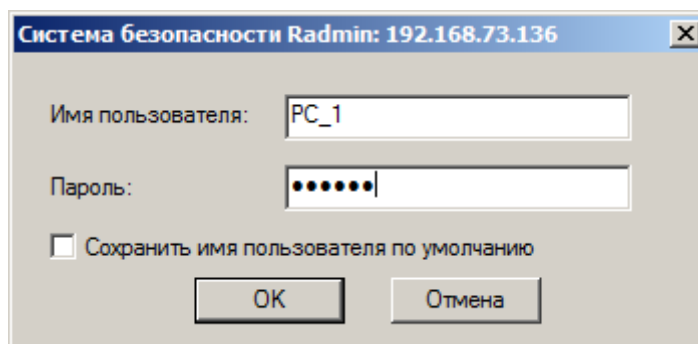


Рис. 10.8. Вводим заданные ранее на сервере имя и пароль пользователя

После нажатия на кнопку ОК видим *рабочий стол* удаленного ПК (рис. 10.9).

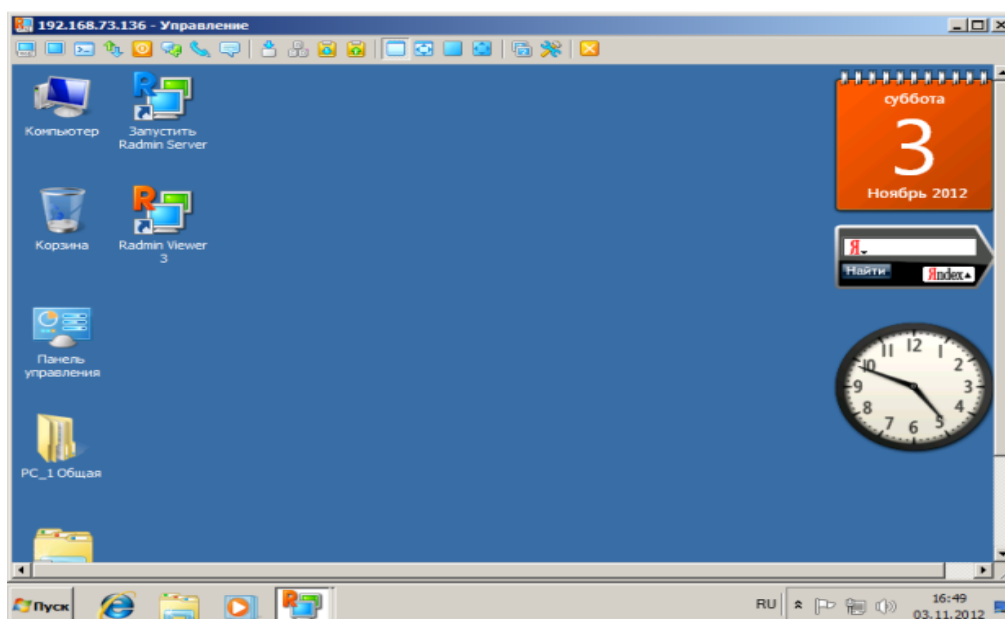


Рис. 10.9. Окно для управления удаленным (ведущим) ПК с ведомого ПК

Окно программы Radmin Viewer приведено на рис. 10.10.

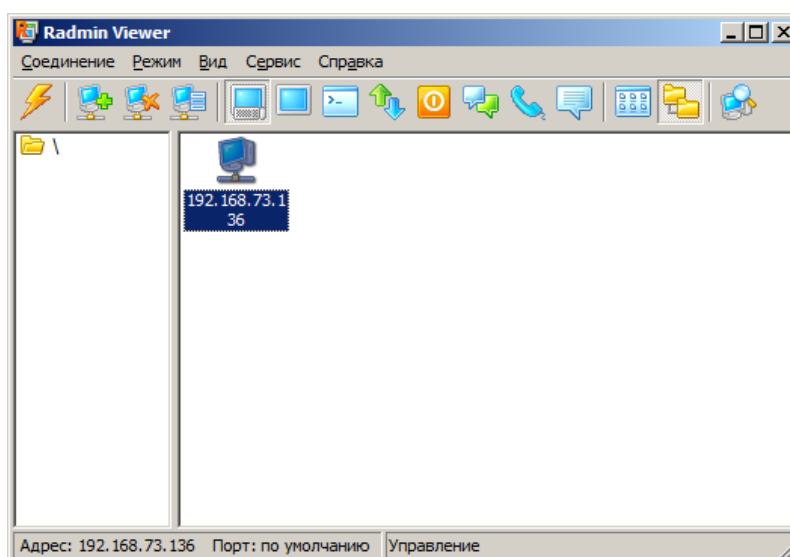


Рис. 10.10. Окно программы Radmin Viewer

Таким образом, программа Radmin состоит из двух частей: клиентской (Radmin Viewer) и серверной (Radmin Server). Вы устанавливаете Radmin Server на удаленном компьютере и получаете возможность видеть экран удаленного компьютера на экране своего компьютера. Ваши манипуляции передаются на удаленный компьютер. Удаленный компьютер может располагаться в Интернет или в локальной сети. Иначе говоря, с помощью Radmin вы можете работать у себя в офисе, не вставая из-за домашнего компьютера

Примечание

В принципе, на каждом ПК можно установить и запускать одновременно и Radmin Server, и Radmin Viewer. Они друг другу не мешают, зато можно управлять удаленным ПК в обе стороны.

Пример 3. Управление компьютером в программе Team Viewer

TeamViewer (<http://www.teamviewer.com/ru/download/windows.aspx>) - программа для осуществления удаленного доступа к компьютеру по сети. Программ устанавливается на удаленном компьютере (хост) и на компьютере для администрирования (администратор). Единственная настройка администратора – пройти авторизацию. После этого у вас на рабочем столе появится аналогичное окно удаленного компьютера, и вы сможете управлять им как обычным ПК. Помимо управления удаленным компьютером, с помощью данной программы можно передавать файлы и общаться в чате. Итак, устанавливается TeamViewer на оба сетевых ПК, затем запускается программа. После старта автоматически получают ID и Пароль (рис. 10.11).

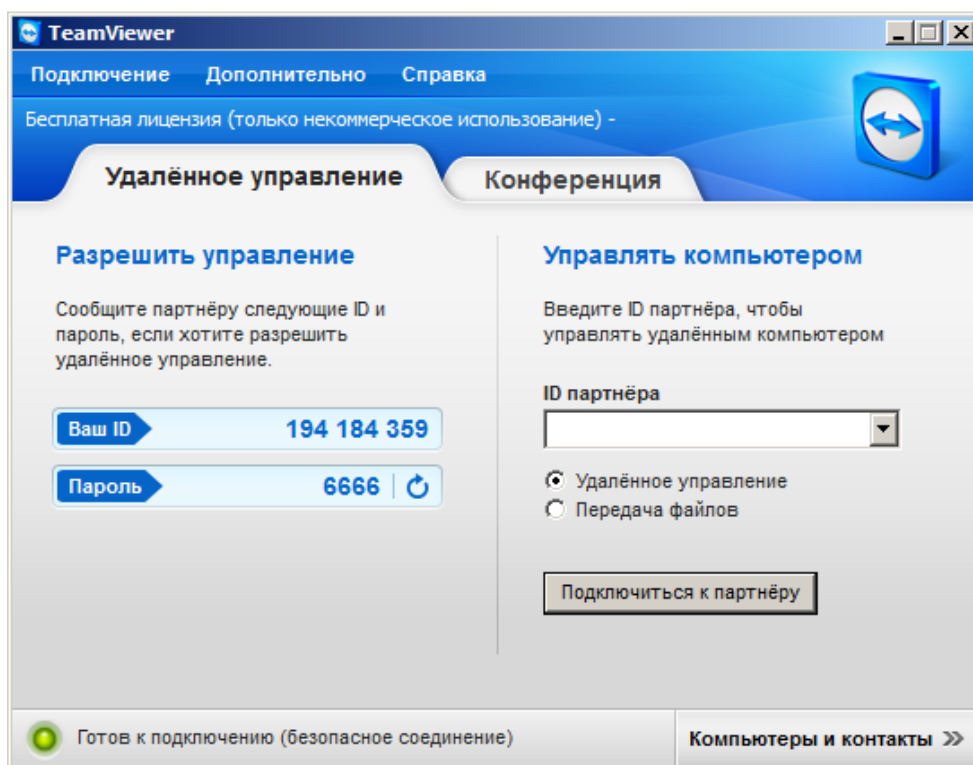


Рис. 10.11. Получаем уникальный идентификатор нашего ПК и пароль на вход в него

Предположим, что на втором ПК мы получили ID 194 187 481 и Пароль 8610. Вводим эти данные. После нажатия на кнопку **Подключиться к партнеру** вы сможете работать на удаленном ПК, как на своем или в режиме **Удаленное управление** (рис. 10.12), или в режиме **Передача файлов** (рис. 10.13).

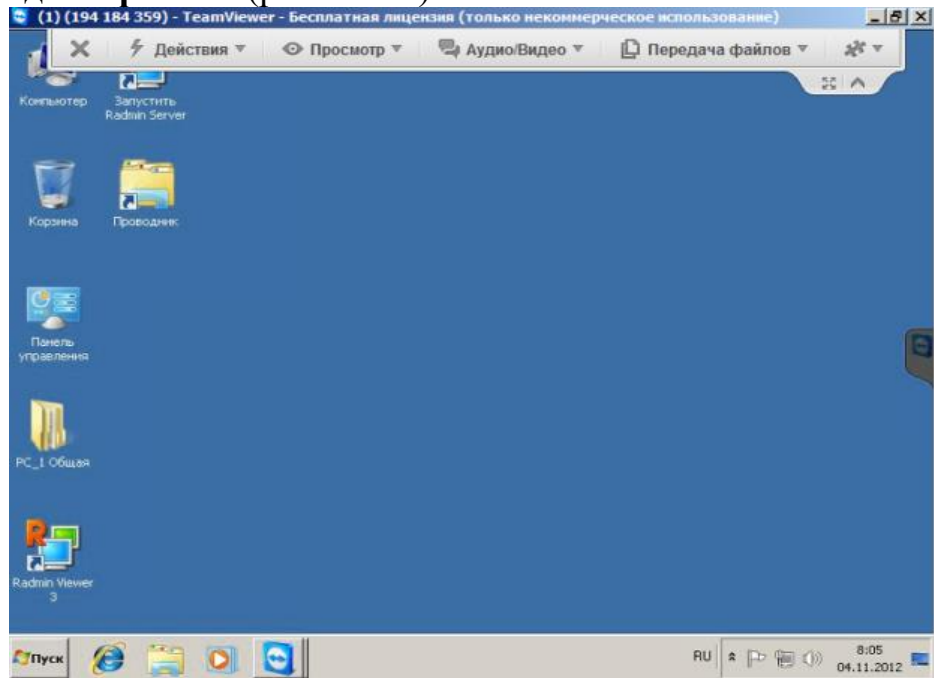


Рис. 10.12. Главное окно удаленного ПК (режим Удаленное управление)

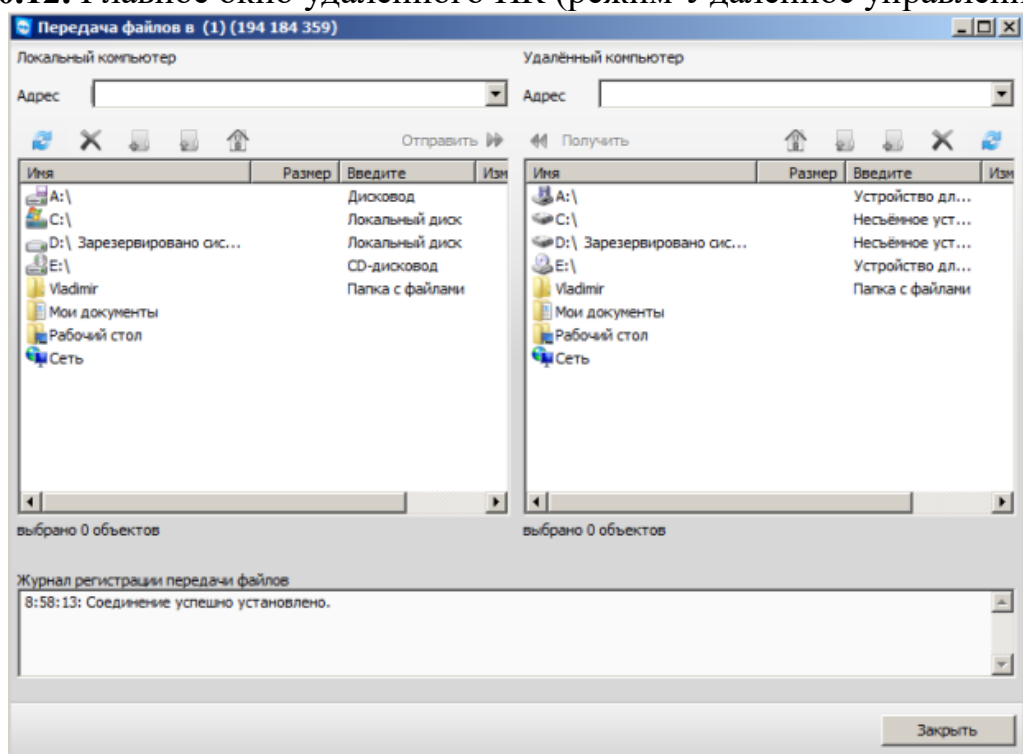


Рис. 10.13. Главное окно удаленного ПК (режим Передача файлов). Слева Локальный, а справа – Удаленный ПК

В заключение заметим, что *программа* полностью работоспособна и бесплатна, но постоянно предлагает приобрести коммерческую лицензию, что может раздражать ее пользователей.

Задания

- Создайте на удаленном ПК сетевой ресурс - папку 123456 и сделайте к ней общий доступ. Какие ограничение в ОС Windows XP устанавливаемое для сетевого ресурса (размер создаваемых файлов; максимальное число пользователей, которые могут подключиться к ресурсу; время работы каждого пользователя; дисковое пространство, выделяемое каждому пользователю).

- Установите связь между ведомым и ведущим ПК не через локальную сеть, а через Интернет.

- Осуществите пересылку файлов с локального на удаленный компьютер командой **Режим-Передача файлов** (рис. 10.7).

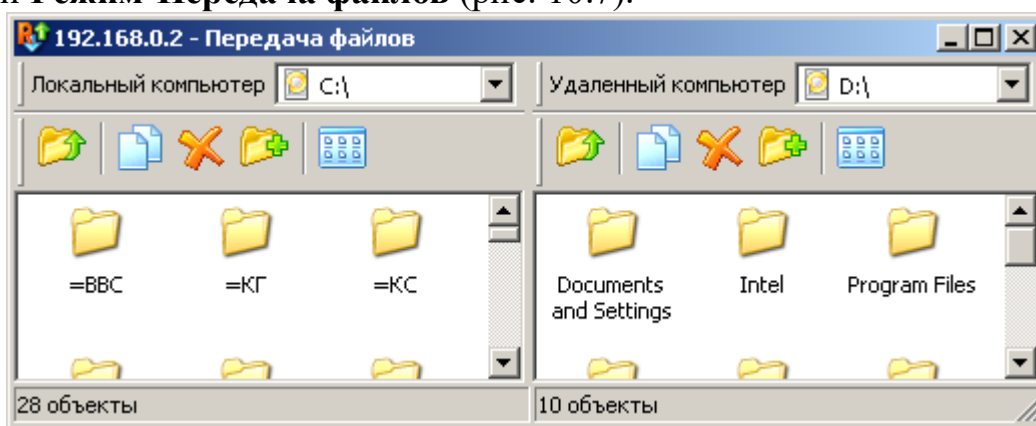


Рис. 10.14. Окно отправки файлов между компьютерами

Командой **Режим-Текстовый чат** организуйте обмен текстовыми сообщениями между ПК (рис. 10.8).

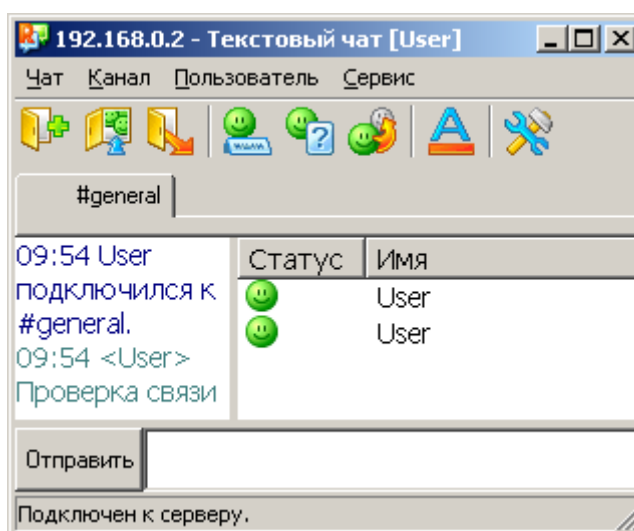


Рис. 10.15. Режим обмена текстовыми сообщениями между ПК
Произведите выключение удаленного ПК (рис. 10.9).

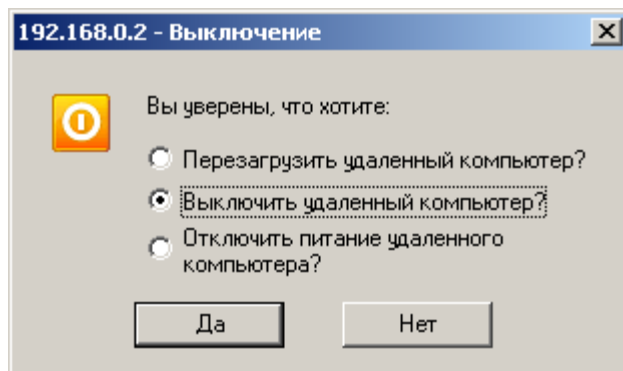


Рис. 10.16. Выключение удаленного ПК

- Установите программу и научитесь удаленно управлять ПК
- Отправьте файл удаленному ПК
- Отправьте на удаленный ПК текстовое сообщение

Краткие итоги

В лабораторной работе мы научились работать в трех полезных сетевых программах:

- Winsent (программа для общения в локальной сети)
- Radmin (программа удаленного управления ПК по сети)
- Team Viewer (программа управления компьютером с использованием

Интернет)

Для лучшего понимания этих тем к работе прилагается скринкаст

Домашнее задание:

Сделать несколько скриншотов использования любого программного обеспечения для удаленного управления ПК. Это может быть второй домашний ПК, ПК друга или виртуальная машина.

Лабораторная работа №9

Назначение серверу роли "Контроллер домена"

Создаем новый домен в новом лесу

Ниже мы опишем последовательность действий для того, чтобы *сервер* сделать **Контроллером Домена**, т.е. чтобы "поднять" на нем службу **Active Directory**.

Сервер, на котором расположена служба *Active Directory*, называется контроллером домена. *Active Directory* имеет иерархическую структуру базы, состоящей из объектов. Объекты разделяются на три основные категории: **ресурсы** (например, принтеры), **службы** (например, электронная почта) и **учётные записи** (пользователей и компьютеров). *Active Directory* предоставляет информацию об объектах, позволяет управлять объектами и доступом к ним.

Нажимаем **Пуск - Управление данным сервером** – рис. 11.1.

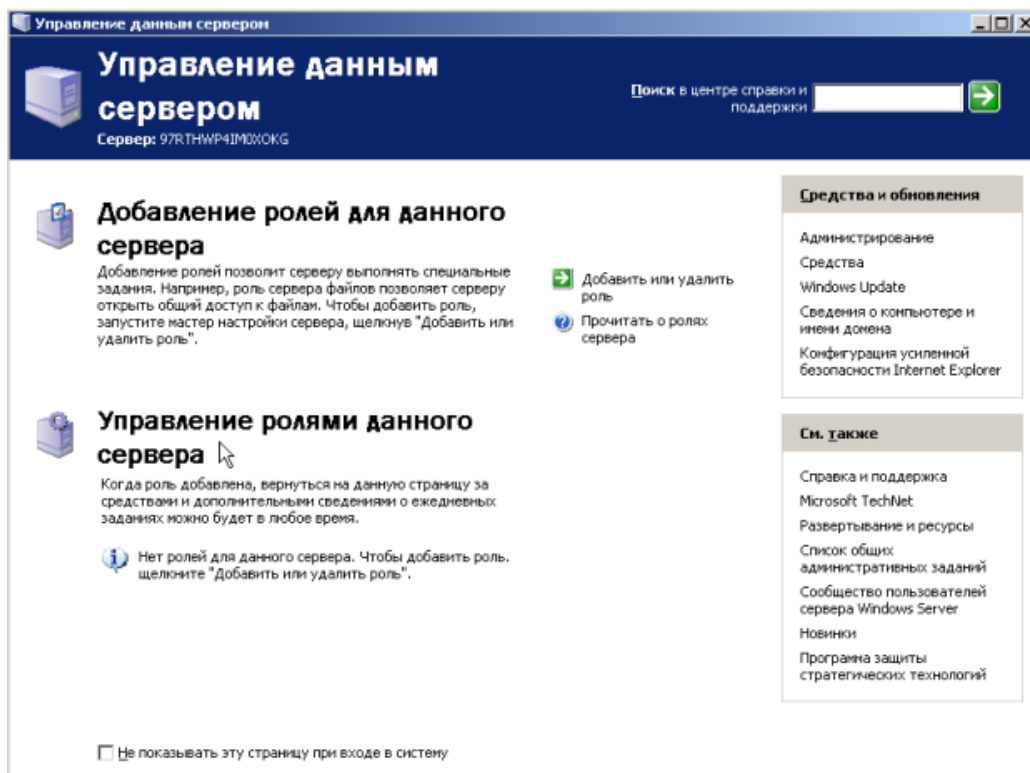


Рис. 11.1. Окно Управление данным сервером

Нажимаем **Добавить** или **удалить** роль, устанавливаем *переключатель* в положение **Особая конфигурация** – рис. 11.2

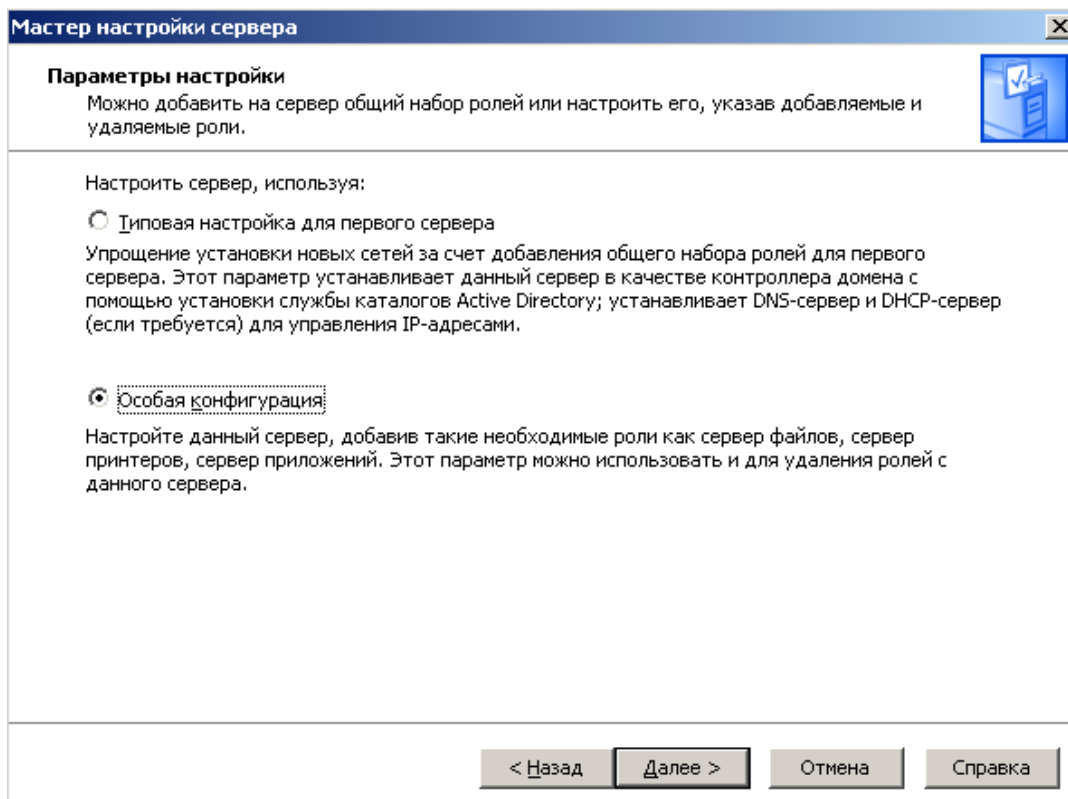


Рис. 11.2. Переключатель в положении Особая конфигурация

Из списка доступных ролей выбираем "Контроллер домена" (Active Directory) – рис. 11.3.

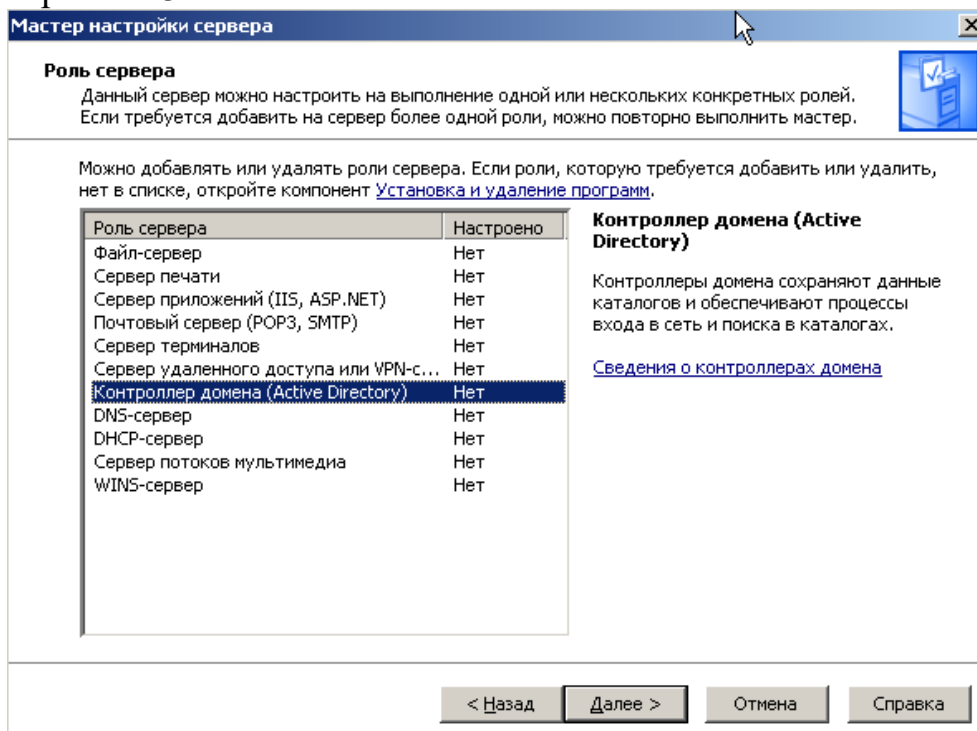


Рис. 11.3. Выбираем роль для сервера из списка

Добавить *контроллер* домена в существующем домене мы не можем, так как у нас нет доменов. Поэтому, для создания домена, выбираем *переключатель* **Контроллер домена в новом домене** – рис. 11.4.

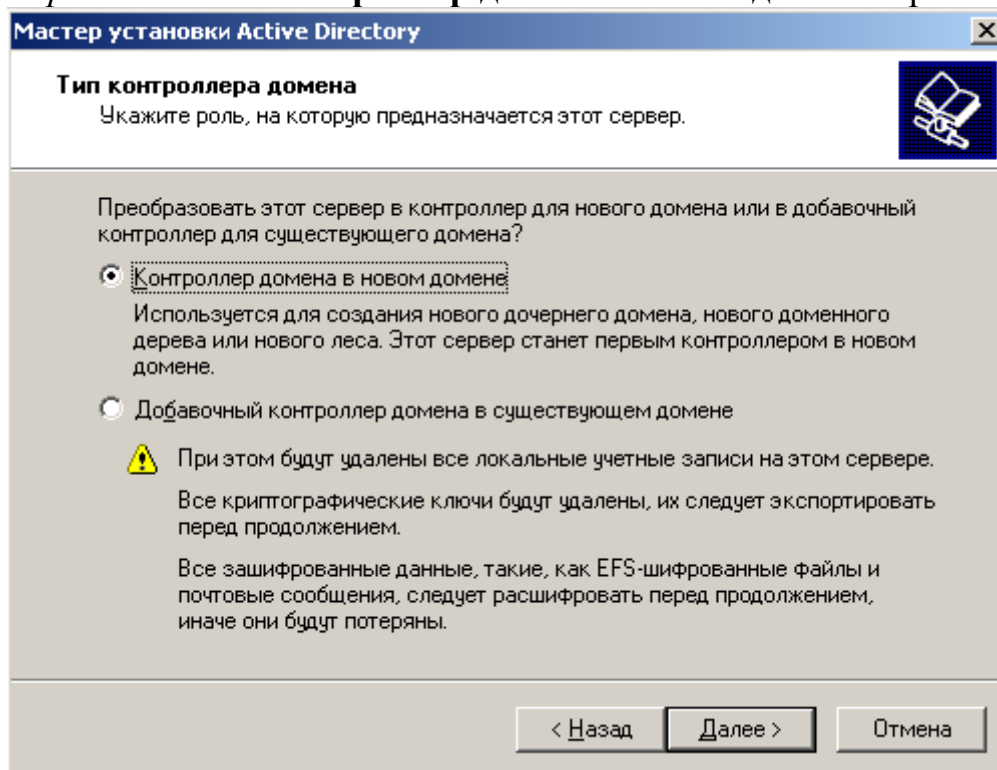


Рис. 11.4. Устанавливаем переключатель Контроллер домена в новом домене

Далее задействуем *переключатель* **Новый домен в новом лесу** - рис. 11.5. Что такое *домен* и *лес* мы поясним в конце этой лекции.

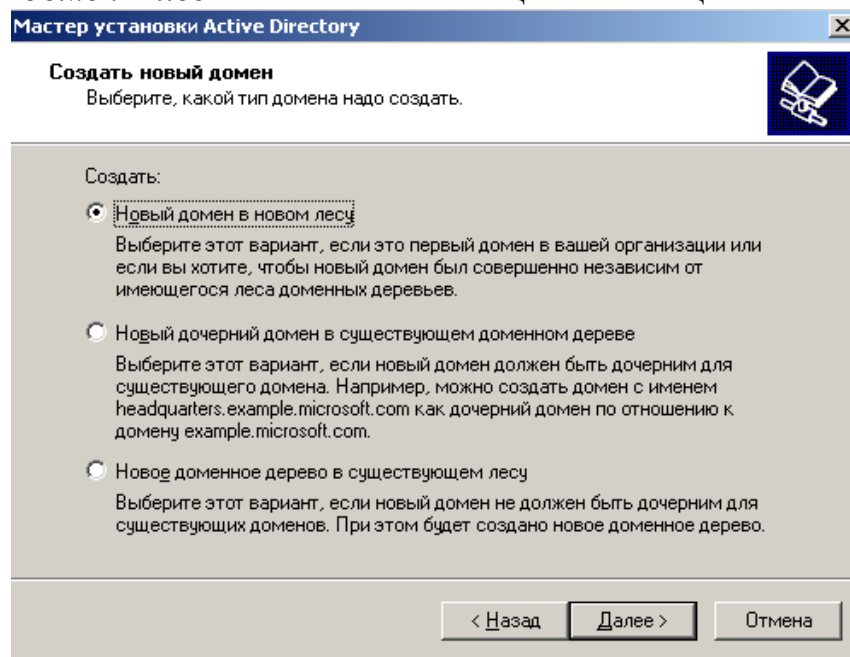


Рис. 11.5. Включаем опцию **Новый домен в новом лесу**

На следующем шаге пишем **полное DNS-имя нового домена**. Домены вида **domaine.com** или **domain.ru** имеют *внешнее пространство имен*, опубликованных в *Интернет*. На такой *сервер* можно зайти из *Интернет*. Мы же выберем *внутреннее пространство имен*, чтобы из *Интернет* доступа не было, и назовем имя домена, например, **DOMAIN.LAN** - рис. 11.6. Так мы повышаем *безопасность* нашей системы. В этом случае через точку можно писать что угодно.

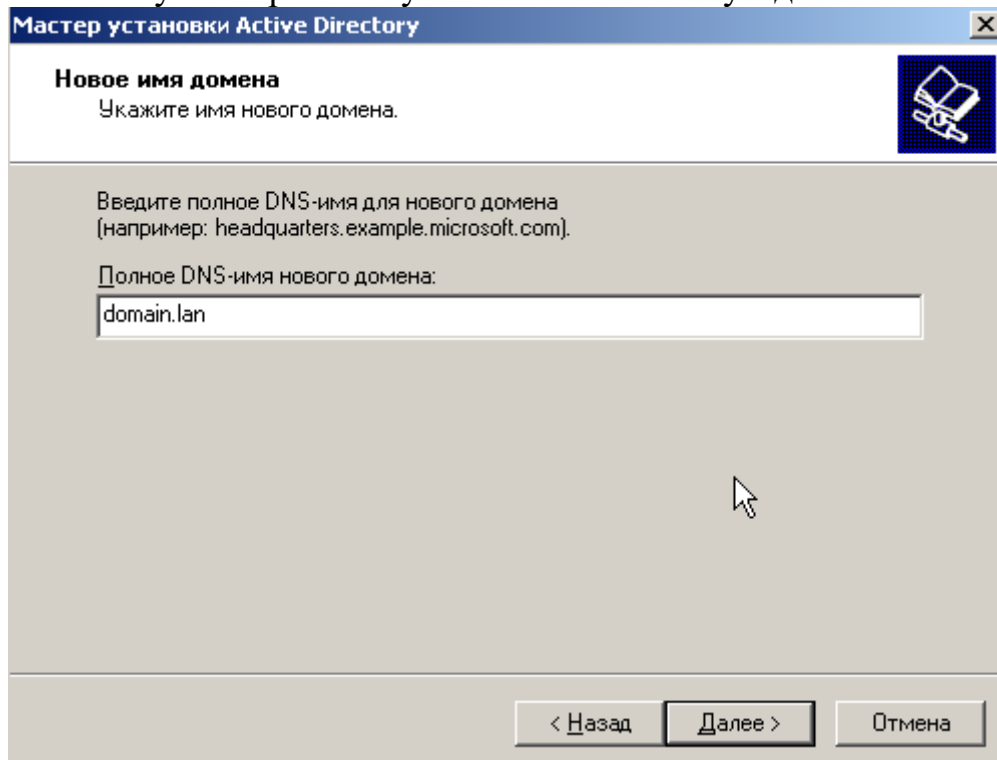


Рис. 11.6. Вводим полное DNS-имя для нового домена

Далее производим несколько шагов с настройками по умолчанию. В следующем окне установим нижний *переключатель*, поскольку о *DNS* мы поговорим позднее (рис. 11.7).

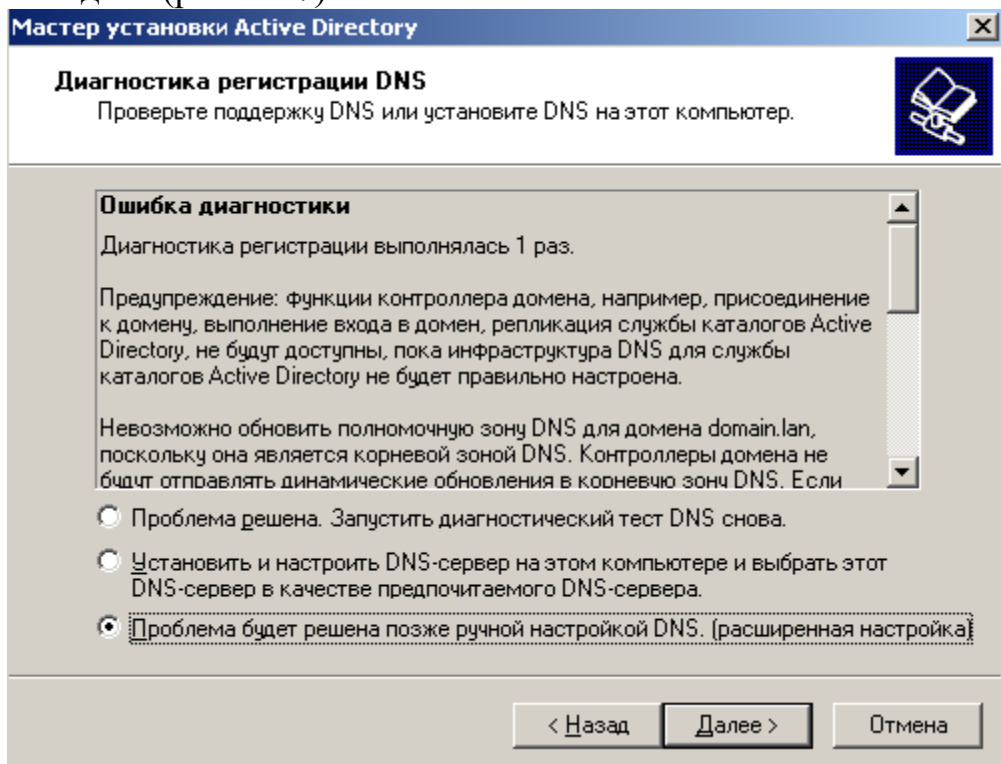


Рис. 11.7. Диагностика и регистрация DNS

Далее выбираем **Разрешения, совместимые только с Windows 2000 или Windows Server 2003** – рис. 11.8.

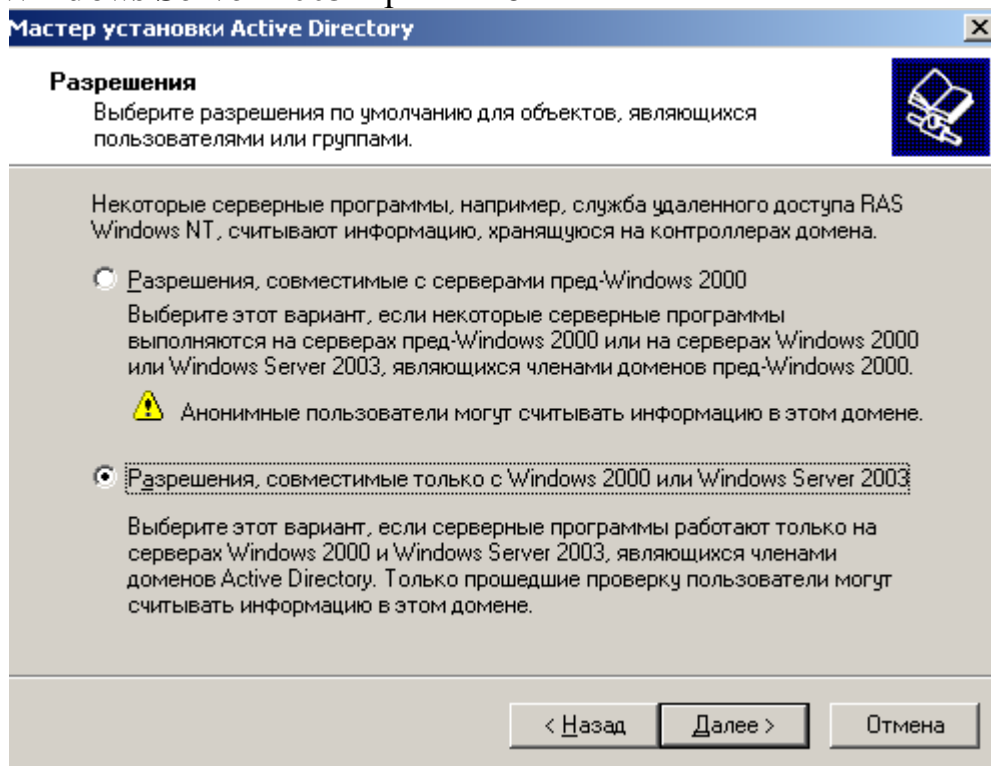


Рис. 11.8. Выбираем разрешения

Теперь вводим *пароль* администратора в режиме восстановления, например, тот же, что был при входе в систему - 123456, (рис. 11.9). Понятно, что простой или пустой *пароль* допустимы только в учебных целях.

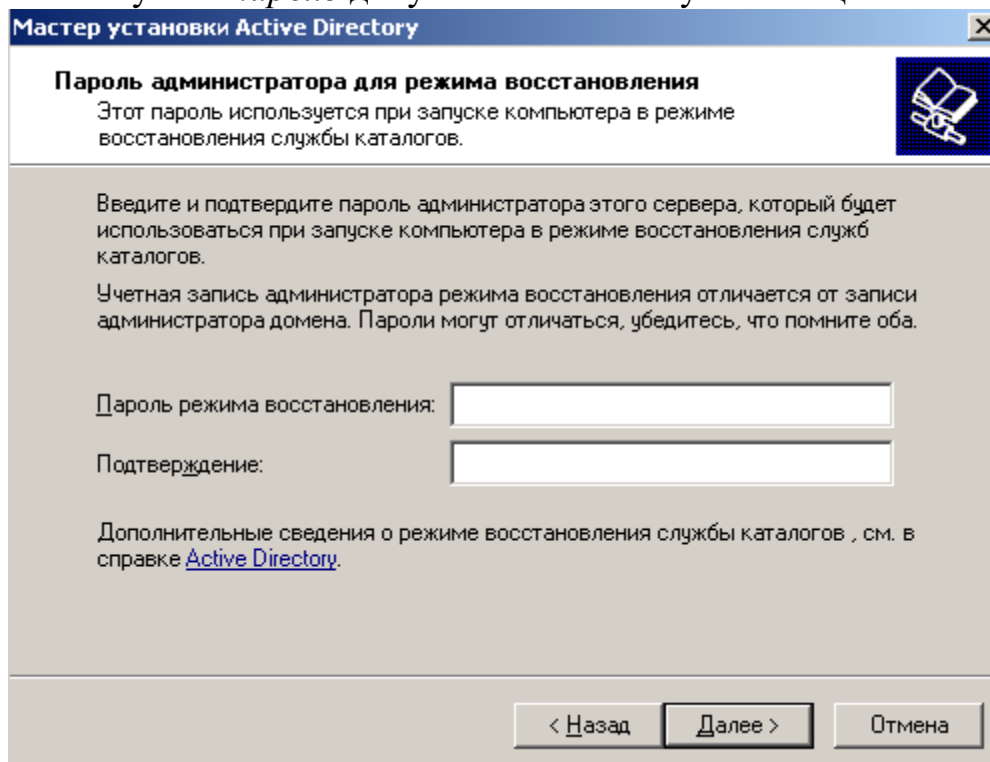


Рис. 11.9. Задаем пароль режима восстановления

Далее ждем, пока произойдет настройка **Active Directory (Активный каталог)** - рис. 11.10. На этом этапе может понадобиться установочный диск SP2.

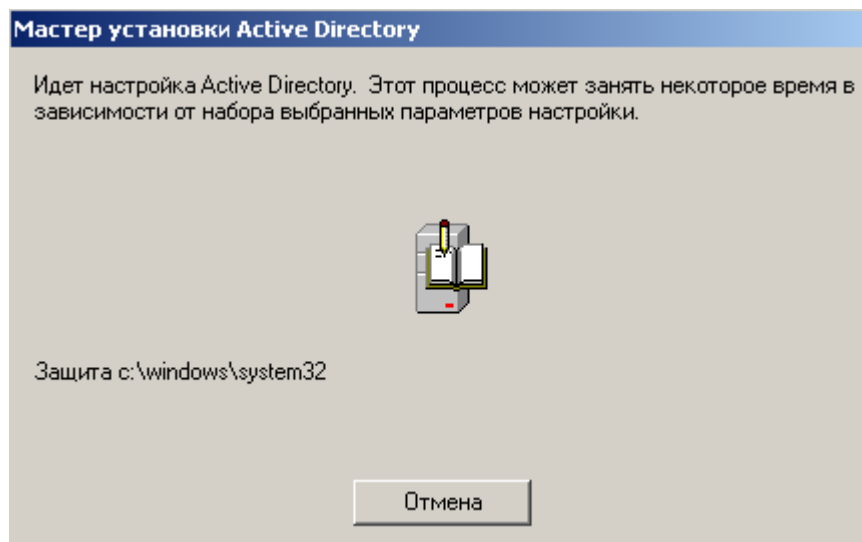


Рис. 11.10. Идет конфигурирование Active Directory

Далее Мастер успешно завершит свою работу следующим окном (рис. 11.11).

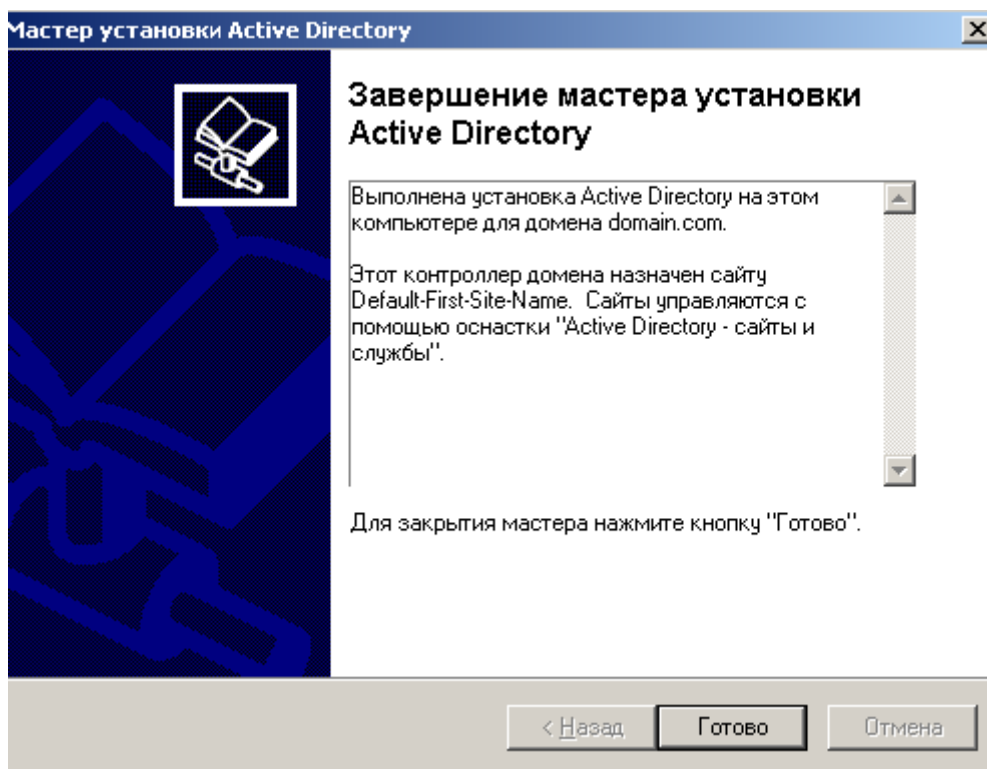


Рис. 11.11. AD установлена

Работа почти закончена: для домена (*domain.lan*) *active directory* установлена, а настройку *DNS* для этого домена мы сделаем позже.

Примечание

Настроить *DNS*-сервер означает привязать доменное имя **domain.lan** к IP адресу компьютера, на котором находится ваш сервер.

После перезагрузки мы увидим, что окно входа в систему изменилось (рис. 11.12).

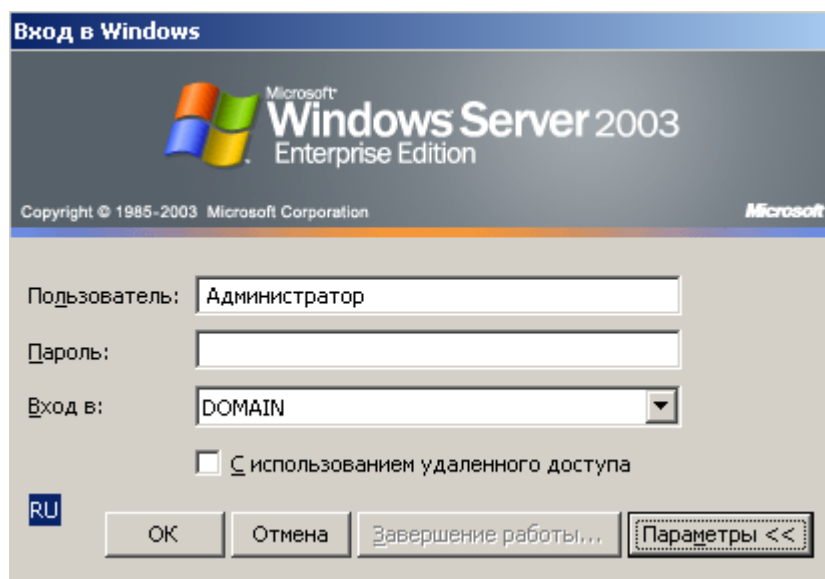


Рис. 11.12. В окне входа в систему появилась строка входа в домен
Далее увидим следующее сообщение (рис. 11.13).

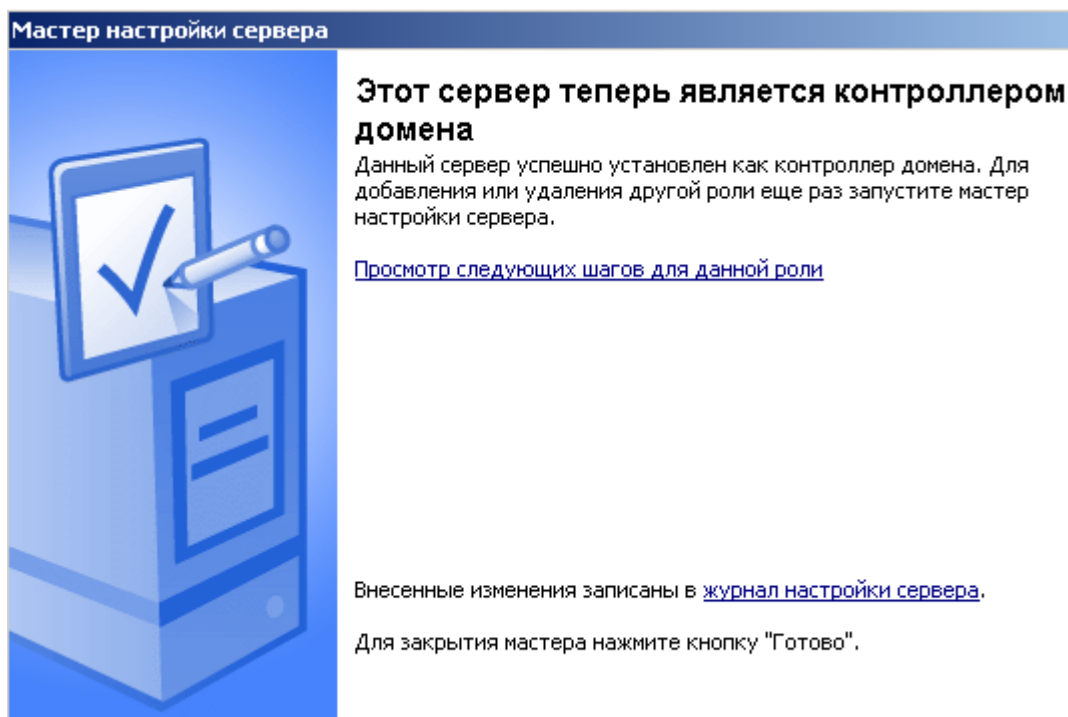


Рис. 11.13. Настройка роли сервера Контроллер домена завершена
Смотрим, что изменилось

Выполним **Пуск-Все программы-Администрирование**. Вы увидите, что у нас появилось пять новых служб (рис. 11.14).

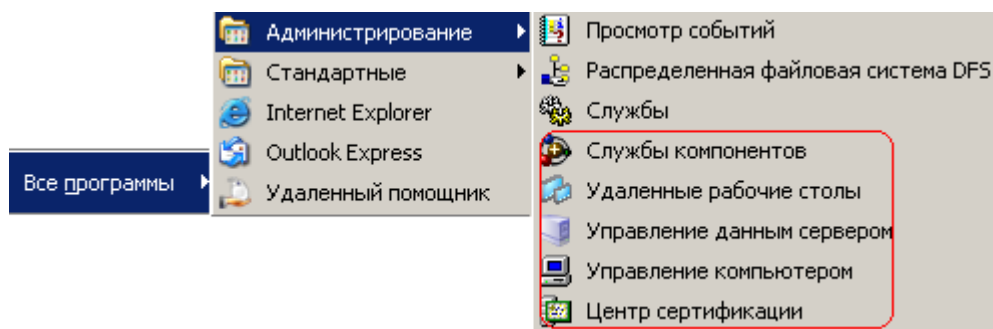


Рис. 11.14. Красным отмечены новые службы

Войдем в службы компонентов и увидим там наш *домен* (рис. 11.15). Доменное имя здесь можно было создать по имени вашей организации, например, *NOVGU.LAN* или *GORGAZ.LAN*.

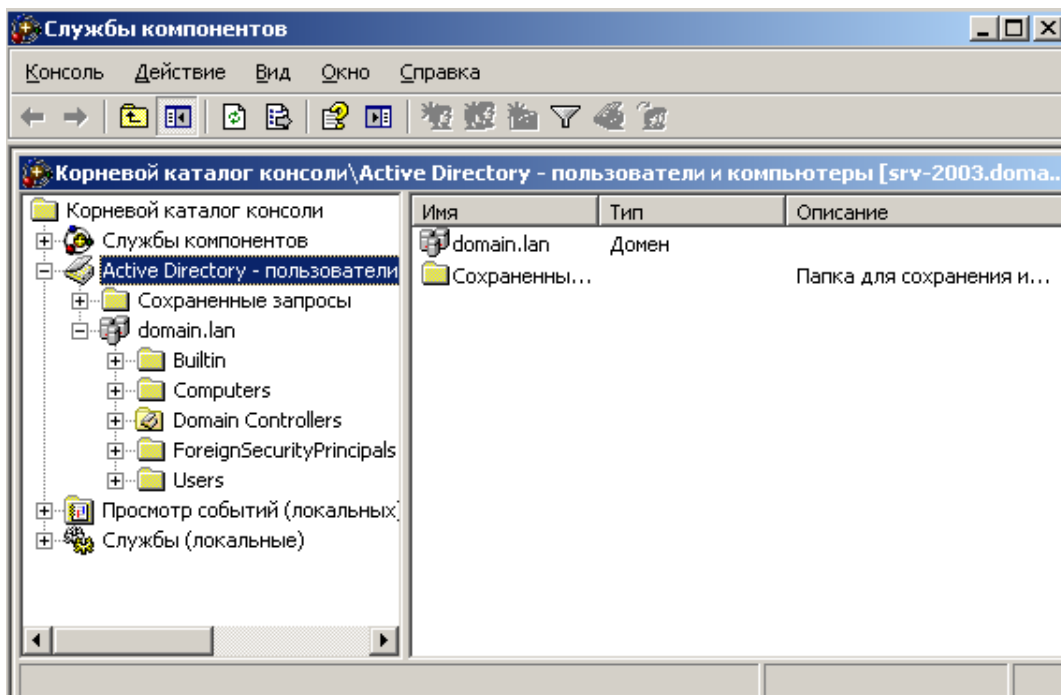


Рис. 11.15. В AD мы видим domain.lan

Пока в домене нет компьютеров, но есть пользователи. Дважды щелкнув на учетную запись **Администратор** мы можем задать характеристики данного пользователя (рис. 11.16).

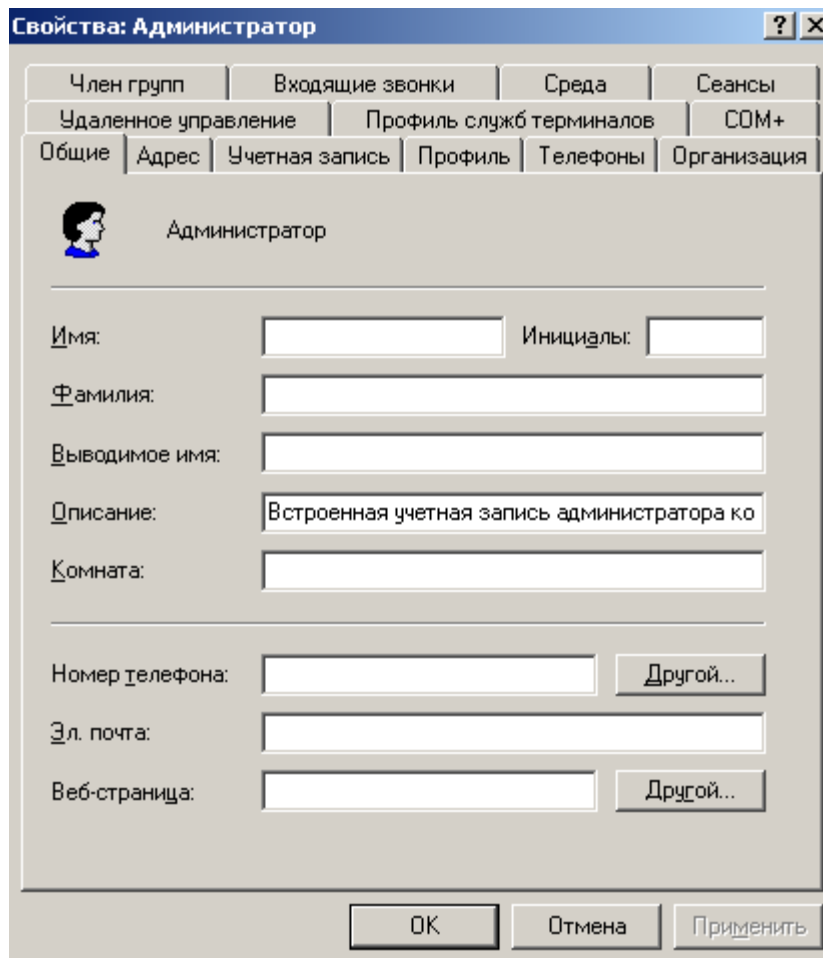


Рис. 11.16. Окно свойств записи Администратор

Таким образом, номер телефона администратора сети или его почту можно, затем найти из кнопки **Пуск-Поиск-Другие параметры поиска-Принтеры, компьютеры или людей** (рис. 11.17). Подобным образом заводится информация не только на администратора, но и на других пользователей AD.

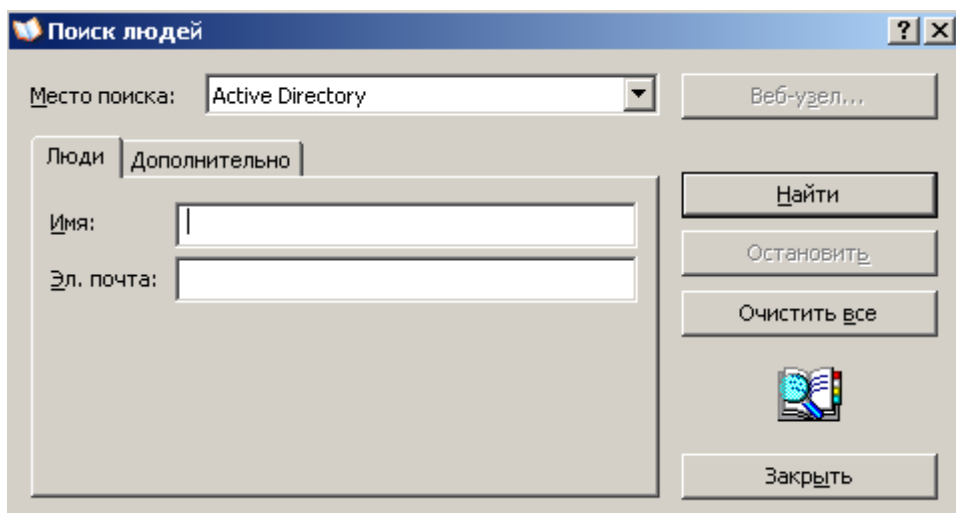


Рис. 11.17. Окно поиска людей в AD

Дополнительный материал – поясняем термины по теме

Ниже мы поясним несколько новых терминов по теме лекции.

Домен, дерево доменов, лес и др.

В сети Клиент-Сервер компьютеры могут объединяться в логические единицы, называемые доменами. В одноранговой сети аналогом домена является *рабочая группа*. Множество доменов образует структуру, похожую на *дерево*.

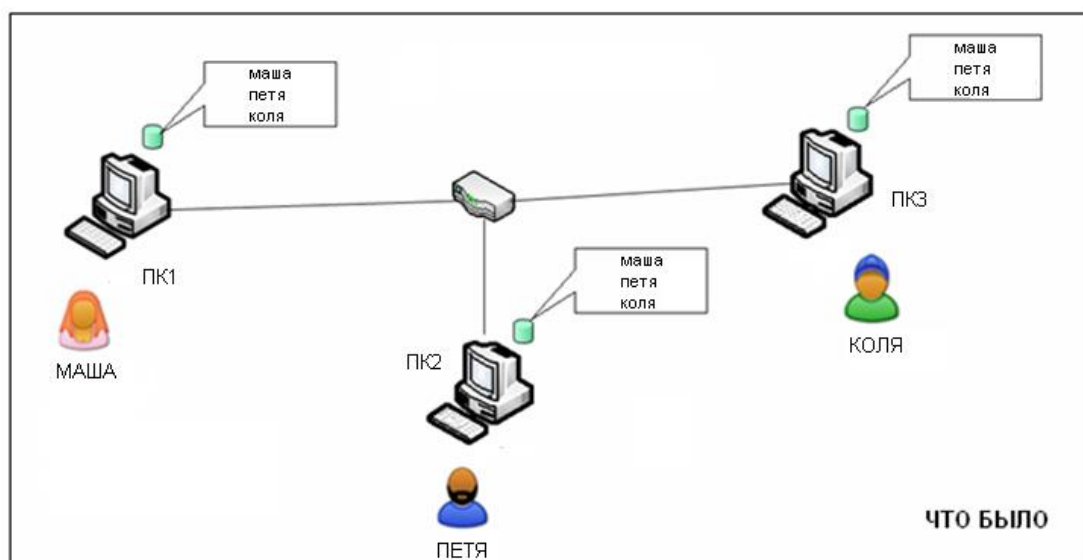
Каждый домен управляется контроллером домена. Компьютер может входить в состав только одного домена, а доменов может быть несколько. При объединении доменов в *лес* их конфигурация становится одинаковой (рис. 11.18).



Рис. 11.18. Пример по аналогии из нашей жизни, дающий представление о лесе доменов

Active Directory (AD-Активный каталог)

AD – справочник о всех объектах сети (пользователях, их паролях *etc.*). Это база данных, содержащая информацию о ресурсах, службах и учетных записях. Для простоты понимания, вы можете представить себе телефонный справочник, содержащий информацию о людях, их телефонах и адресах. На рис. 11.19 показано как было в локальной сети до использования сервера и Active Directory (Активный каталог), и как стало после.



а) Ситуация в сети “До” создания AD

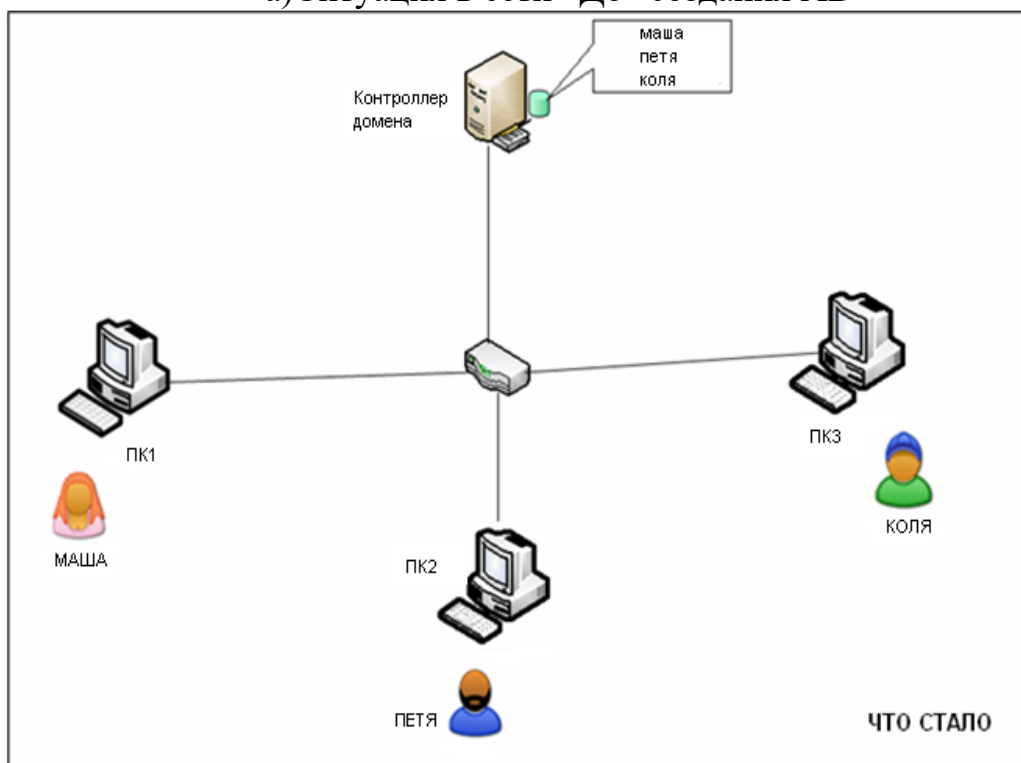


Рис. 11.19. б) Ситуация в сети “После” создания AD

Как видим из рисунка, "До" – учетные записи пользователя и их ресурсы хранились на локальных ПК. "После" – на сервере. Таким образом, одно из достоинств AD заключается в том, что с появлением контроллера домена учетные записи хранятся не на локальных ПК, а на сервере. Поэтому, при поломке одного из ПК *пользователь* может авторизоваться на любом из локальных ПК и работать, используя ресурсы сервера.

Задание 1. Сколько в лесу (рис. 11.20) деревьев и доменов?

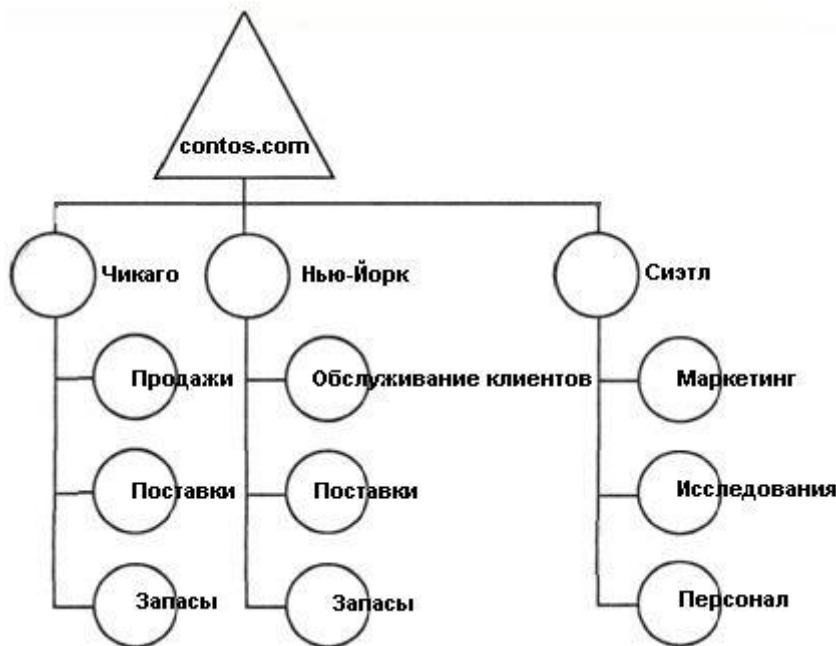


Рис. 11.20. Пример леса доменов

В качестве примера-подсказки на рис. 11.21 показан лес из двух деревьев:

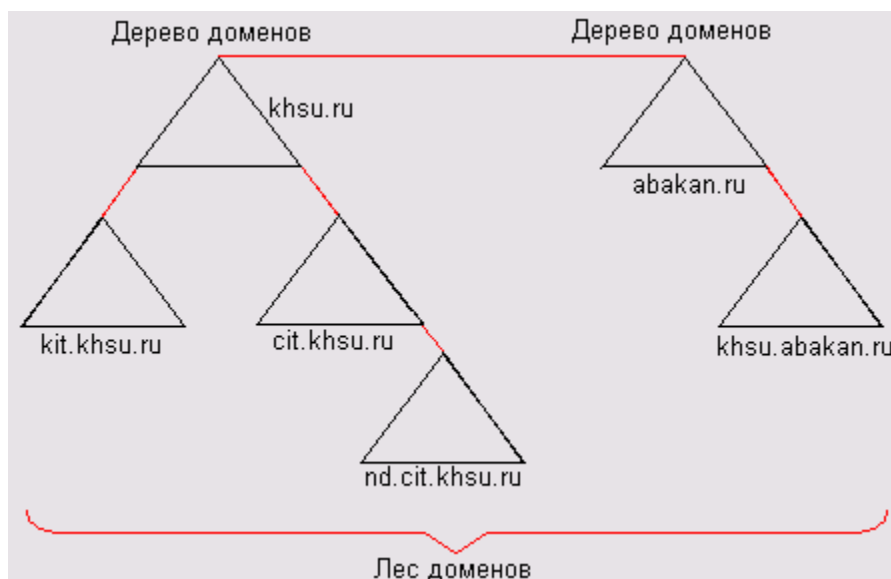


Рис. 11.21. Один лес, который содержит два дерева доменов

Краткие итоги

В этой работе мы научились устанавливать AD и создали новый домен в новом лесу скринкаст. Познакомились с рядом новых терминов

(домен, дерево доменов, лес, Active Directory) и выполнили практическое задание. Лабораторную работу дополняет скринкаст.

Лабораторная работа №10

Установка DNS сервера

Выбор для сервера роли DNS сервера

DNS (*Domain Name System* — система доменных имён) — компьютерная система для получения IP-адреса по имени хоста и обратно.

Назначим нашему серверу роль DNS сервера (рис. 12.1).

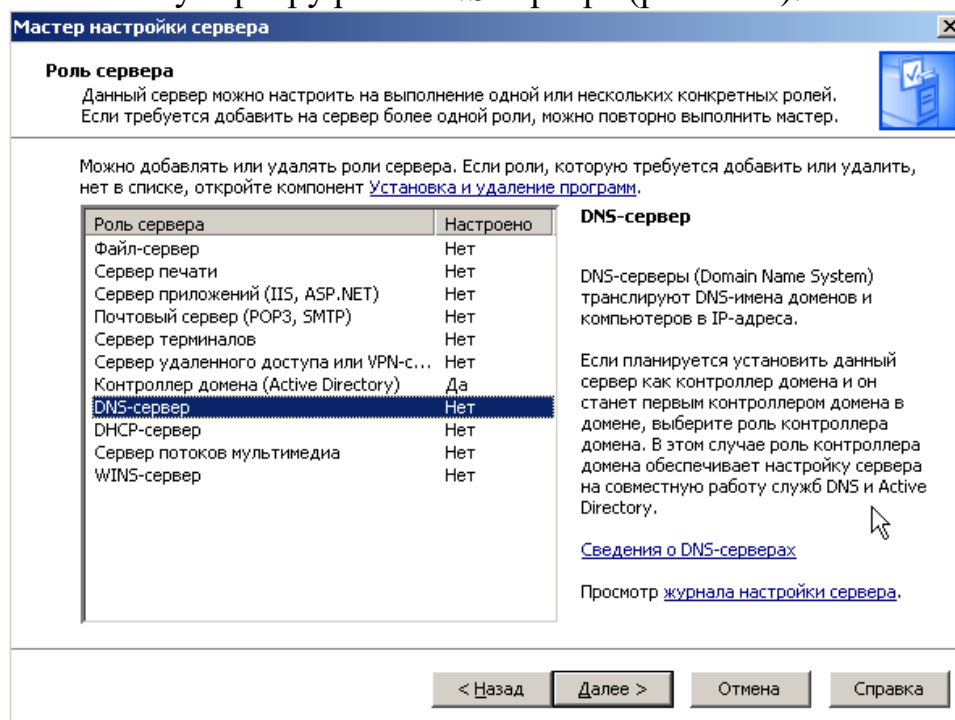


Рис. 12.1. Мастер настройки сервера-DNS сервер

Далее появится предложение изменить динамический IP адрес на статический (рис. 12.2).

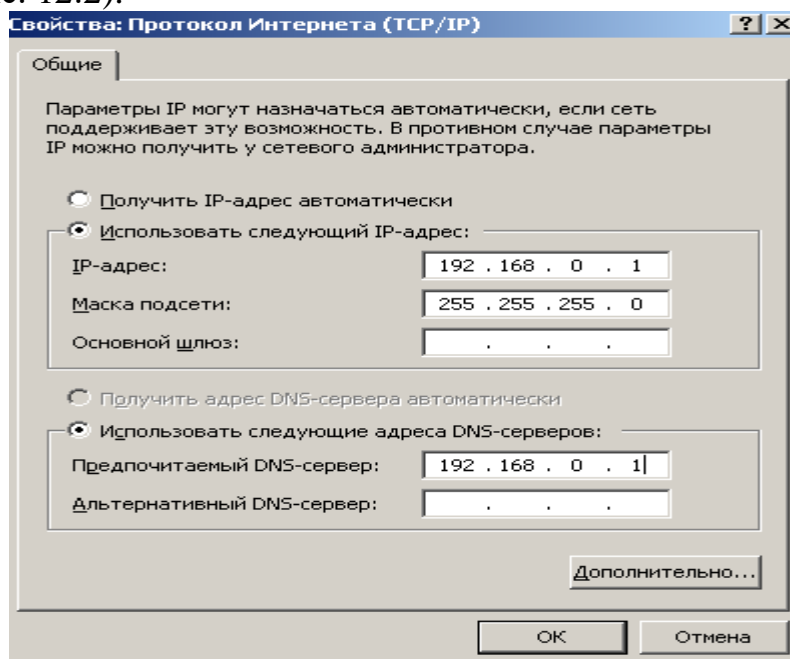


Рис. 12.2. Задаем серверу статический IP адрес

Далее появится Мастер настройки DNS сервера (рис. 12.3).

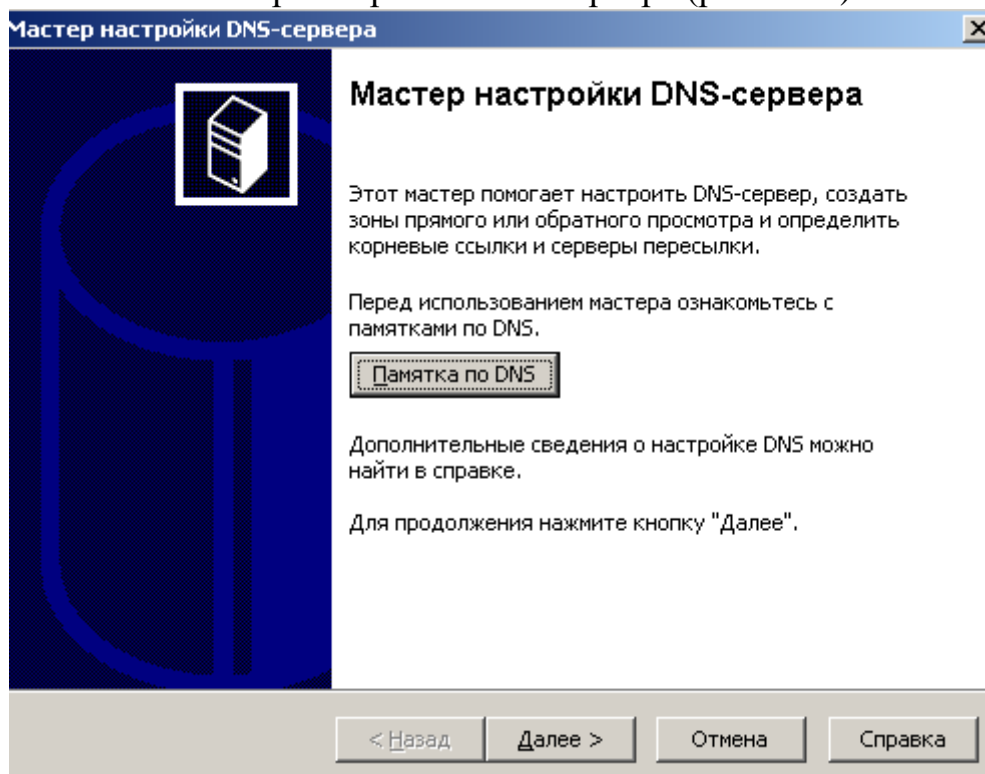


Рис. 12.3. Окно Мастер настройки DNS сервера

Поскольку сеть у нас небольшая, то установим верхний переключатель (рис. 12.4).

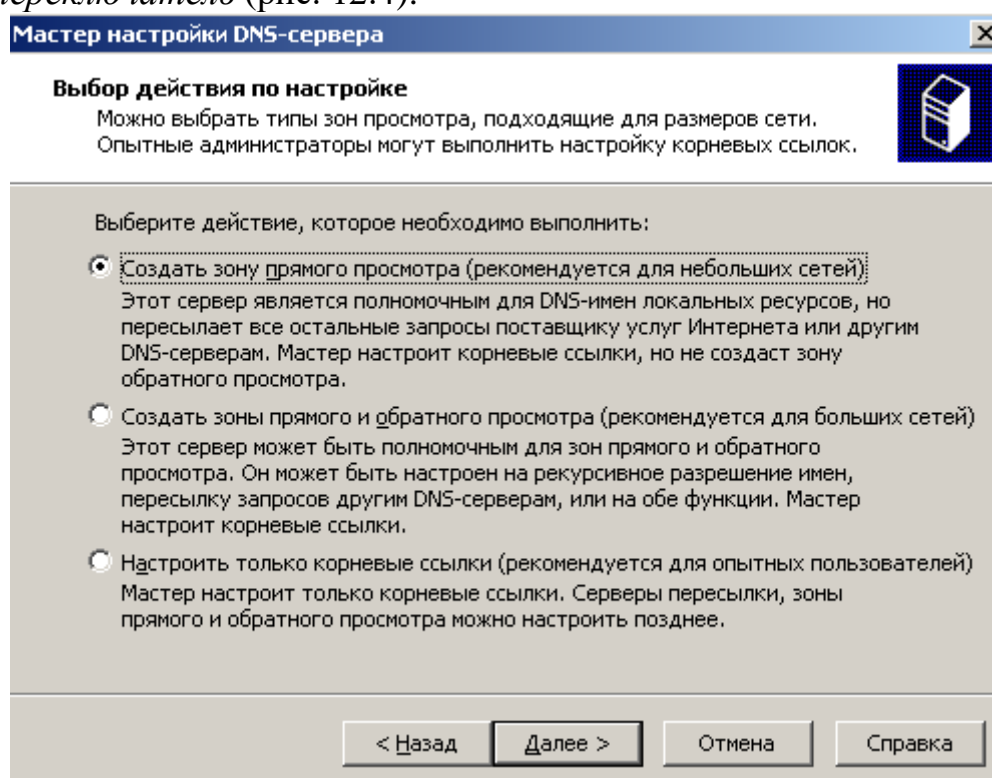


Рис. 12.4. Создание зоны прямого просмотра

Зону прямого просмотра назовем так же, как и домен – *domain.110*. Далее исходим из того, что у нас только один *DNS сервер*, больше пересылать запросы некому (рис. 12.5).

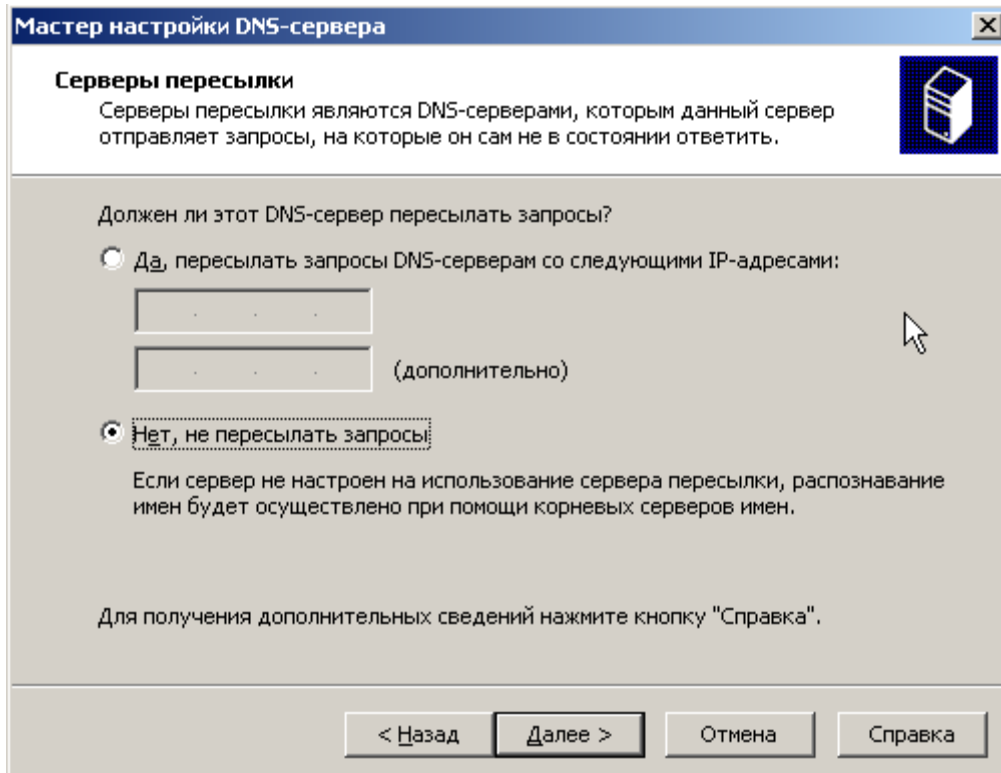


Рис. 12.5. Активируем нижний переключатель
Настройка сервера завершена (рис. 12.6).

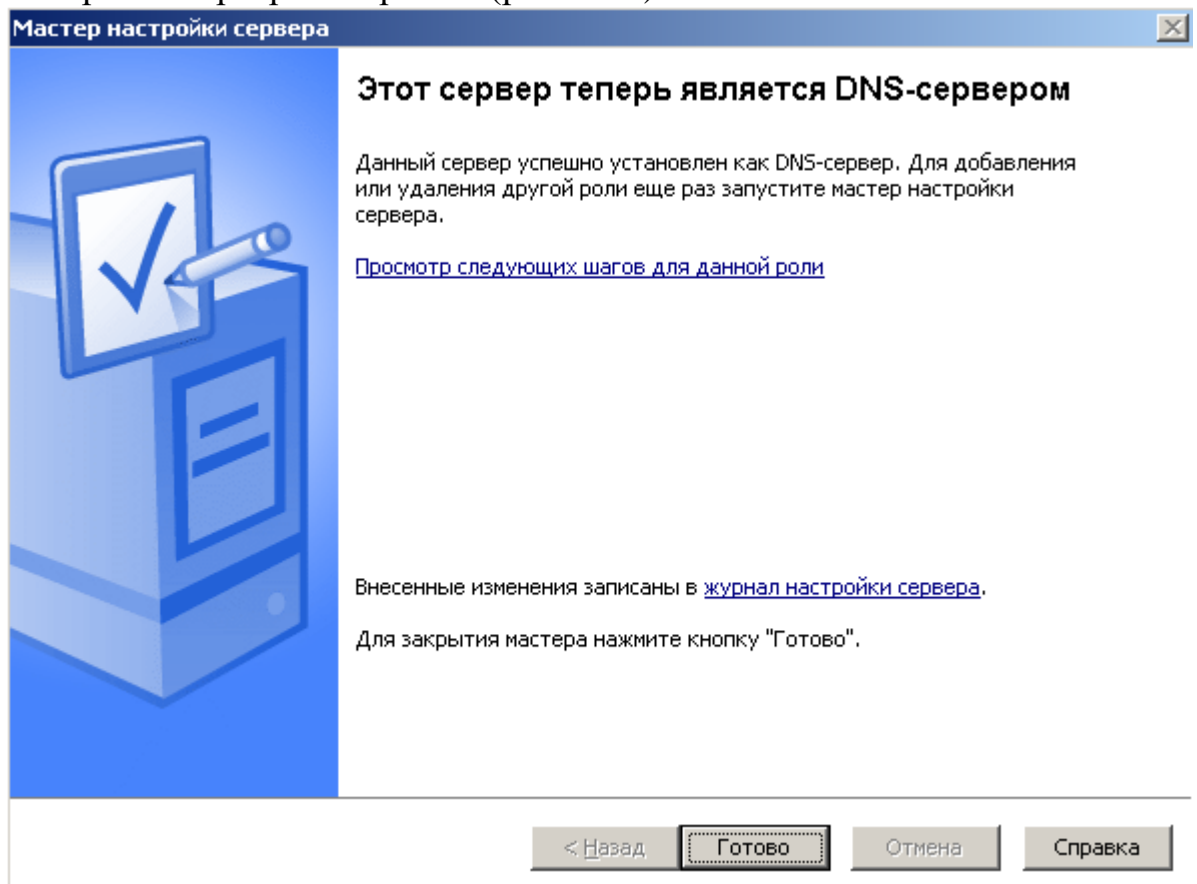


Рис. 12.6. Сервер получил роль DNS сервера

Выполнив команду, **Пуск-Все программы-Администрирование** мы увидим, что появилась новая оснастка (рис. 12.7).

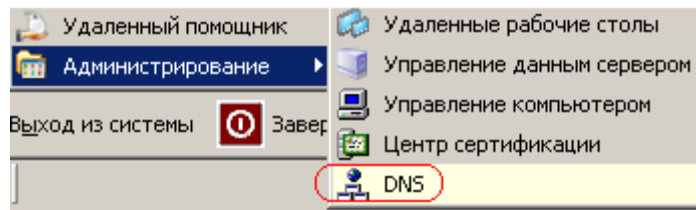


Рис. 12.7. На рисунке новая оснастка отмечена красным

Откроем ее (рис. 12.8). Здесь в зоне прямого просмотра вы можете увидеть соответствие имени сервера `srv-2003` его IP адресу `192.168.0.1`.

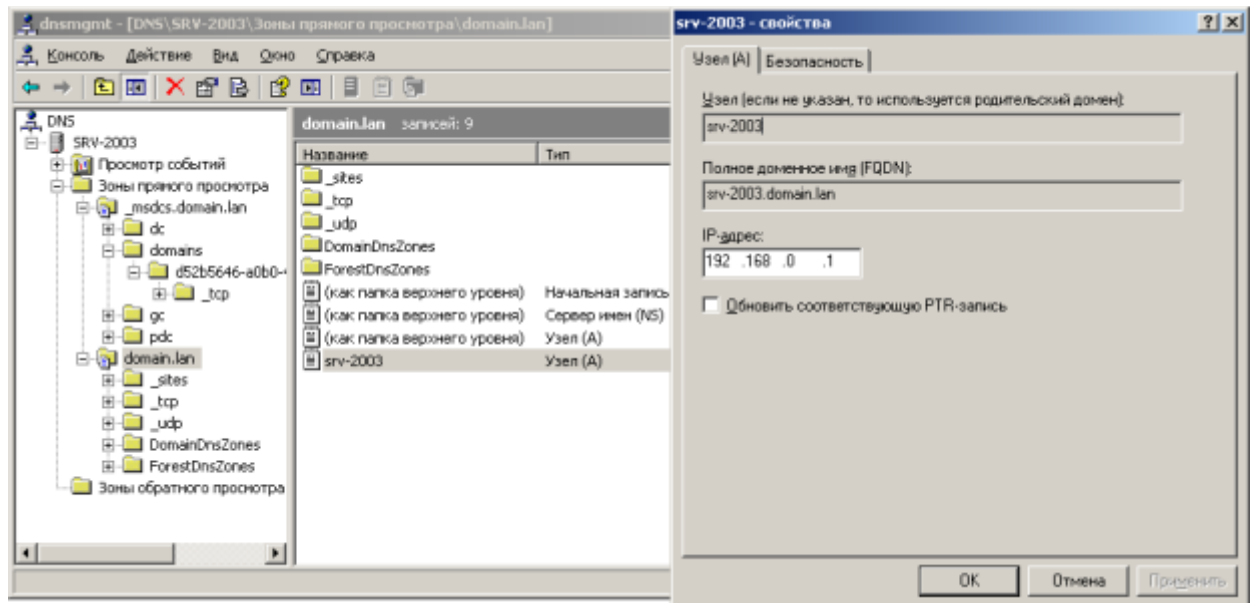


Рис. 12.8. На рисунке открыта зона прямого просмотра

Создание зоны обратного просмотра

Щелкните на строчку **Зона обратного просмотра** и выберите команду **Создать новую зону** (рис. 12.9).

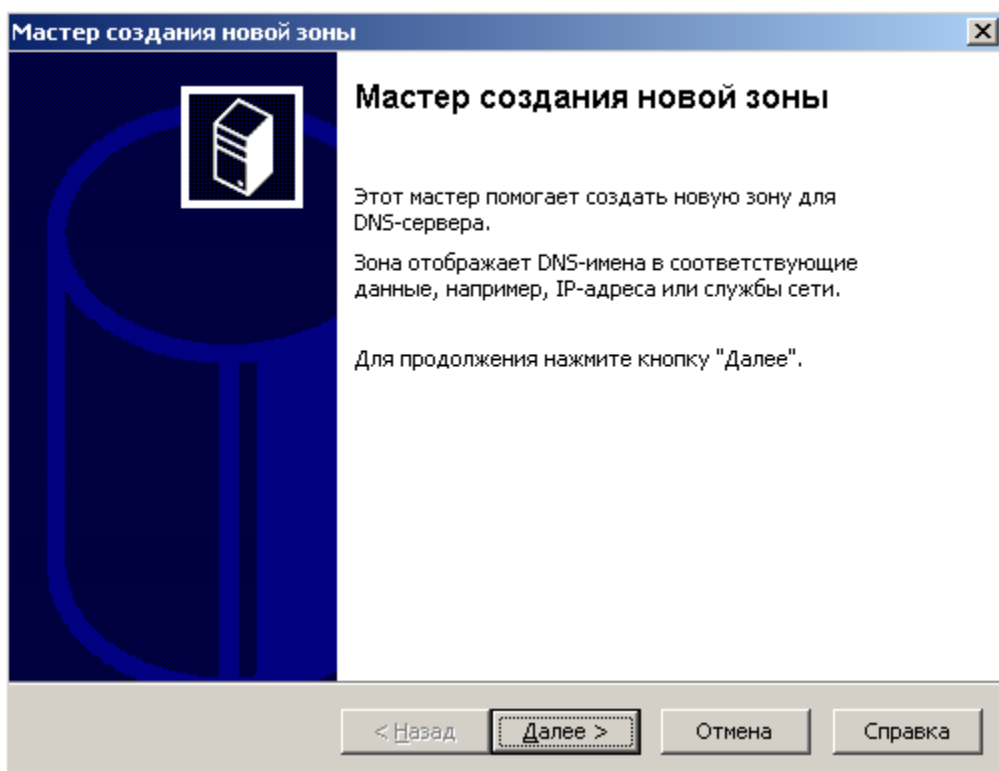


Рис. 12.9. Окно мастера создания новой зоны
Далее устанавливаем верхний *переключатель* - рис. 12.12.

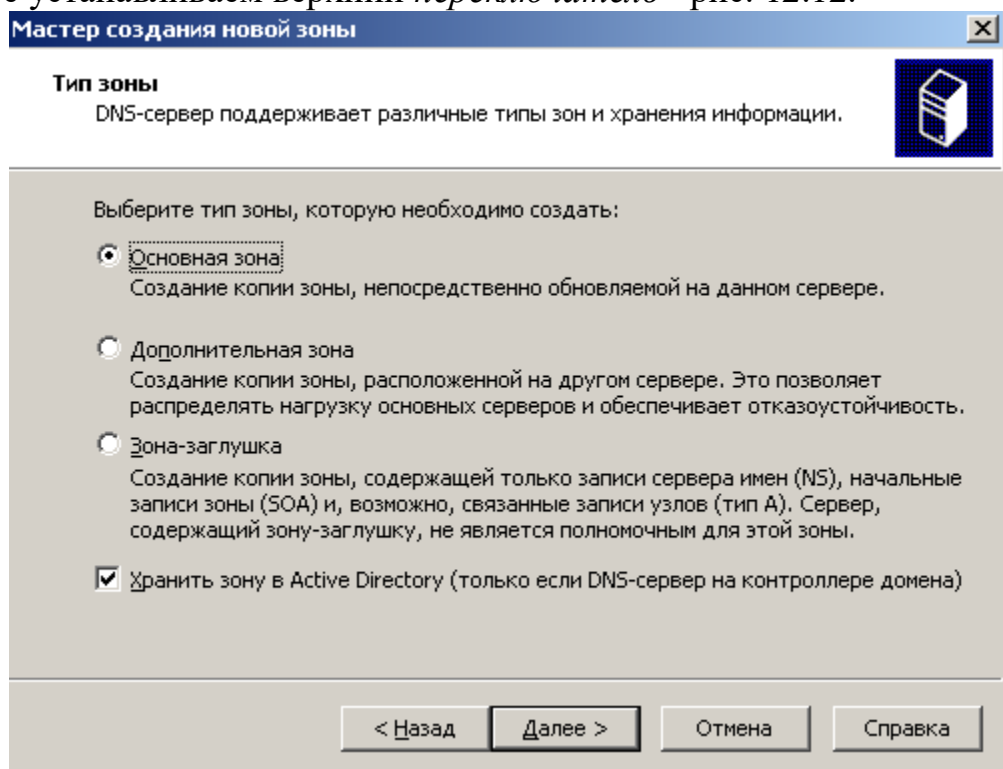


Рис. 12.10. Устанавливаем переключатель Основная зона

Примечание

Основная зона устанавливается на основной сервер (она - главная), **Дополнительная зона** необходима для резервирования и разгрузки основного сервера. Если загрузка первого DNS сервера велика (или он отключился), то часть запросов можно отправить на второй, альтернативный

DNS и отказоустойчивость системы повышается (рис. 12.11). Зона - заглушка содержит IP адрес сервера, который может обслужить запрос.

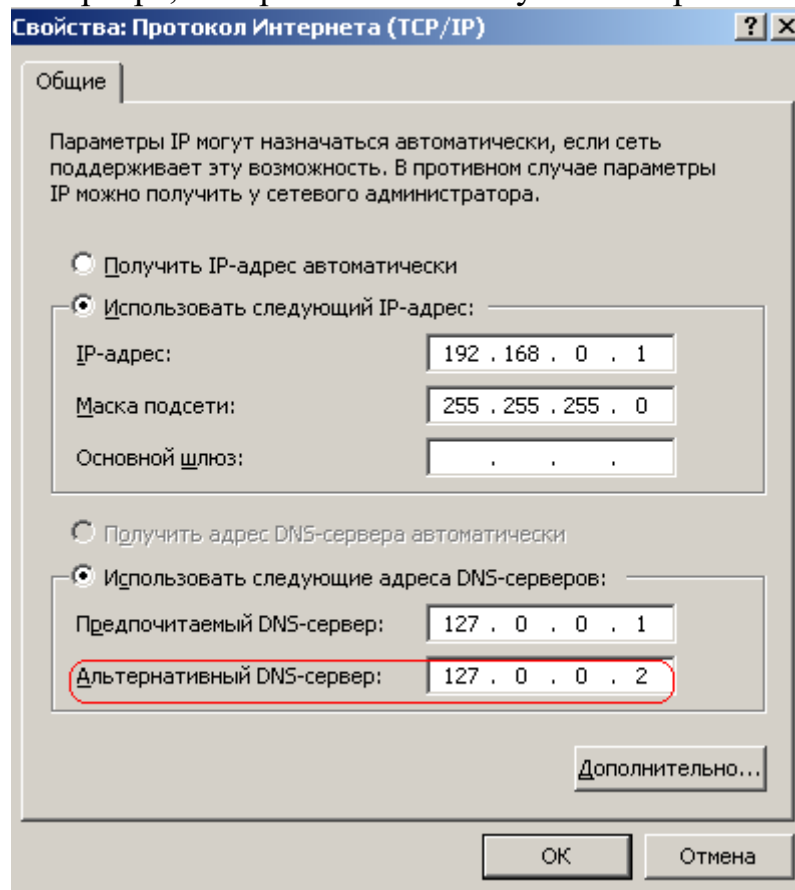


Рис. 12.11. Пример использования альтернативного DNS сервера
Наша сеть 192.168.0.1 относится к классу сетей C, поэтому значение 192.168.0 мы менять не можем - это и есть код сети (ID) – рис. 12.12.

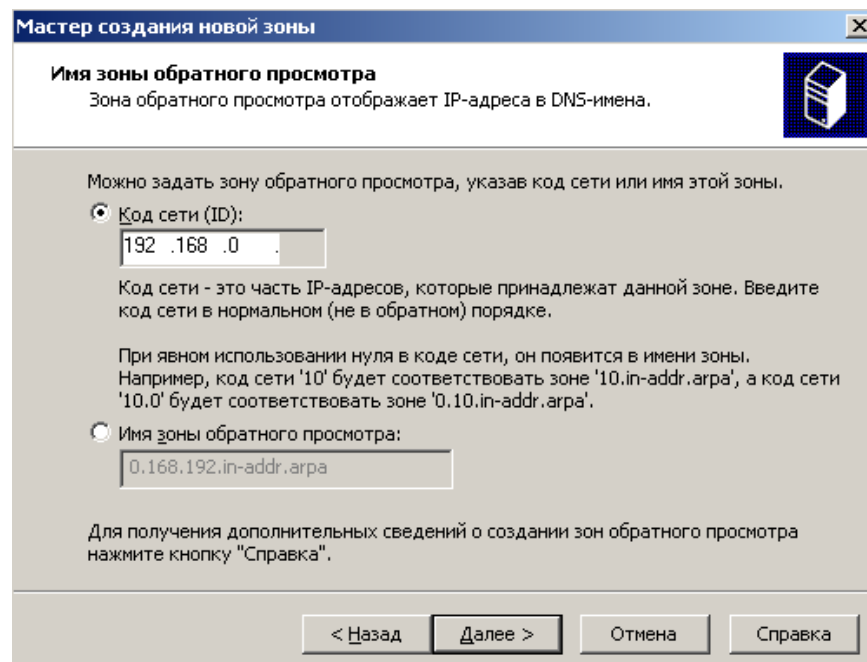


Рис. 12.12. Задаем код сети

Зона обратного просмотра создана. После подключения первого ПК здесь появится соответствие *IP* адреса ПК его имени.

Записи ресурсов DNS

Вся *DNS* состоит из этих RR записей, по которым *пользователь* может искать ресурсы в сети. Запустим *консоль* управления *DNS* и зайдем в зону прямого просмотра (рис. 12.13).

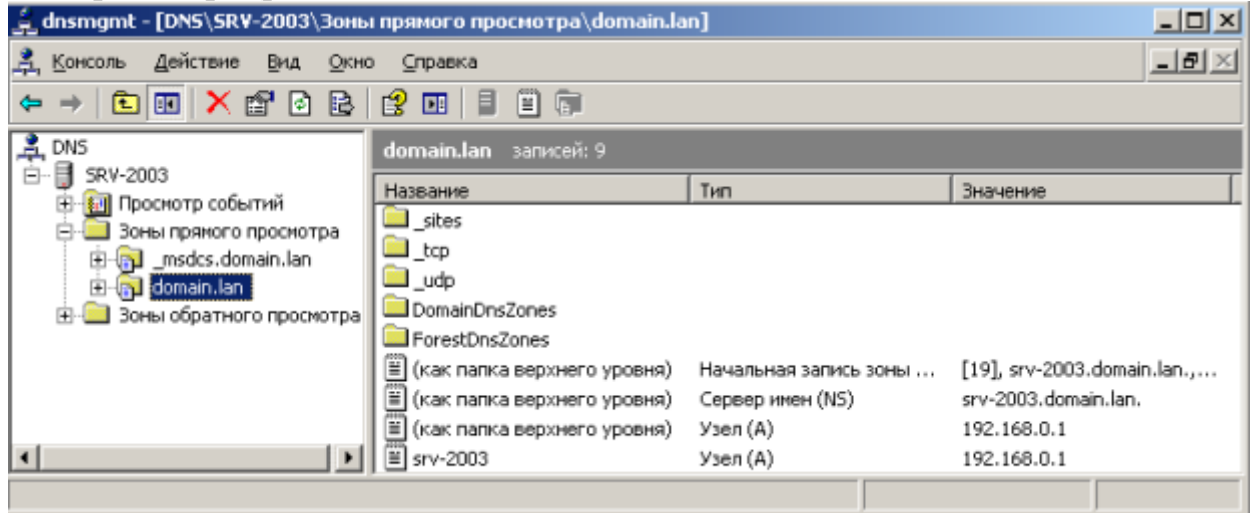


Рис. 12.13. Зоны прямого просмотра

Выполним *двойной щелчок* на строчке **Начальная запись зоны** (рис. 12.14). В данном окне наиболее интересный *параметр* **Срок жизни (TTL)**. Предположим, что *компьютер* с именем ПК-1 и *IP* адресом 192.168.0.1 хочет соединиться с именем ПК-2 и *IP* адресом 192.168.0.2. Он выдает *запрос* на *DNS сервер* и тот сообщает, что с *компьютер* с именем ПК-2 имеет *IP адрес* 192.168.0.2. Эта *запись* кэшируется (помещается в *память*) на **Срок жизни (TTL)**. Подобный подход к запросам снижает нагрузку на *DNS сервер*.

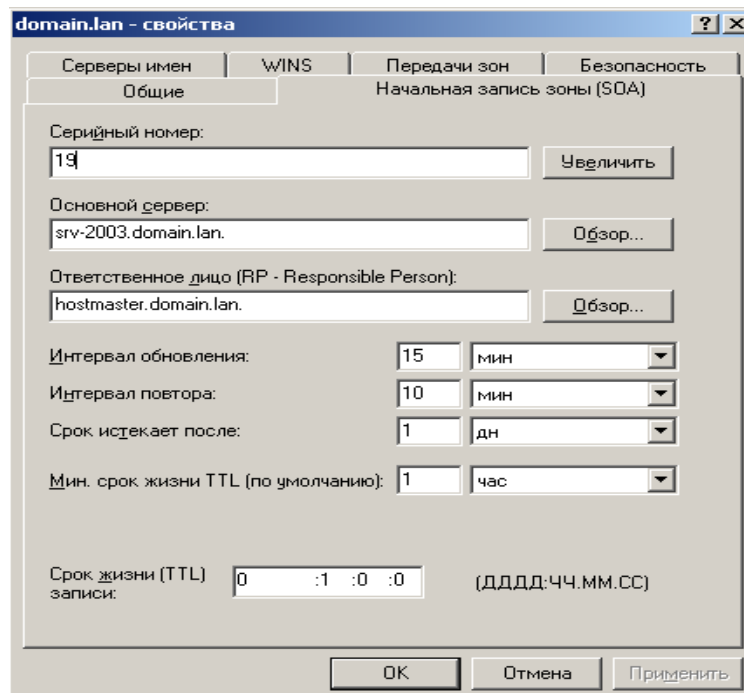


Рис. 12.14. Вкладка Начальная запись зоны

Теперь выполним *двойной щелчок* на строчке **Сервер имен** (рис. 12.15). У нас *DNS сервер* один, поэтому *запись* здесь одна. Но, здесь записей может быть столько, сколько имеется *DNS серверов*.

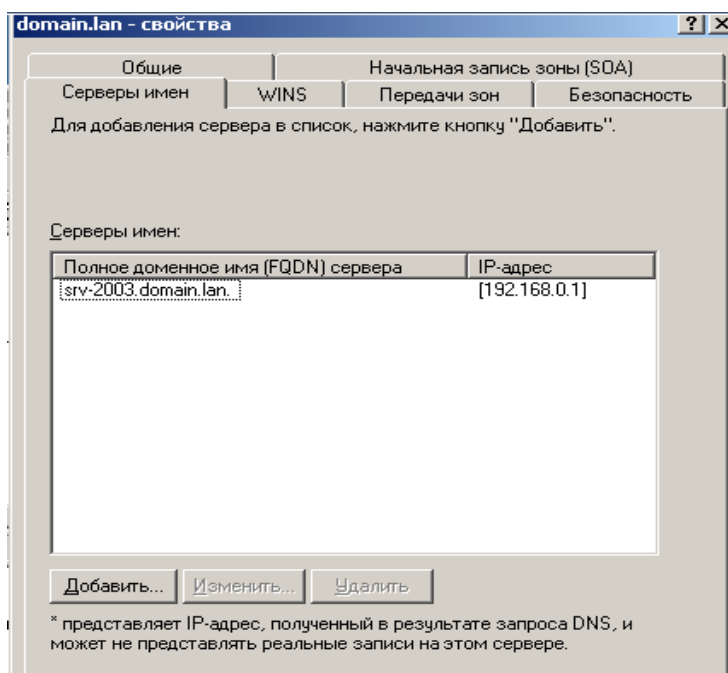


Рис. 12.15. Вкладка Серверы имен

Выполним *двойной щелчок* на строчке **Узел А** (рис. 12.16). Эта *запись* устанавливает прямое соответствие между именем ПК и его *IP* адресом.

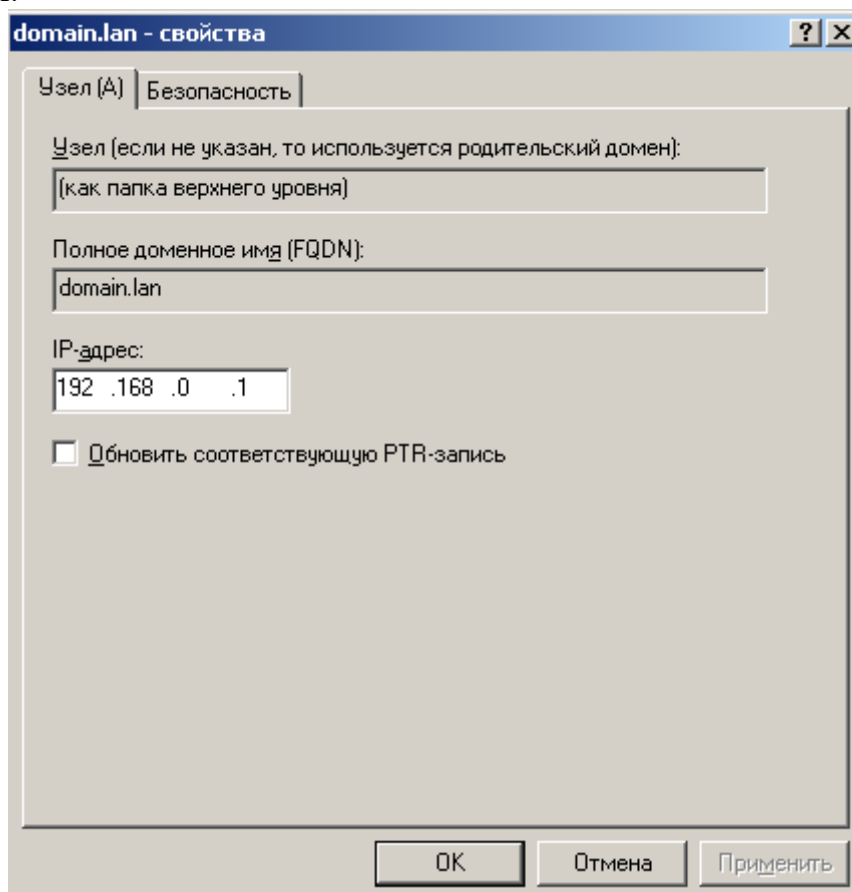


Рис. 12.16. Вкладка Узел А

Теперь изучим записи зоны обратного просмотра. После перезагрузки ПК здесь будет три записи (рис. 12.17).

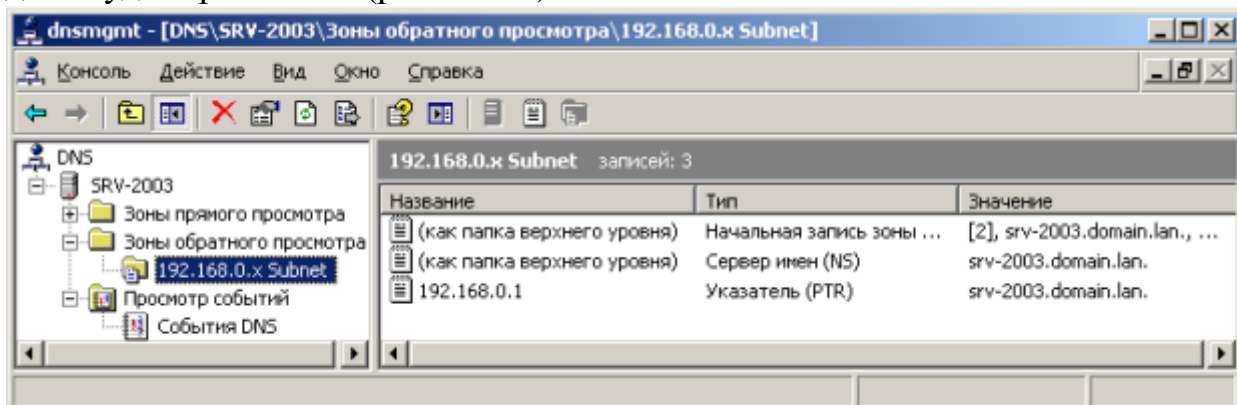


Рис. 12.17. Записи зоны обратного просмотра

Двойным щелчком мыши зайдем в **Указатель (PTR)** – рис. 12.18. Как видим, именно здесь задается обратное соответствие IP адреса и имени ПК.

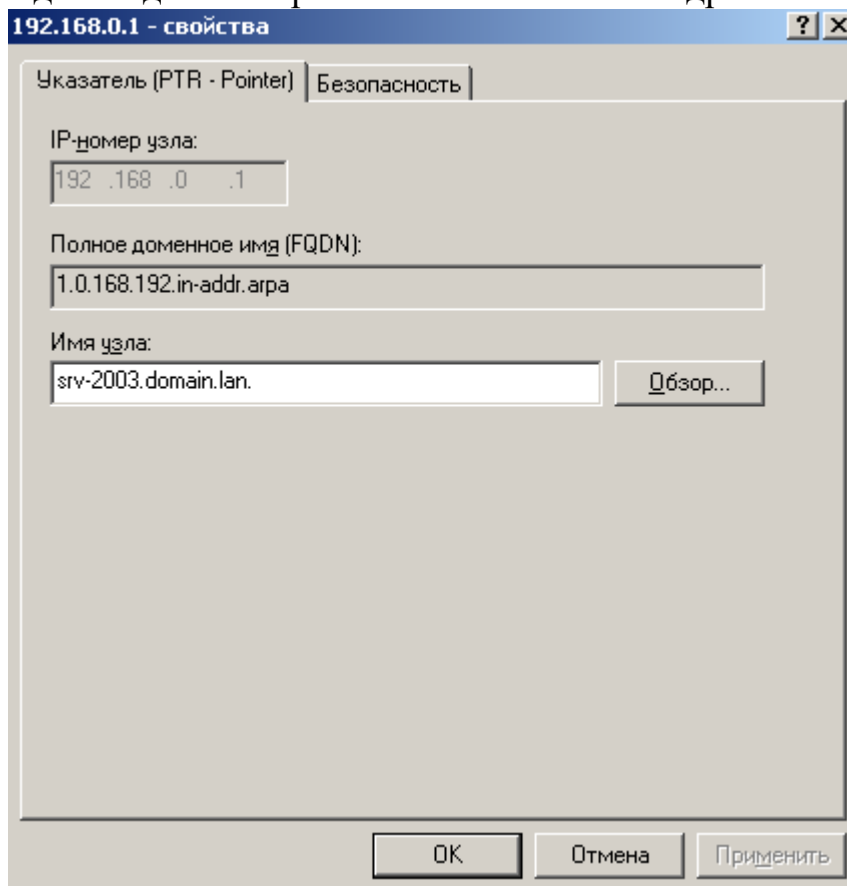


Рис. 12.18. Вкладка Указатель (PTR)

Проверка работы зон прямого и обратного просмотра

Далее вызовем командную строку и пропингуем наш сервер (рис. 12.19). Видим, что зона прямого просмотра работает нормально и имени SRV-2003 ставится в соответствие IP адрес 192.168.0.1.

```
С:\ Командная строка
Microsoft Windows [Версия 5.2.3790]
(C) Корпорация Майкрософт, 1985-2003.

C:\Documents and Settings\Администратор>ping srv-2003

Обмен пакетами с srv-2003.domain.lan [192.168.0.1] с 32 байт данных:

Ответ от 192.168.0.1: число байт=32 время=15мс TTL=128
Ответ от 192.168.0.1: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.1: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.1: число байт=32 время<1мс TTL=128

Статистика Ping для 192.168.0.1:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (<0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 15 мсек, Среднее = 3 мсек

C:\Documents and Settings\Администратор>_
```

Рис. 12.19. Окно Командная строка

Если пропинговать не имя, а *IP адрес* и использовать ключ "-а", то увидим, что зона обратного просмотра также работает хорошо (рис. 12.20). *Обмен данными* идет нормально.

```
С:\ Командная строка

C:\Documents and Settings\Администратор>ping -a 192.168.0.1

Обмен пакетами с srv-2003.domain.lan [192.168.0.1] с 32 байт данных:

Ответ от 192.168.0.1: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.1: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.1: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.1: число байт=32 время<1мс TTL=128

Статистика Ping для 192.168.0.1:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (<0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 0 мсек, Среднее = 0 мсек

C:\Documents and Settings\Администратор>
```

Рис. 12.20. Зона обратного просмотра работает хорошо

Краткие итоги

В этой работе мы произвели выбор для сервера роли *DNS* сервера, создали зоны прямого и обратного просмотра, а также проверили *корректность* работы этих зон. К работе прилагается скринкаст.

Лабораторная работа №11

Построение диаграмм сети

Краткие теоретические сведения

Программа построения диаграмм сети EDraw Network Diagrammer

При проектировании сетей иногда используется EDraw *Network Diagrammer* – программа создания диаграмм сети с большим количеством примеров и шаблонов.

Основные диаграммы:

Топологические схемы сети

Проектирование сетей Cisco

Диаграммы кабельных сетей

Диаграммы LAN (локальная компьютерная сеть)

Диаграммы сетей WAN (глобальная сеть)

Сетевая диаграмма (граф сети) - графическое *отображение работ* проекта сети и их взаимосвязей. Отличием от блок-схемы является то, что *сетевая диаграмма* моделирует только логические зависимости между элементарными работами. Она не отображает входы, процессы и выходы.

Программа имеет как сходство с программой 10 Страйк: Схема Сети, так и принципиальные отличия. Например, в ней можно нарисовать не только изображение сети (рис.), но и изображение помещения, где эту *сеть* планируется установить (рис. 2).

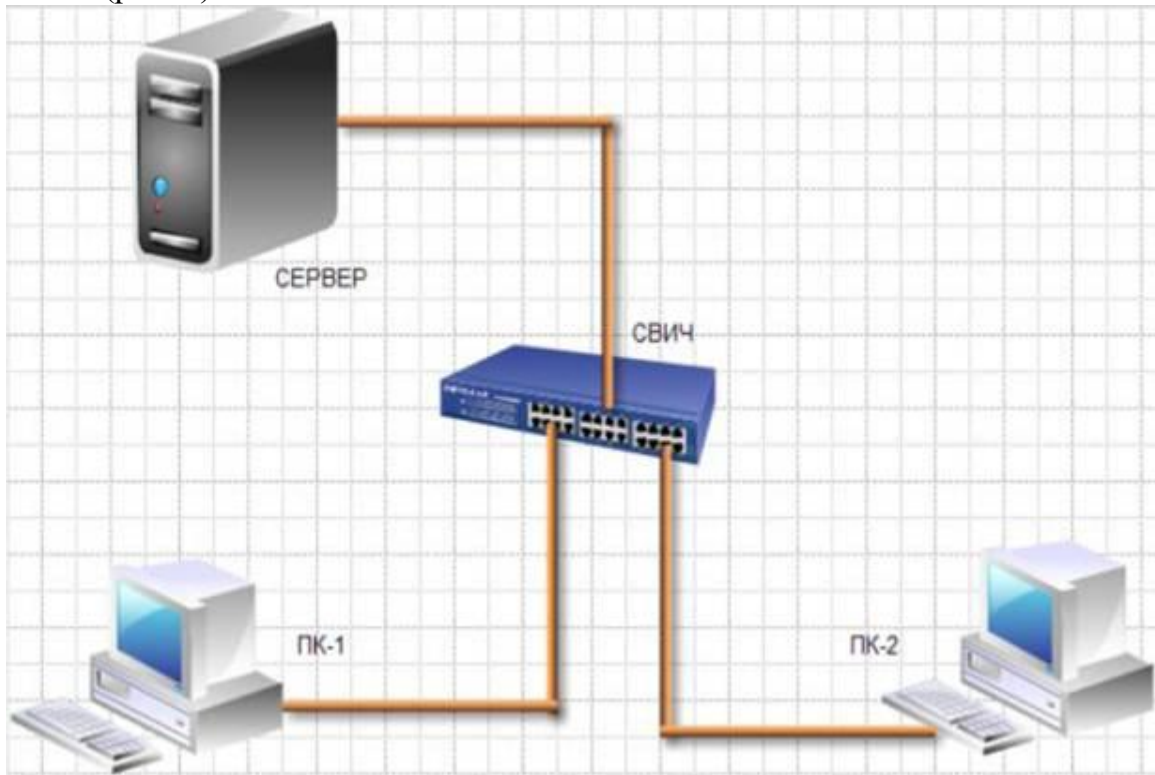


Рис. 1. Пример элементарной схемы сети, выполненной в EDraw Network Diagrammer

Задание 1

1. Постройте схему, изображенную на рисунке 1.
2. Для выбора компьютеров и мониторов из библиотеки (Libraries) нужно выбрать команду **Network-Computers and Monitors**, а для выбора кабелей – команду **Network and Peripherals**.

Задание 2 Нарисуйте схему помещения, изображенного на рисунке 2.

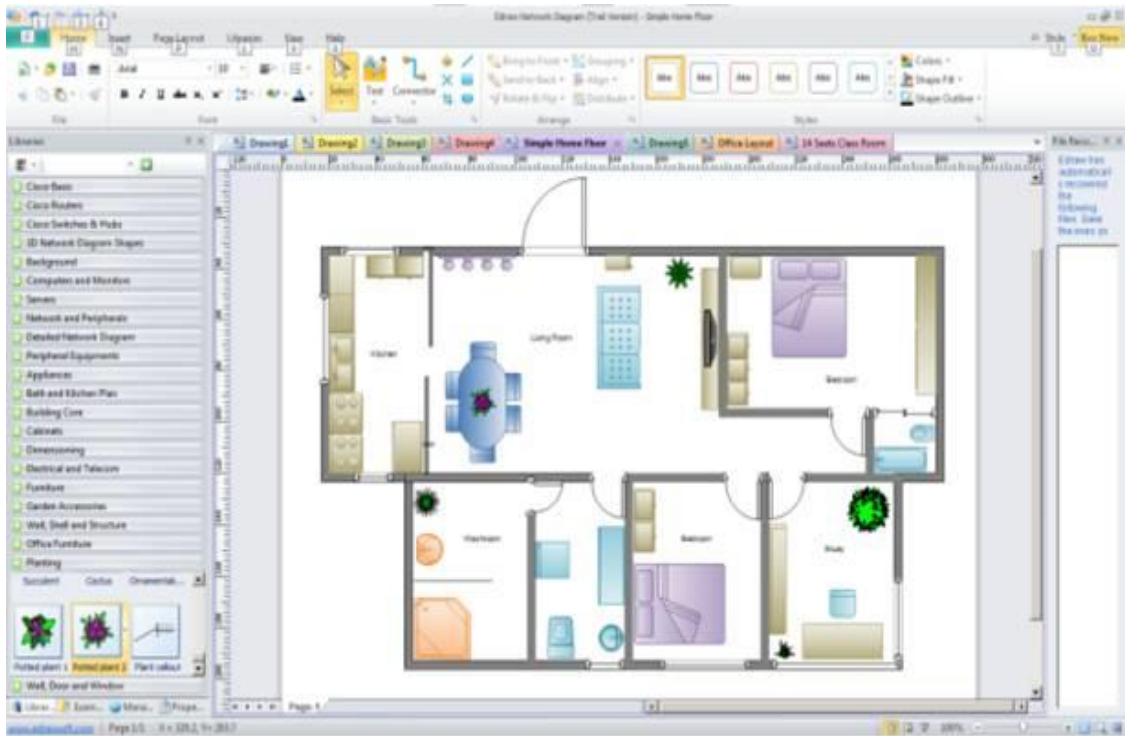


Рис.2. Изображение офисного помещения, нарисованного в Edraw Network Diagrammer

В этом случае из библиотеки нужно выбрать вариант **Floor Plans** (рис. 3).

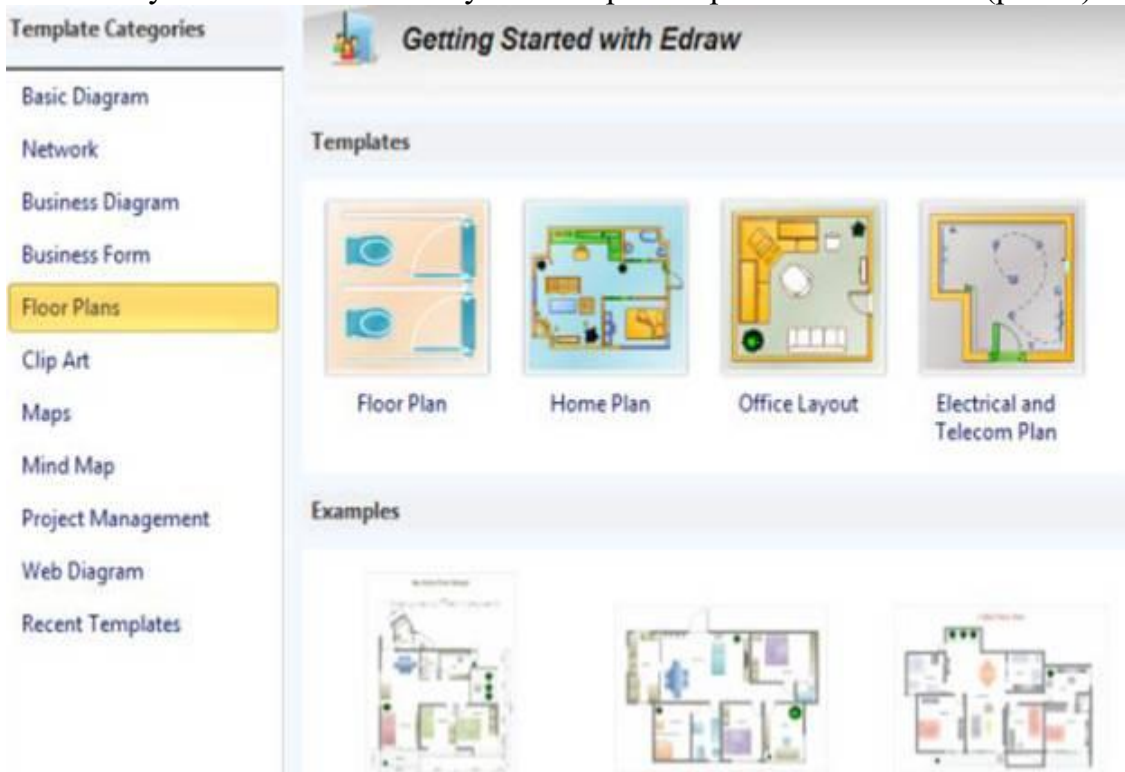


Рис 3. Различные схемы офисов, для размещения в них ПК

Задание 3. В программе Edraw Network Diagrammer повторите схему, показанную на рис.4. Поясни-те, что за устройства присутствуют в данной сети и как они работают.

Рис. 4.



Рис. 4. Схема сети небольшого офиса

Задание 3. Повторите рисунок, изображающий расположение компьютеров в компьютерном классе (рис.5).

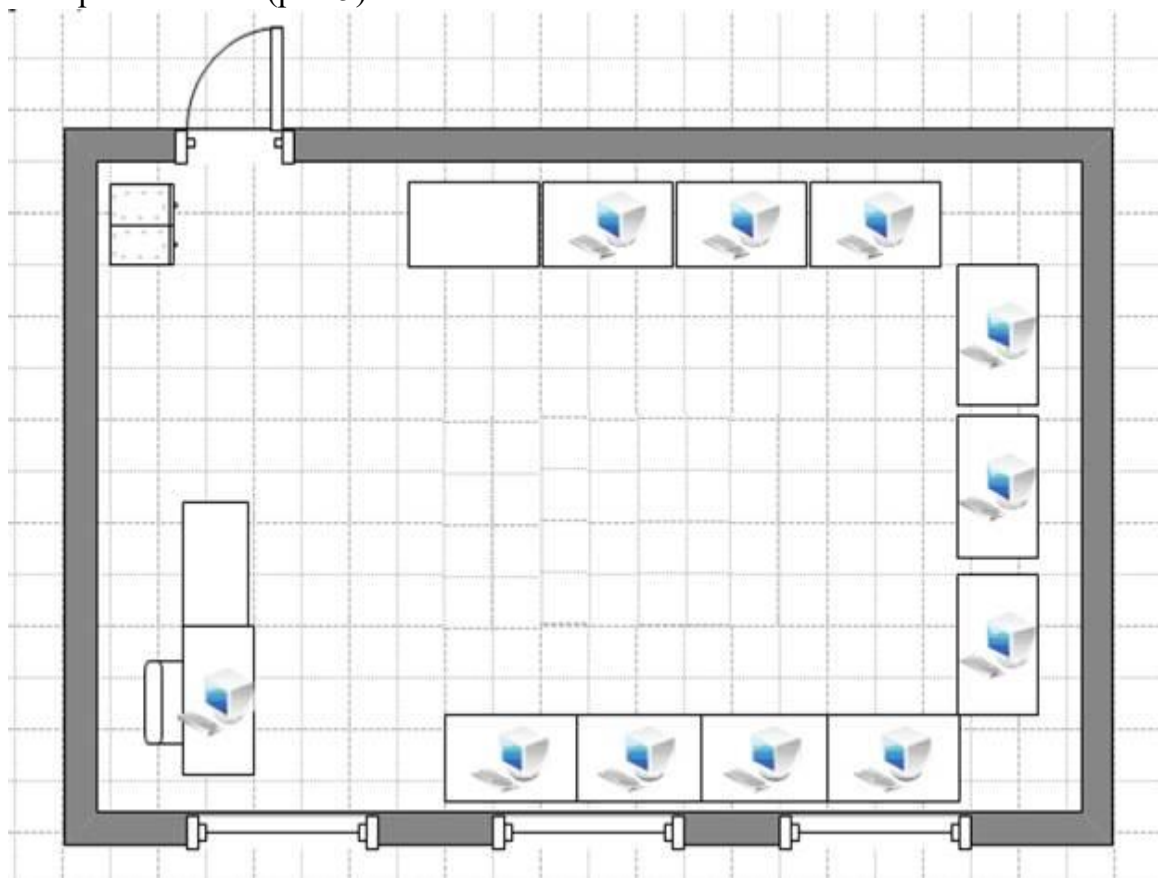


Рис. 5. Расположение компьютеров в некотором ВЦ

Контрольное задание :

Используя возможности программы **EDraw Network Diagrammer** создайте схему помещения и расположения [компьютерной техники](#) в нашем кабинете (по аналогии с рис. 5)

Лабораторная работа №12 Создание пользователей домена

Знакомимся с консолью

Консоль MMC (Microsoft Management Console) группирует средства администрирования, которые используются для администрирования компьютеров, служб, других системных компонентов и сетей.

Windows Server 2003 содержит *консоль* для управления сервером и его компонентами. Иначе говоря, *консоль* - это специальная панель для выполнения команд и операций, проведения настроек и установок сервера.

Включим *сервер*, затем на нем выполним команду **Пуск-Выполнить-mmс** появится Консоль1 (рис. 14.1).

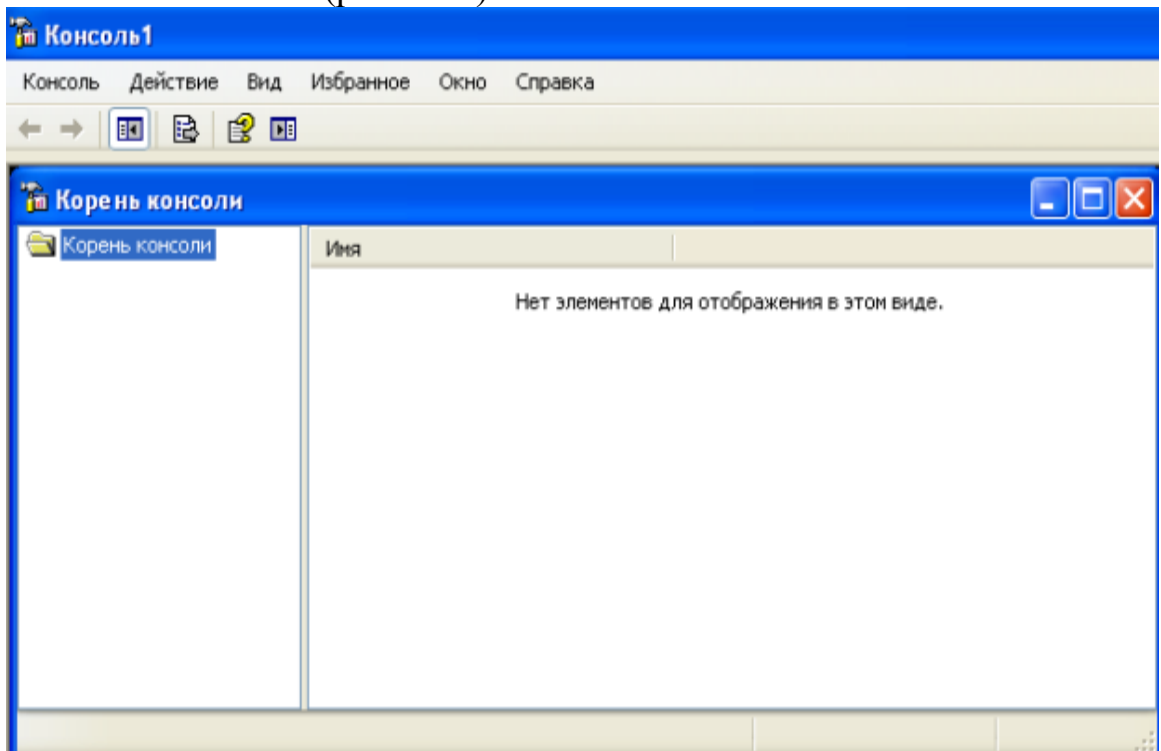


Рис. 14.1. Консоль1

Выполним команду **Консоль-Добавить** или **удалить оснастку** и добавим две службы (оснастки), установленные нами ранее – *AD* и *DNS* (рис. 14.2 и рис. 14.3).

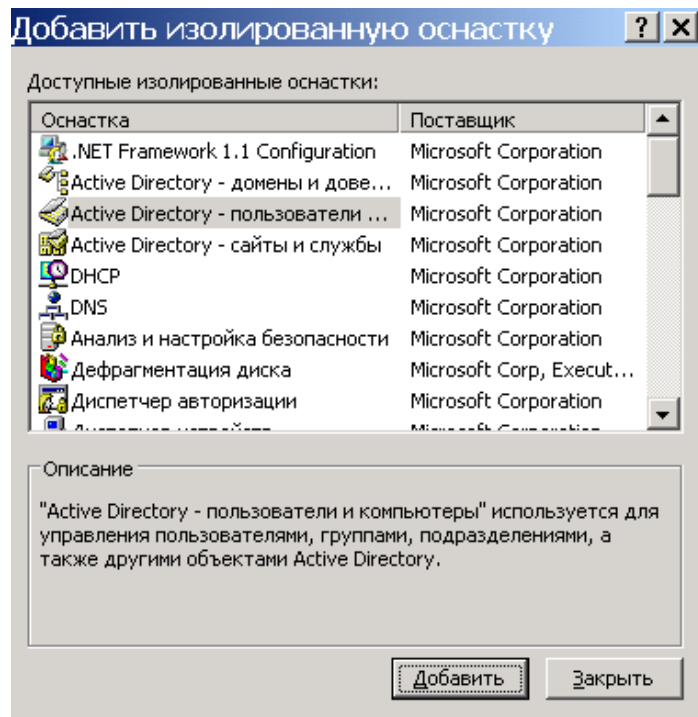


Рис. 14.2. Окно выбора служб (оснасток)

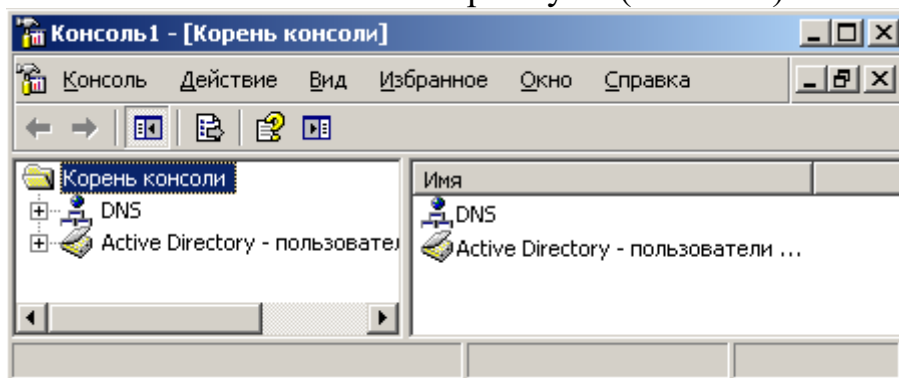


Рис. 14.3. В консоль мы добавили две службы (оснастки)

Выполняем команду **Консоль-Сохранить**, как и



сохраняем консоль `Консоль1.msc` на рабочий стол. Теперь запустим консоль с рабочего стола и откроем **Active Directory**, где увидим наш домен (рис. 14.4).

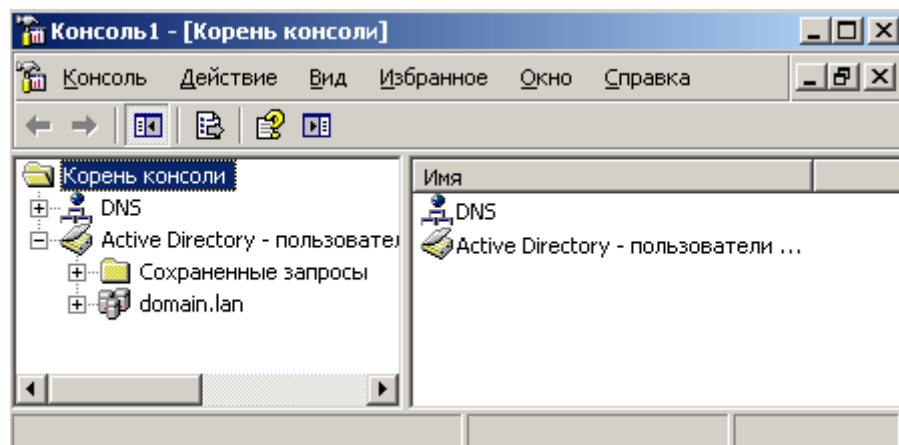


Рис. 14.4. Видим наш домен (domain.lan)

В домене создаем подразделения, а в них-объекты

Наша цель – создать в домене нового пользователя – **Администратор** с помощью консоли. Для этого в домене правой кнопкой вызовем *меню* и команду **Создать-Подразделение**, чтобы создать раздел (*объект*) **Администратор** (рис. 14.5).

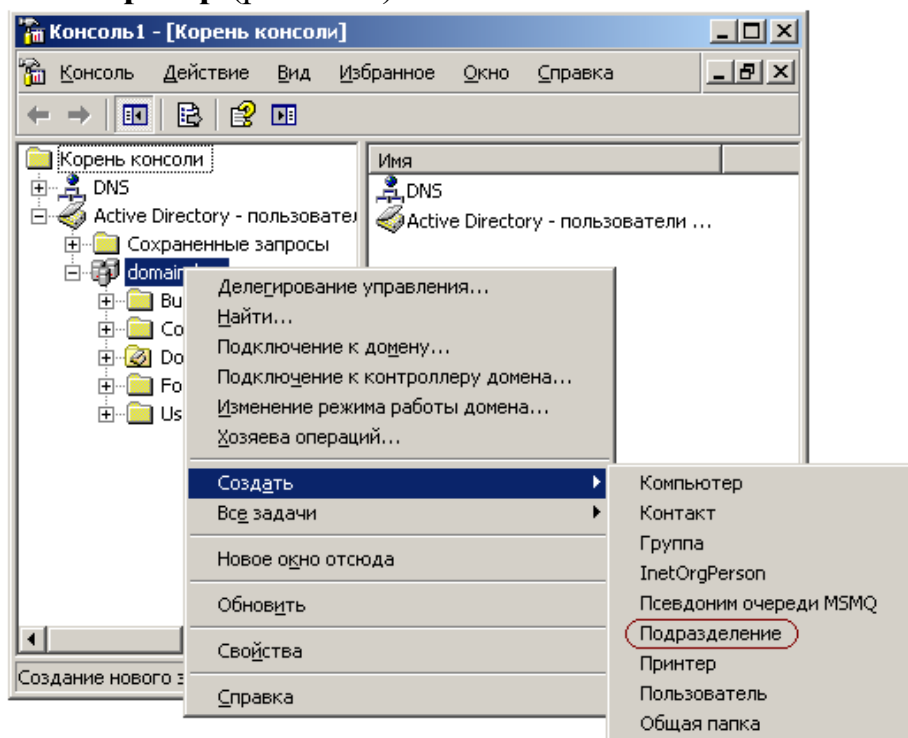


Рис. 14.5. Создаем подразделение Администратор

Аналогично создадим подразделение **Пользователи** (рис. 14.6 и рис. 14.7).

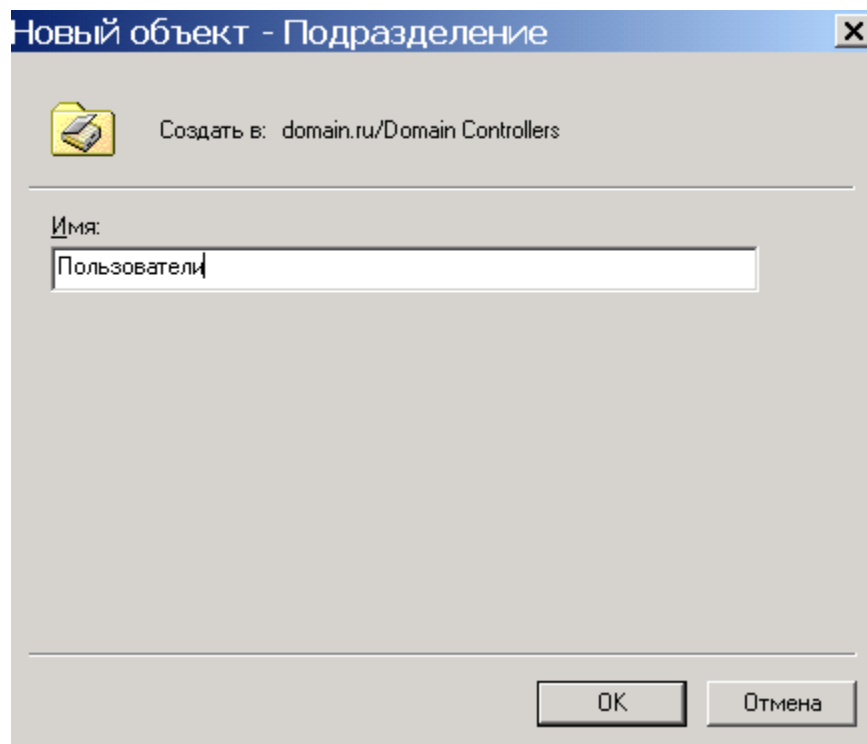


Рис. 14.6. Имя Пользователи не появится само - его нужно написать

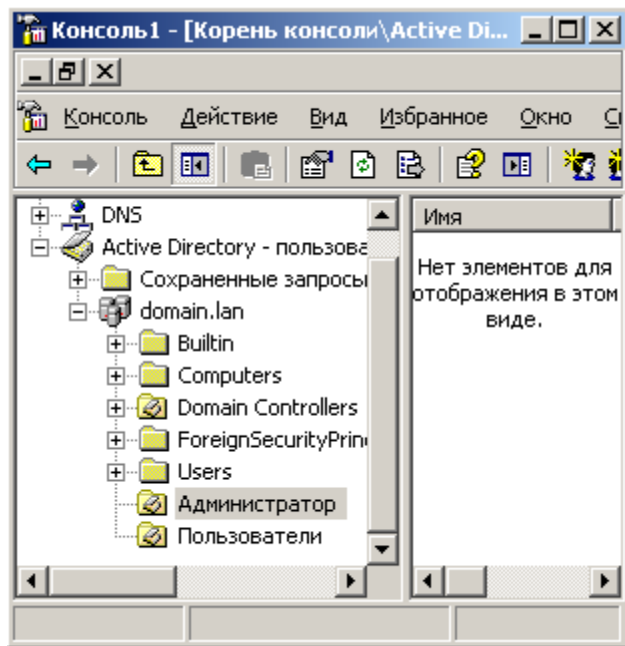


Рис. 14.7. В Active Directory мы создали два подразделения

Для пользователей в *поле* справа (**Имя-Тип-Описание**) щелкаем правой кнопкой мыши и выполняем команду **Создать-Пользователь** (рис. 14.8). Заполняем форму.

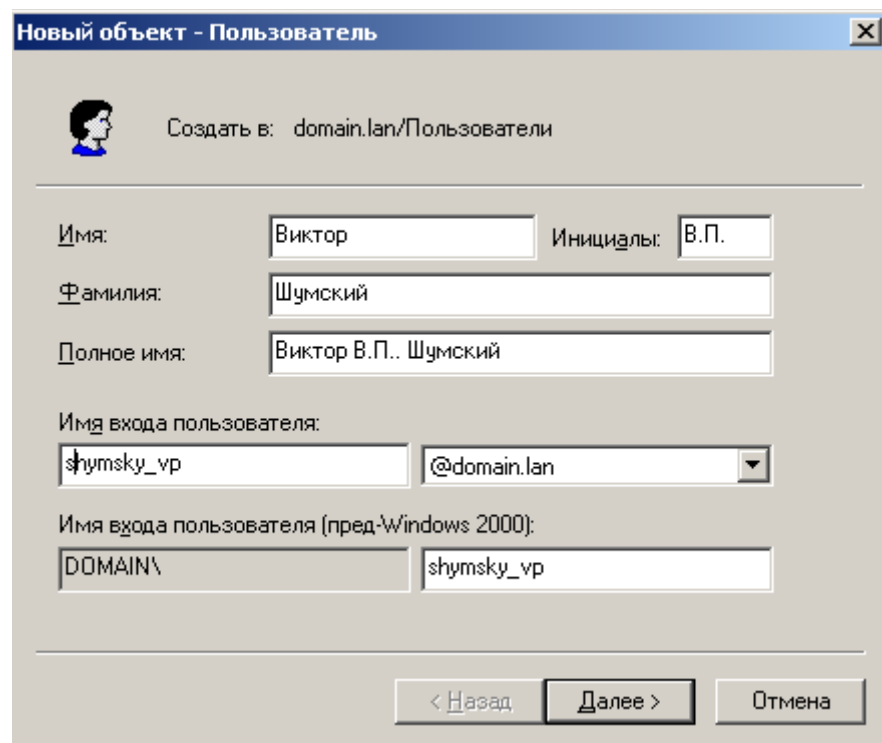


Рис. 14.8. В домене создаем пользователя

Нажимаем **Далее** и вводим *пароль* и создаем нового пользователя доменом. Аналогично создаем администратора (рис. 14.9).

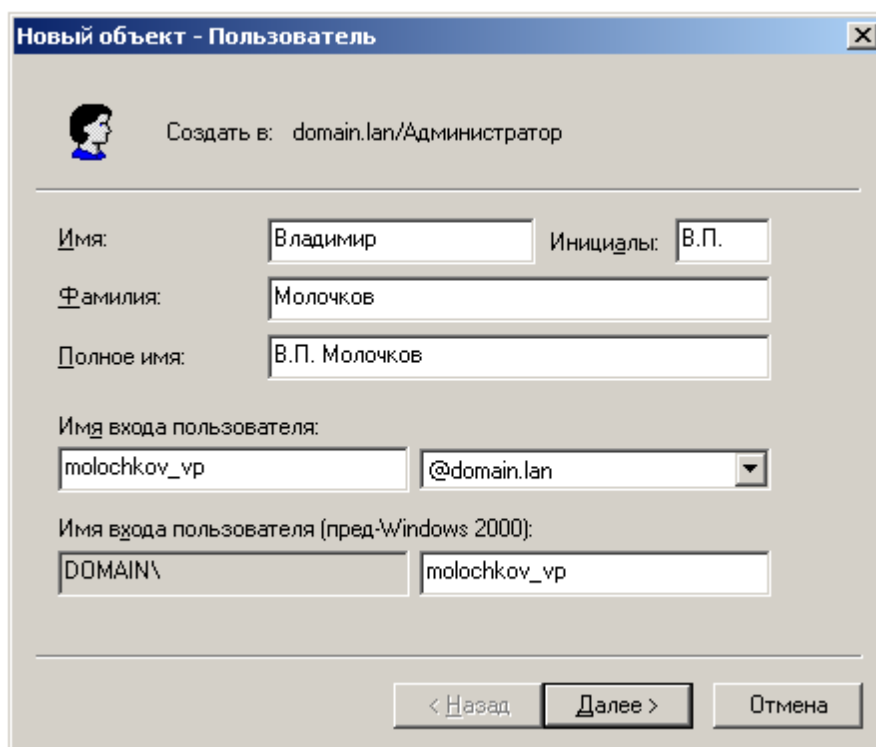


Рис. 14.9. Создаем администратора

Таким способом мы создадим одного пользователя в разделе **Администратор** и одного пользователя в разделе **Пользователи**. То есть, в домене мы создали подразделение, а в нем – два объекта. Если теперь дважды щелкнуть на пользователе-администраторе, то мы увидим, что он является членом группы **Пользователи домена** (рис. 14.10). Второй *пользователь* (Шумский) – также *член группы Пользователи домена*.

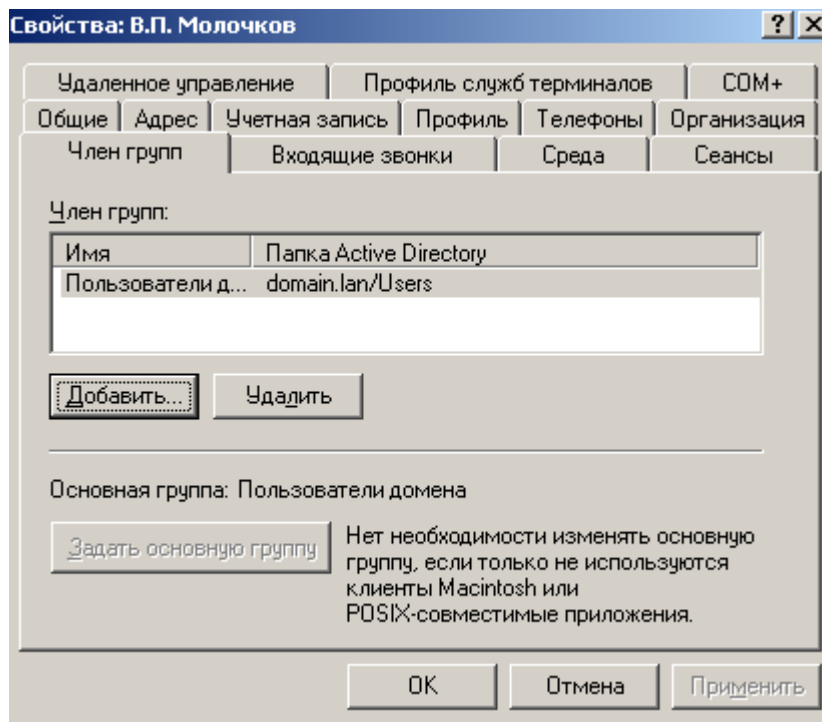


Рис. 14.10. Молочков является членом группы Пользователи домена

Нажмем в этом окне на кнопку **Добавить** и далее - **Дополнительно-Поиск**, ищем администратора домена, ОК (рис. 14.11).

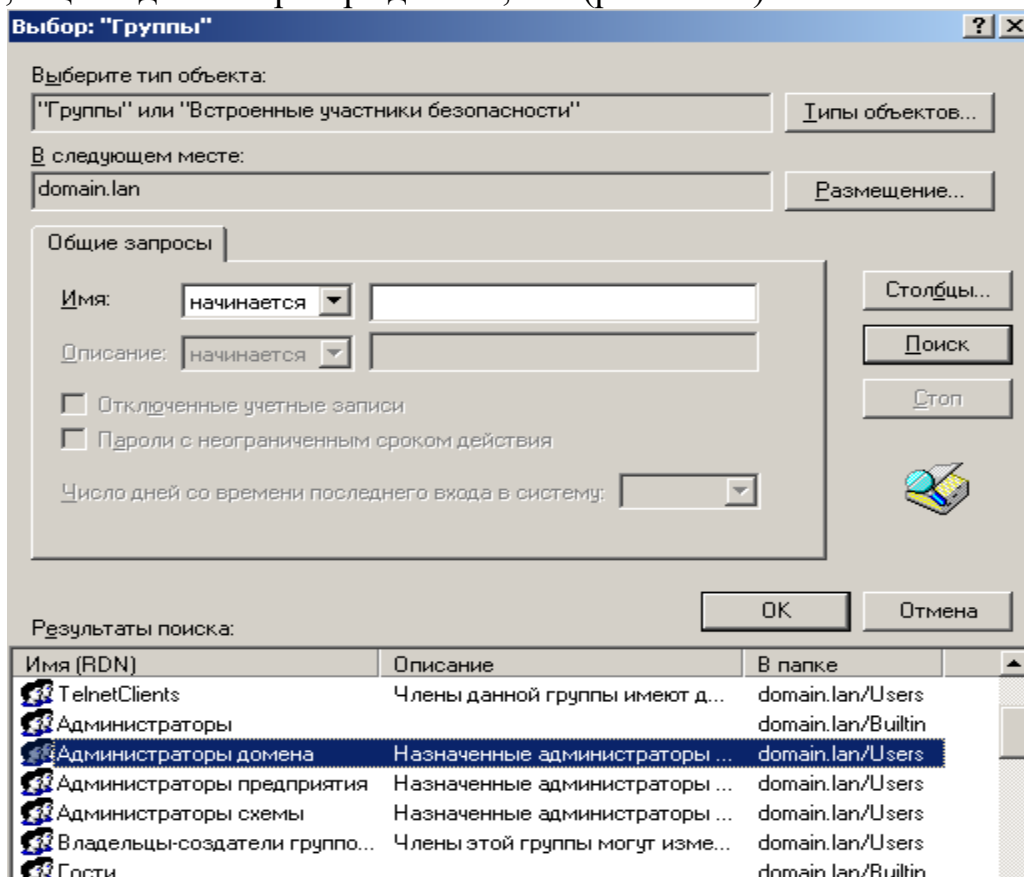


Рис. 14.11. Выбираем группу администраторы домена

Теперь стаем на строчку **Администраторы домена** и нажимаем на кнопку **Задать основную группу** (рис. 14.12).

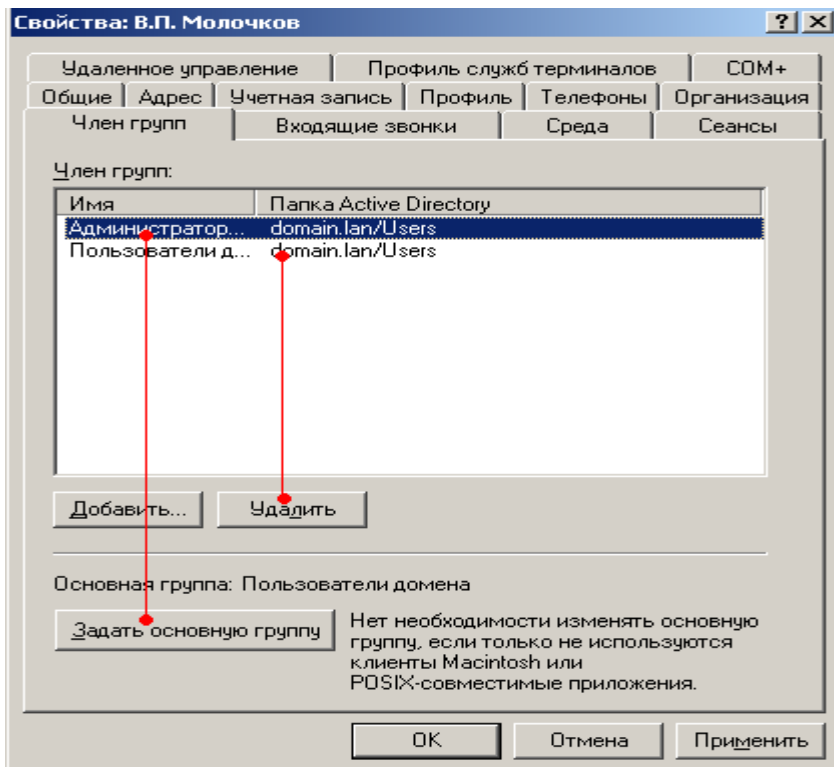


Рис. 14.12. Основная группа-Администраторы домена

Группу **Пользователи домена** удаляем кнопкой **Удалить** и нажимаем **ОК**. Теперь, если в консоли щелкнуть на пользователе Молочков, то вы увидите, что он - *член группы* администраторов домена – рис. 14.13. Задача решена.

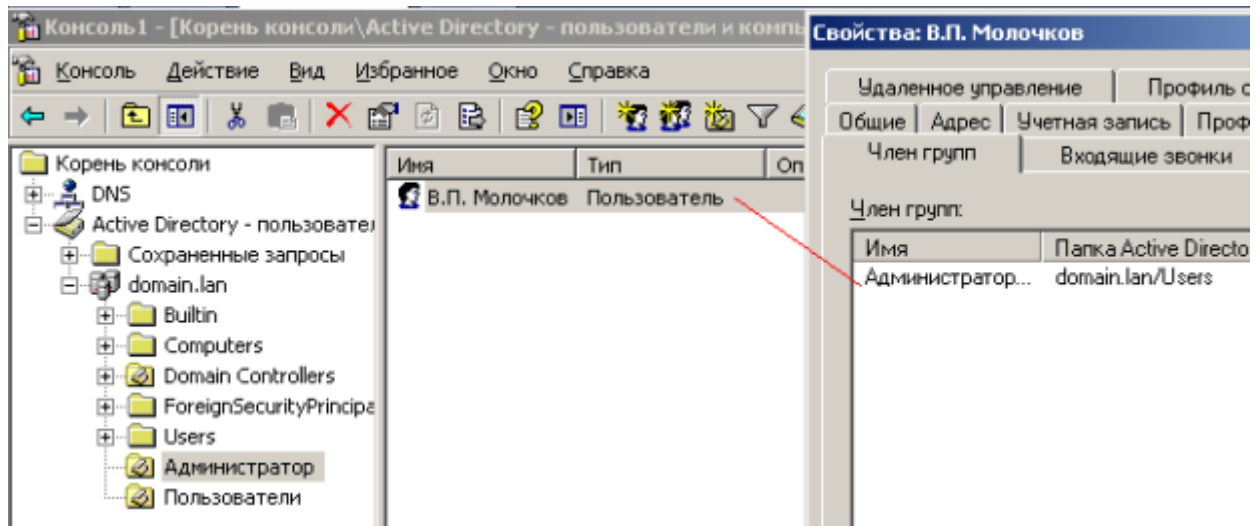


Рис. 14.13. Молочков - член группы администраторов домена (из пользователей домена он удален)

Ресурсы в локальной сети

Давайте на сервере, на диске C:\ создадим две папки: **MUSIC** (только чтение), **VIDEO** (полный доступ) и затем откроем эти ресурсы для доступа пользователям (рис. 14.14).

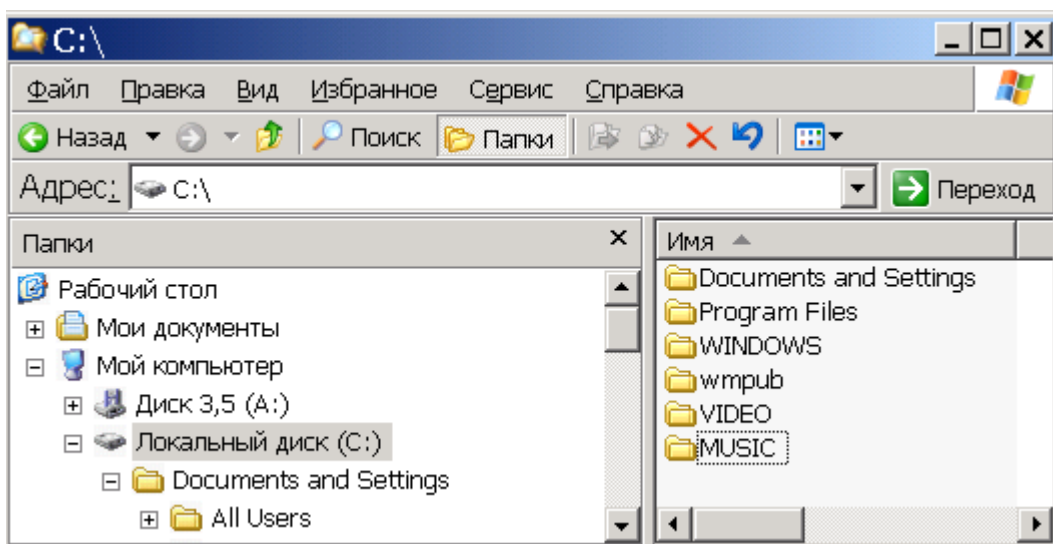


Рис. 14.14. На диске C:\ созданы две папки

Для назначения папкам требуемых атрибутов щелкнем на ней правой кнопкой мыши и выберем команду **Общий доступ и безопасность**. В данном окне установим *переключатель Открыть общий доступ* (рис. 14.15).

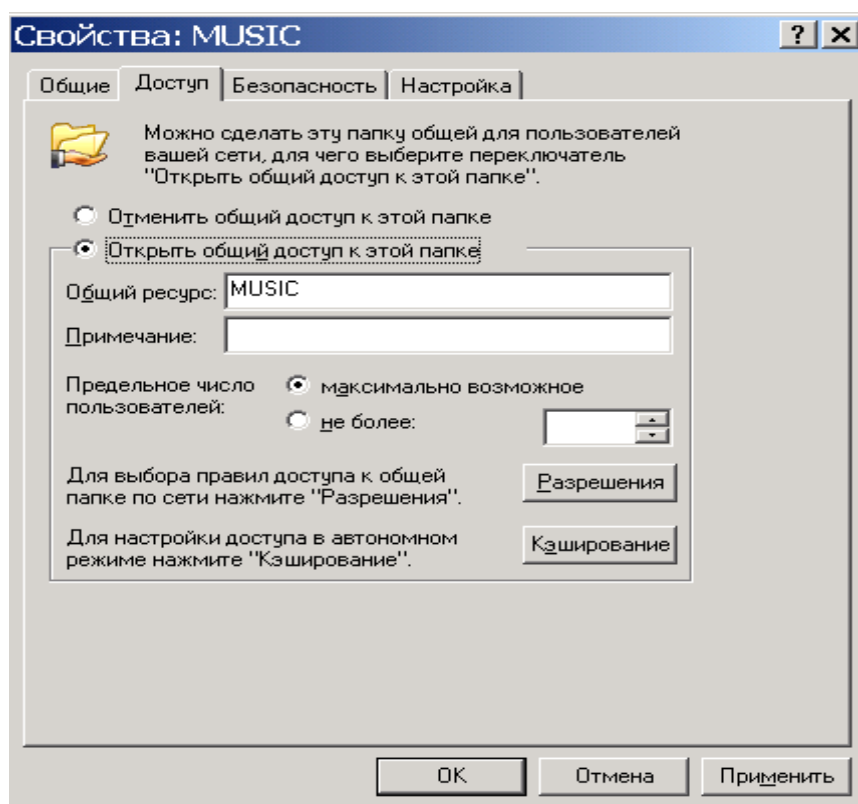


Рис. 14.15. Окно свойств папки Music

На вкладке **Разрешения** вы можете осуществить *поиск* пользователей домена и добавить их в пользователи папки MUSIC (рис. 14.16). Для этого нужно выполнить команды **Добавить-Дополнительно-Поиск** и указать нужного пользователя. Строку **Все** пользователи, стоящую по умолчанию, следует удалить.

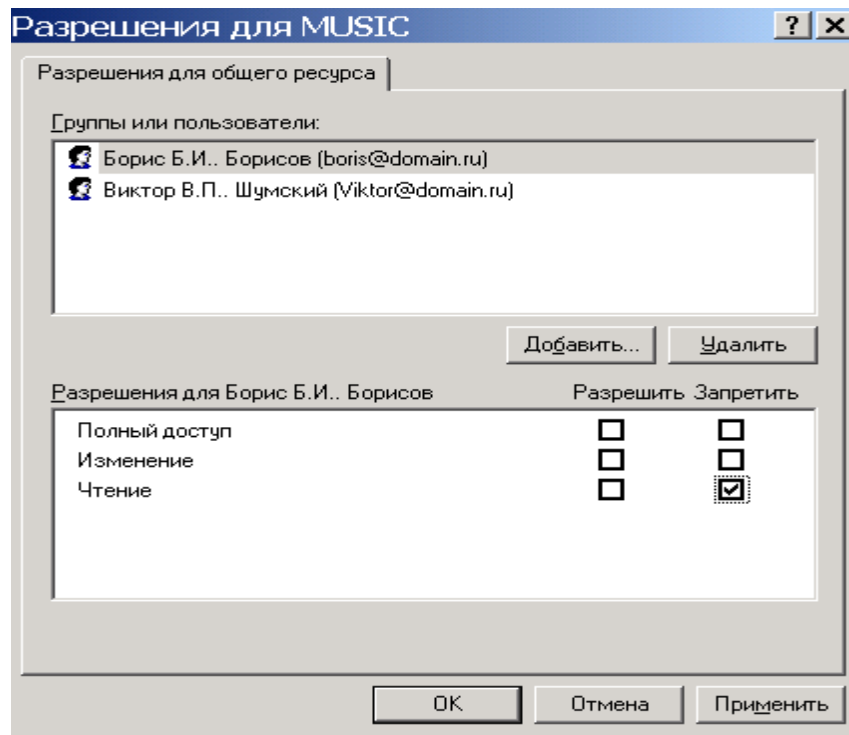


Рис. 14.16. Добавлены два пользователя с правом чтения папки MUSIC
 Таким же образом устанавливается *доступ* для других папок (рис. 14.17).

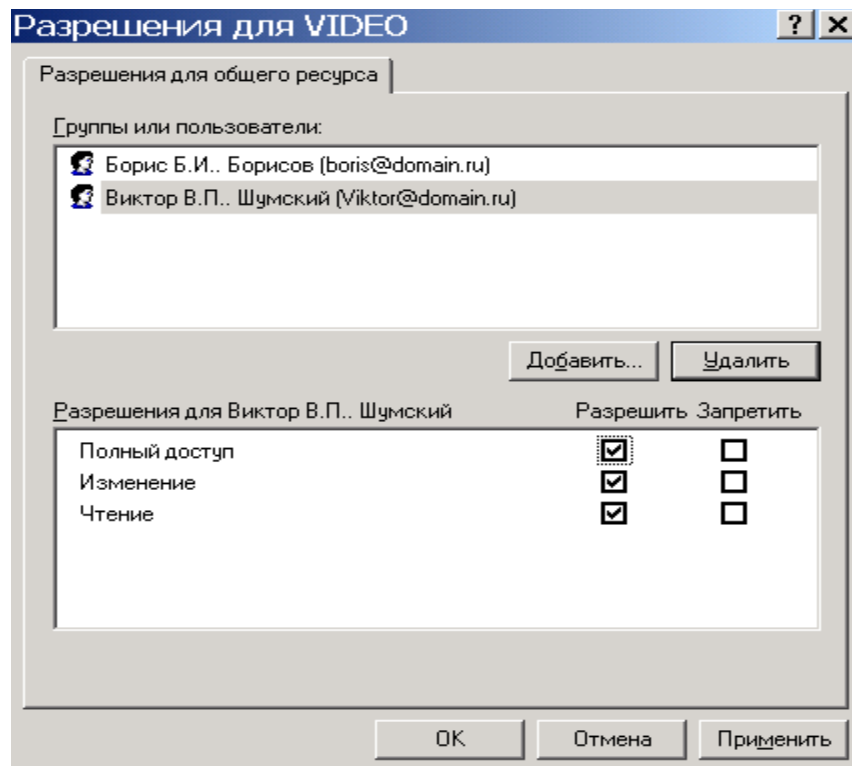


Рис. 14.17. К папке VIDEO пользователи имеют полный доступ
 Перегрузим и посмотрим, как эти ресурсы будут выглядеть со стороны клиента, для этого войдем в *домен* под именем пользователя Борисов (рис. 14.18).

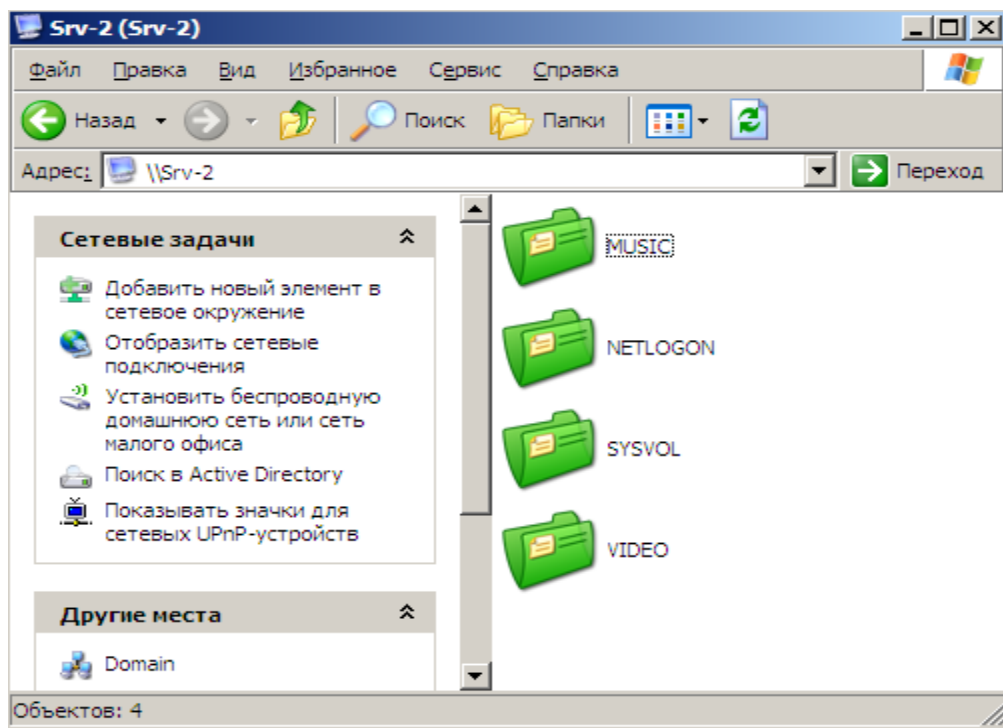


Рис. 14.18. Вид папок на сервере в сетевом окружении

Как вариант на клиенте можно выполнить команду **Пуск-Выполнить** и `\\Srv-2003`. Мы увидим, что со стороны клиента сетевые ресурсы сервера открыты и он может ими пользоваться согласно назначенным ему разрешениям. Пользователи, не входящие в *домен*, папки также увидят, но пользоваться ими не смогут (Для компьютеров не входящих в *домен* папки закрыты).

Краткие итоги

В работе мы научились производить практические действия с консолью, создавать в домене *подразделения*, а в них-объекты. Научились организовывать *доступ* к ресурса в локальной сети. К работе прилагается скринкаст.

Лабораторная работа №13

Администрирование сети

Изменение групповой политики (ГП) для одного пользователя (не администратора)

Изменим ГП для пользователя домена В.П. Шумский, а именно, запретим ему в рабочее время заниматься видеомонтажом в программе [Windows Movie Maker](#).

Сначала создадим политику для него кнопкой **Создать** (рис. 15.1).

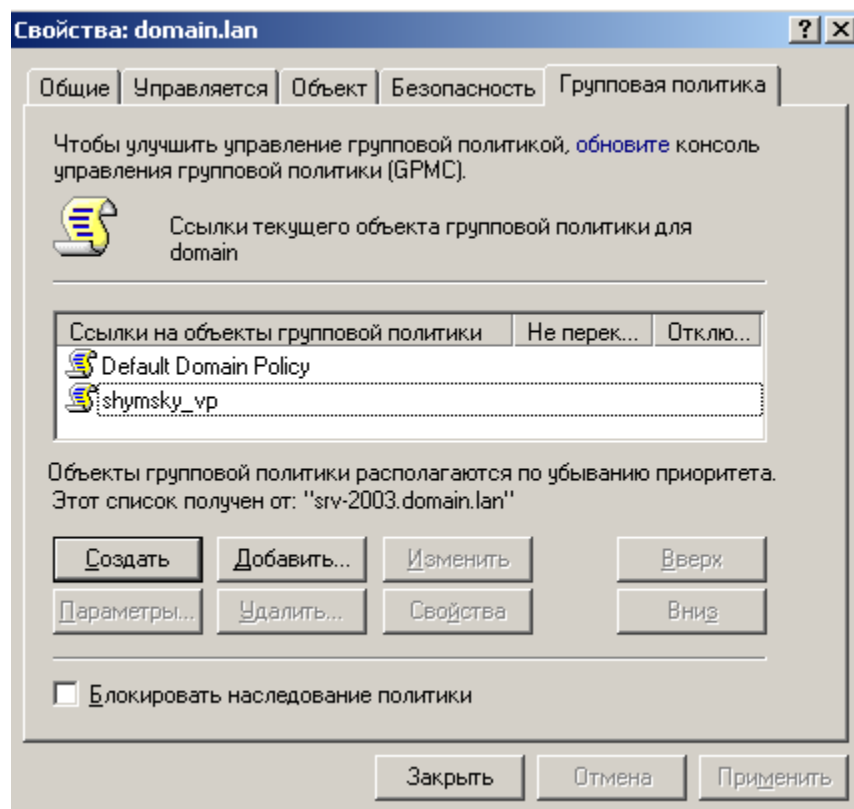


Рис. 15.1. В данном окне нажмем на кнопку Создать

Далее необходимо, чтобы созданная нами политика не применялась ко всем пользователям. Для этого выделяем пользователя, нажимаем кнопку **Свойства** и переходим на вкладку **Безопасность**, где убираем флажок **Применение групповой политики** для всех пользователей (рис. 15.2).

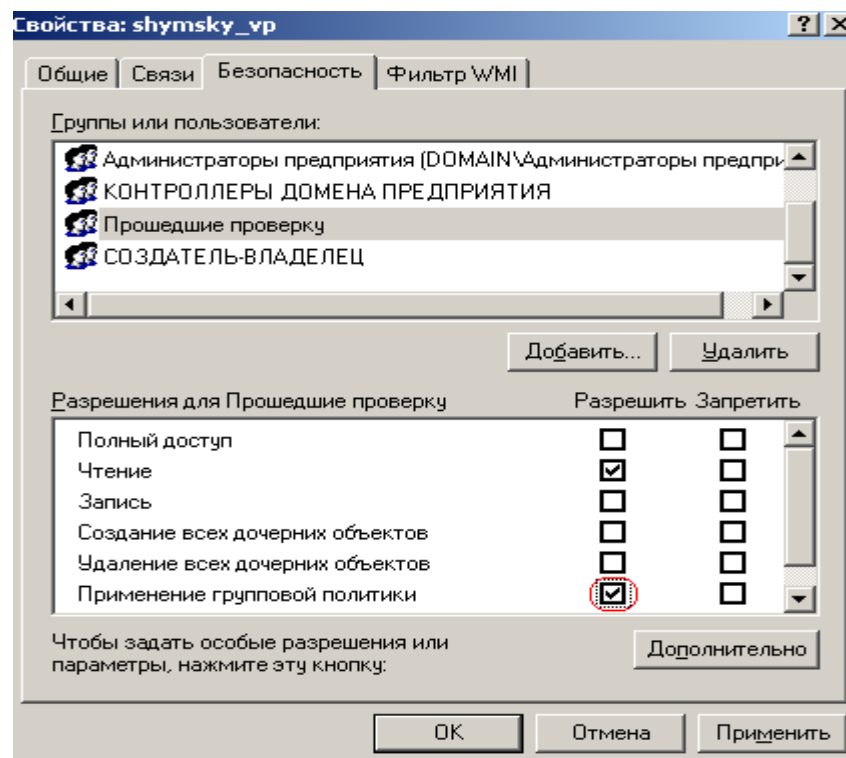


Рис. 15.2. Выделенный флажок необходимо убрать

Теперь в этом же окне нажмите на кнопку **Добавить**, найдите и добавьте пользователя (рис. 15.3). Для Шумского нужно разрешить применение групповой политики.

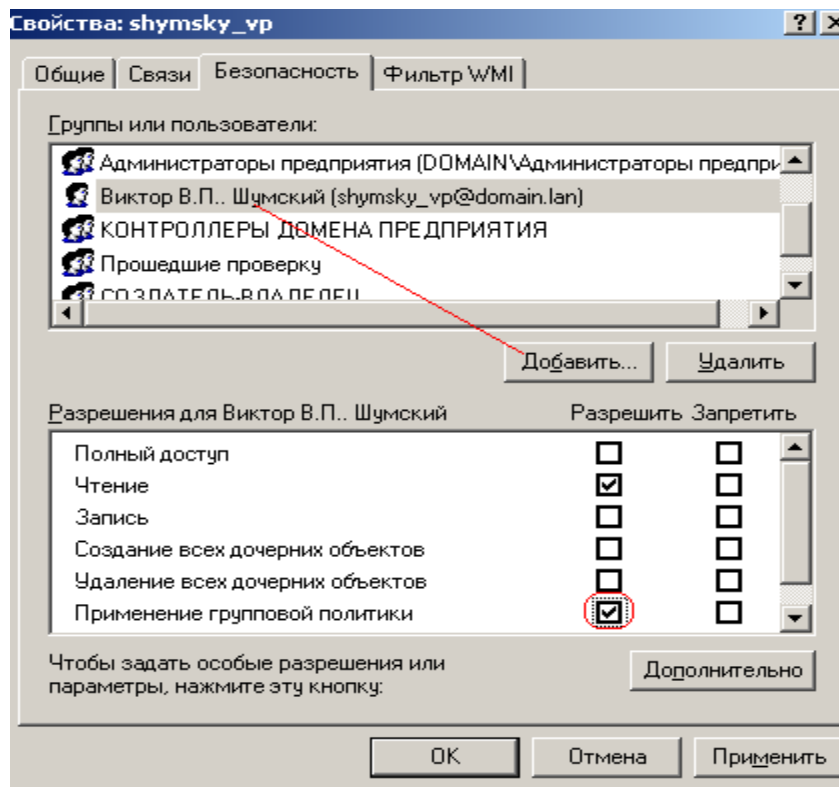


Рис. 15.3. В.П. Шумский добавлен и к нему будет применена групповая политика

Жмем **Применить**, а затем **Изменить** (рис. 15.4).

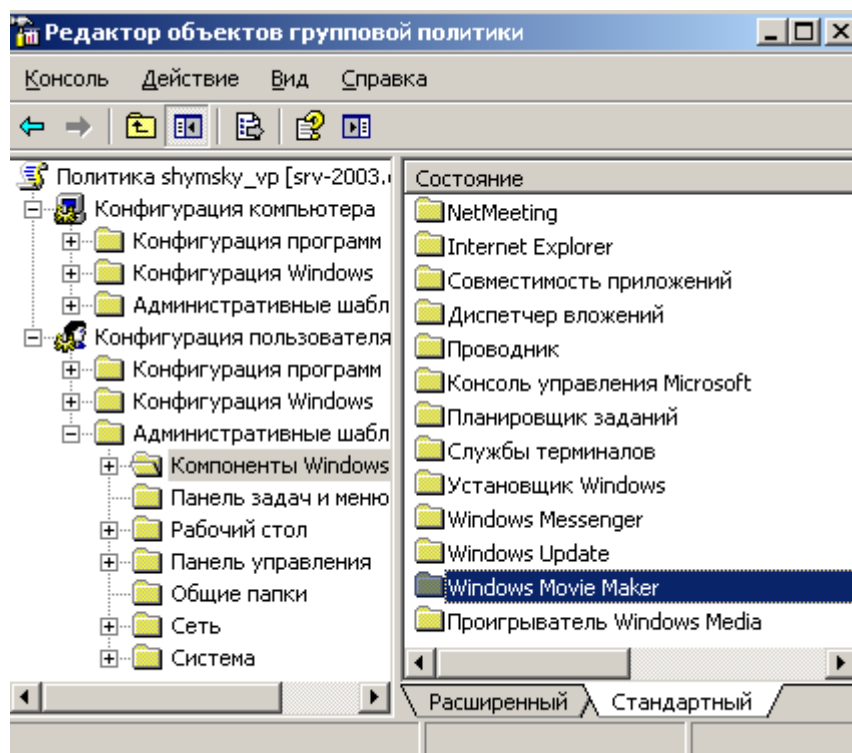


Рис. 15.4. В компонентах Windows находим Windows Movie Maker
 Далее активируем *переключатель* **Запретить выполнение программы Windows Movie Maker** – рис. 15.5.

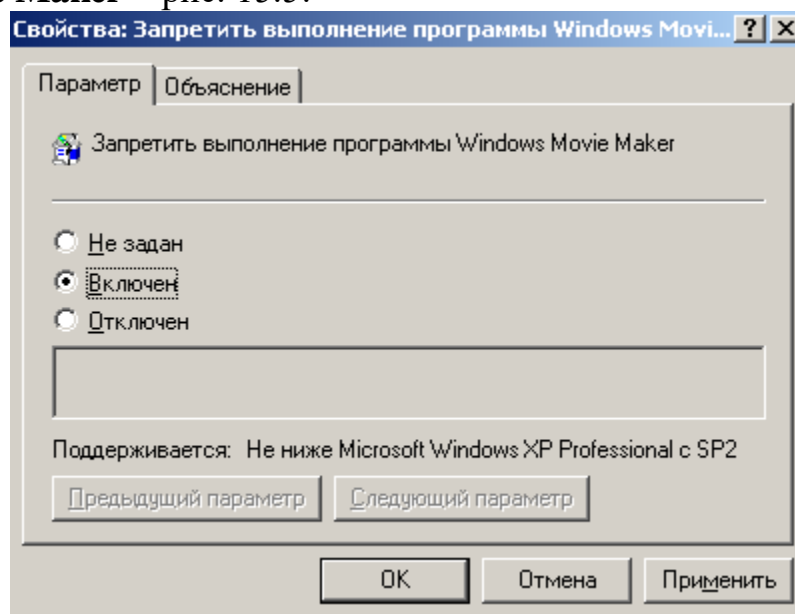


Рис. 15.5. Активируем средний переключатель
 Теперь зайдем в клиента под учетной записью Виктора Шумского (рис. 15.6).

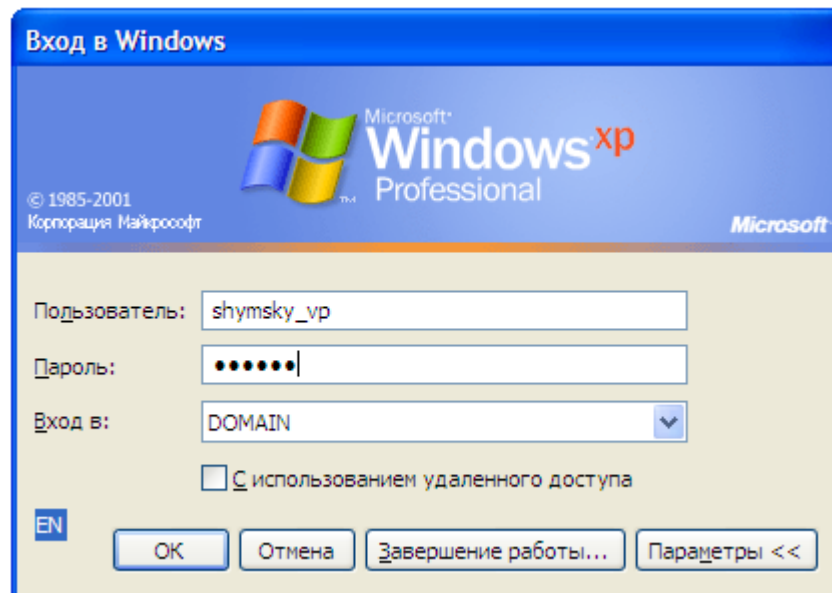


Рис. 15.6. Входим в домен

При попытке запустить программу Movie Maker получаем следующее сообщение (рис. 15.7).

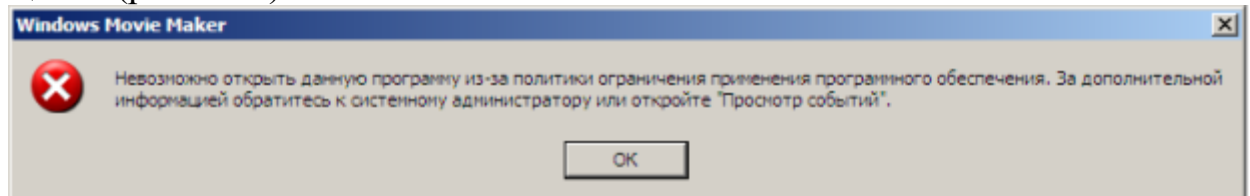


Рис. 15.7. Групповая политика работает

Наша цель достигнута. Попробуйте зайти в домен под другим пользователем и вы увидите, что программа Movie Maker успешно работает.

Создание групповой политики для группы пользователей

Вспомним, что ранее мы создали группу пользователей *Admin* (рис. 15.8).

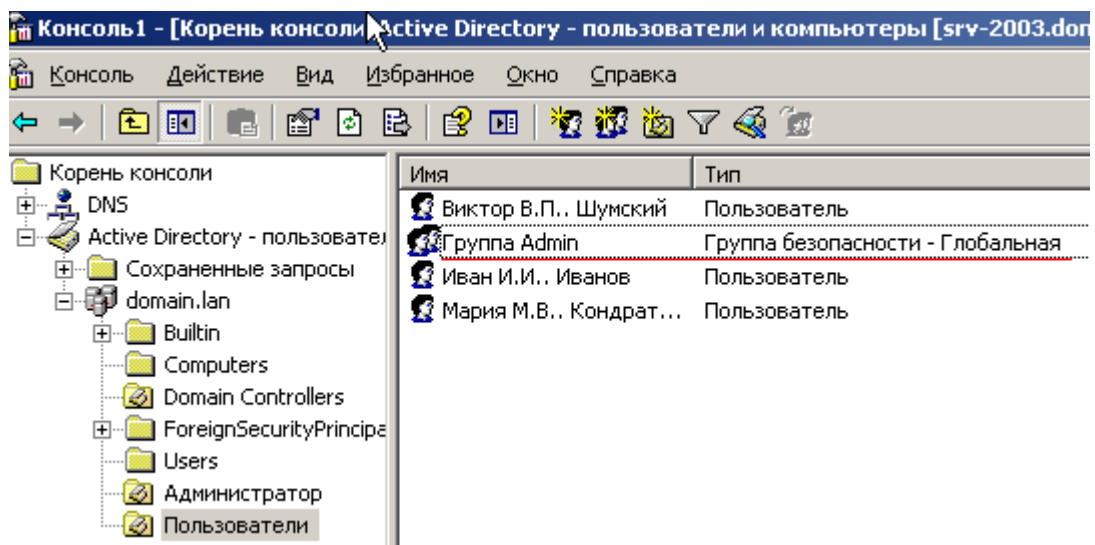


Рис. 15.8. В AD имеется группа пользователей Admin

Теперь для этой группы пользователей мы создадим групповую политику. Делается это так же, как для отдельного пользователя. Щелкаем на

названии домена в AD правой кнопкой мыши, выполняем команду **Свойства-Групповая политика-Создать** (рис. 15.9).

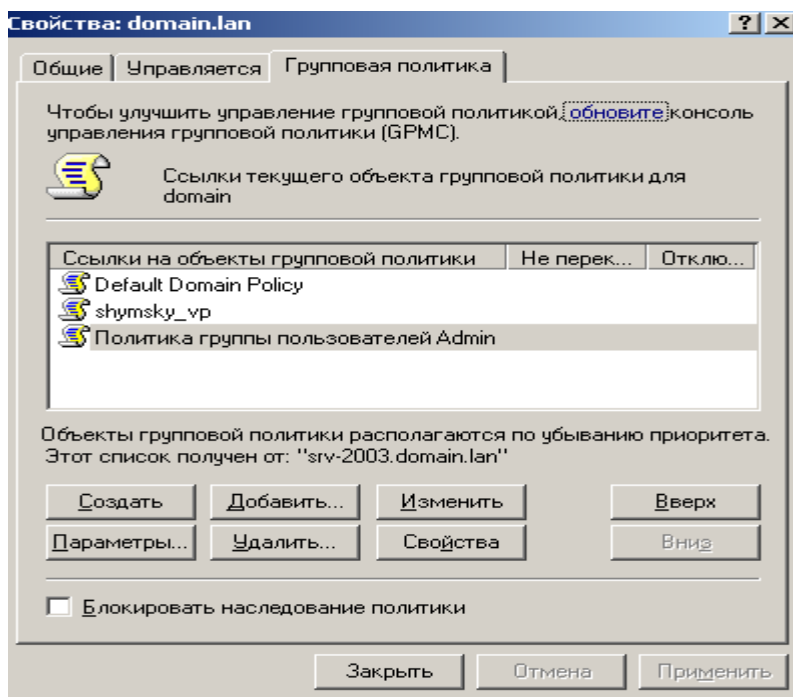


Рис. 15.9. Создаем политику группы пользователей Admin

Выполняем команду **Свойства-Безопасность** и везде убираем флажок **Применение групповой политики** (рис. 15.10).

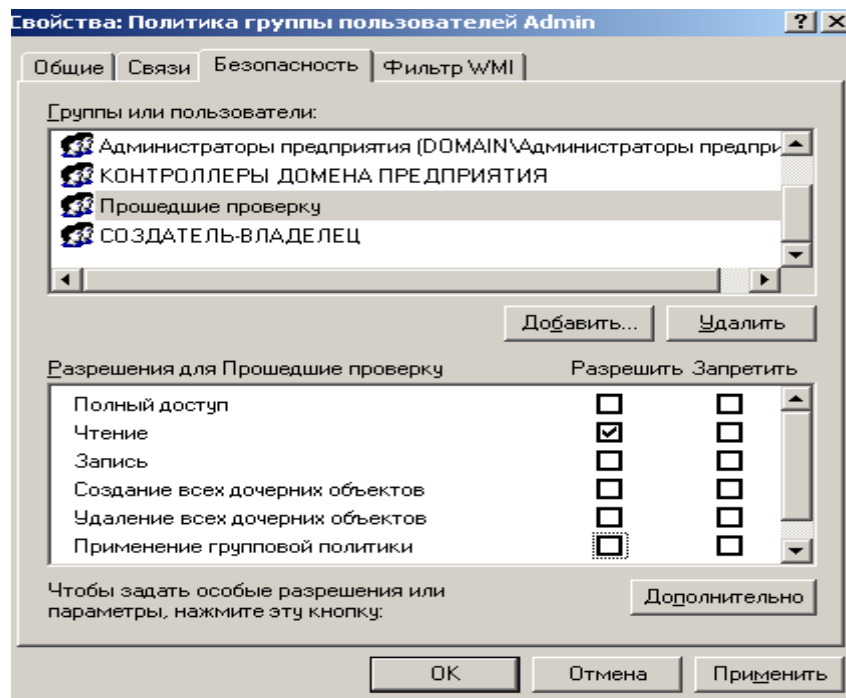


Рис. 15.10. В данном окне следует для всех групп и пользователей убирать флажок **Применение групповой политики**

В этом же окне нажимаем на кнопку **Добавить** и кнопкой **Поиск** находим группа **Admin** (рис. 15.11).

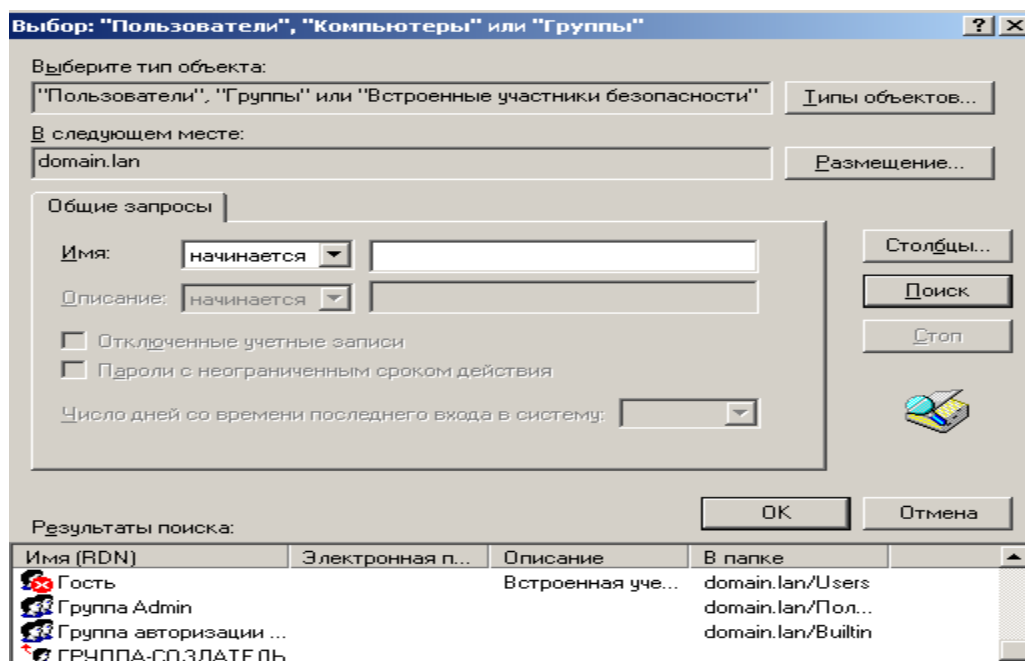


Рис. 15.11. Здесь нажимаем ОК

Для группы *Admin* активируем флажок применения групповой политики (рис. 15.12).

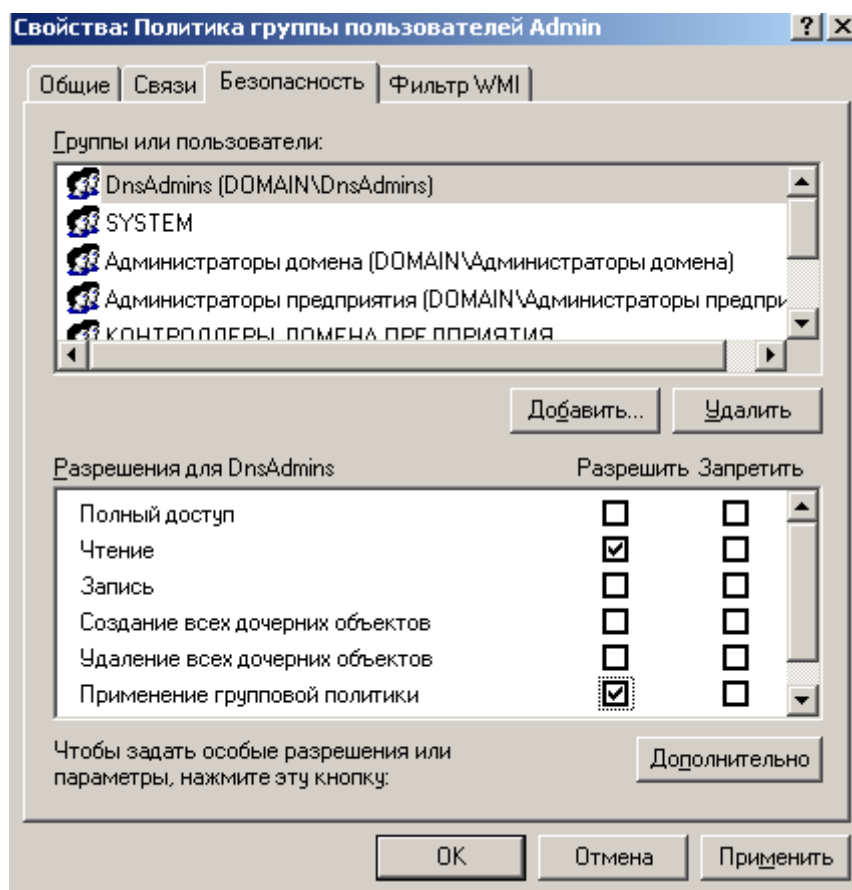


Рис. 15.12. Admin активируем флажок применения групповой политики

Если теперь войти в группу *Admin*, то увидим, что пока в ней один член, для которого будет применена *групповая политика* этой группы. Но кнопкой **Добавить** мы можем добавить сюда других пользователей, и автоматически *групповая политика* также будет применена и к ним (рис. 15.13).

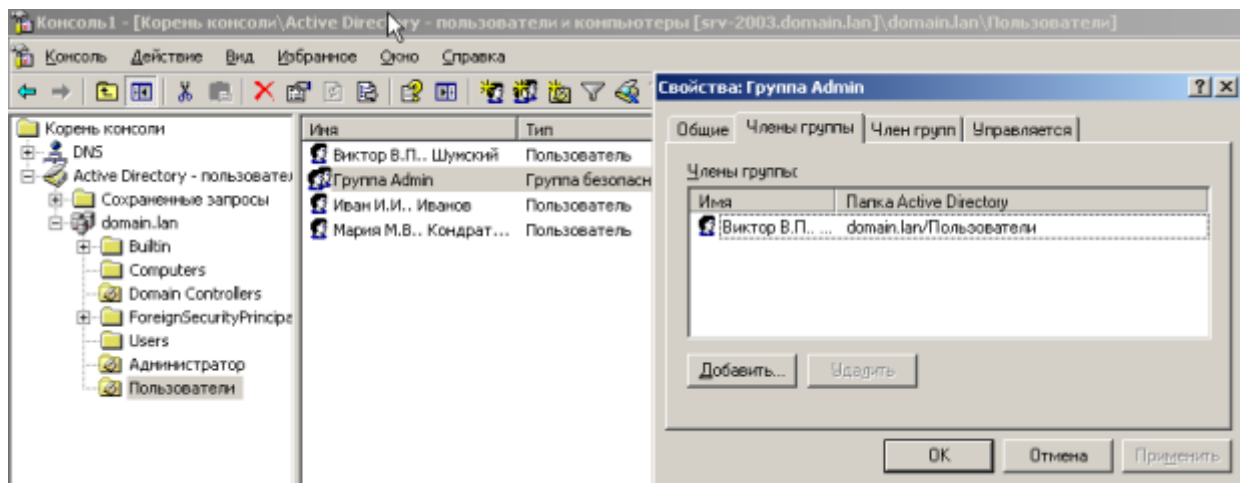


Рис. 15.13. Окно свойств группы Admin

Примечание

Если групповых политик несколько, то они применяются к пользователям все. Однако приоритетным является верхний уровень (рис. 15.14).

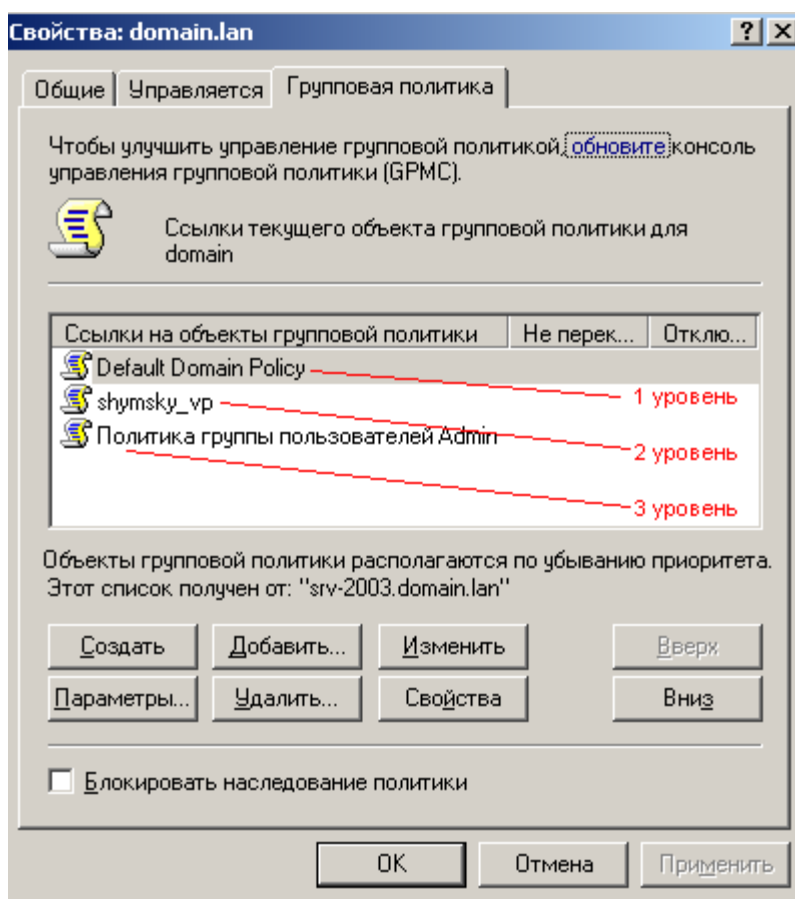


Рис. 15.14. Уровни приоритета групповых политик

Это означает, что если в политике 3 уровня написано запретить использовать Movie Maker, а в политике 1 уровня написано разрешить использовать Movie Maker, то использование Movie Maker будет разрешено. Однако приоритеты уровней можно изменить кнопкой **Параметры**. Другими словами, если мы нажмем на кнопку **Параметры** и активируем флажок **Не**

перекрывать, то политика по умолчанию первого уровня будет изменена политикой 3 уровня и использование Movie Maker будет запрещено (рис. 15.15).

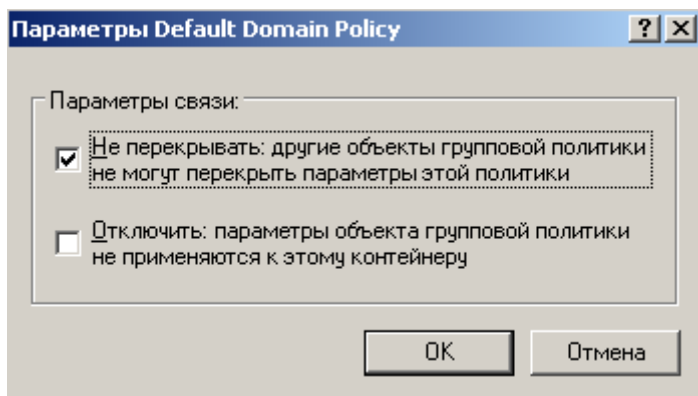


Рис. 15.15. Окно Параметры связи

Конфигурирование групповых политик для компьютеров

Когда мы создаем групповую политику для пользователя – она применяется для всех ПК, входящих в *домен*. При создании групповой политики для компьютеров мы можем запретить ряду пользователей входить в *компьютер*. Создадим новую групповую политику (рис. 15.16).

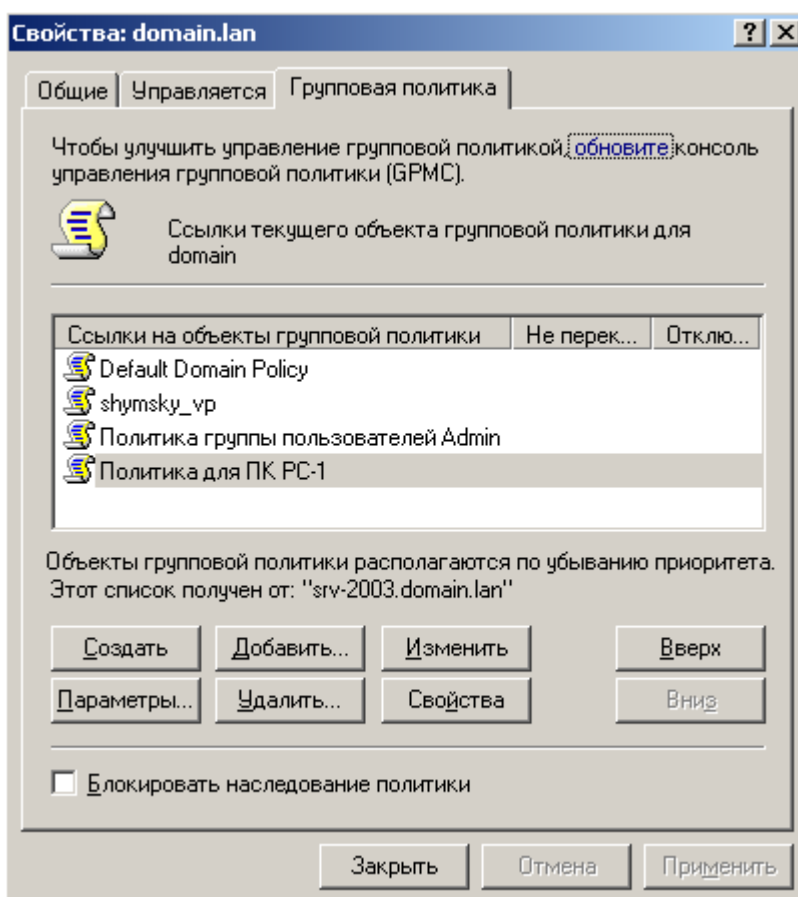


Рис. 15.16. Создадим Политика для ПК PC-1

Щелкнем мышкой на строчке **Политика для ПК PC-1** дважды и далее раскроем пункты **Конфигурация Windows-Параметры безопасности** (рис. 15.17).

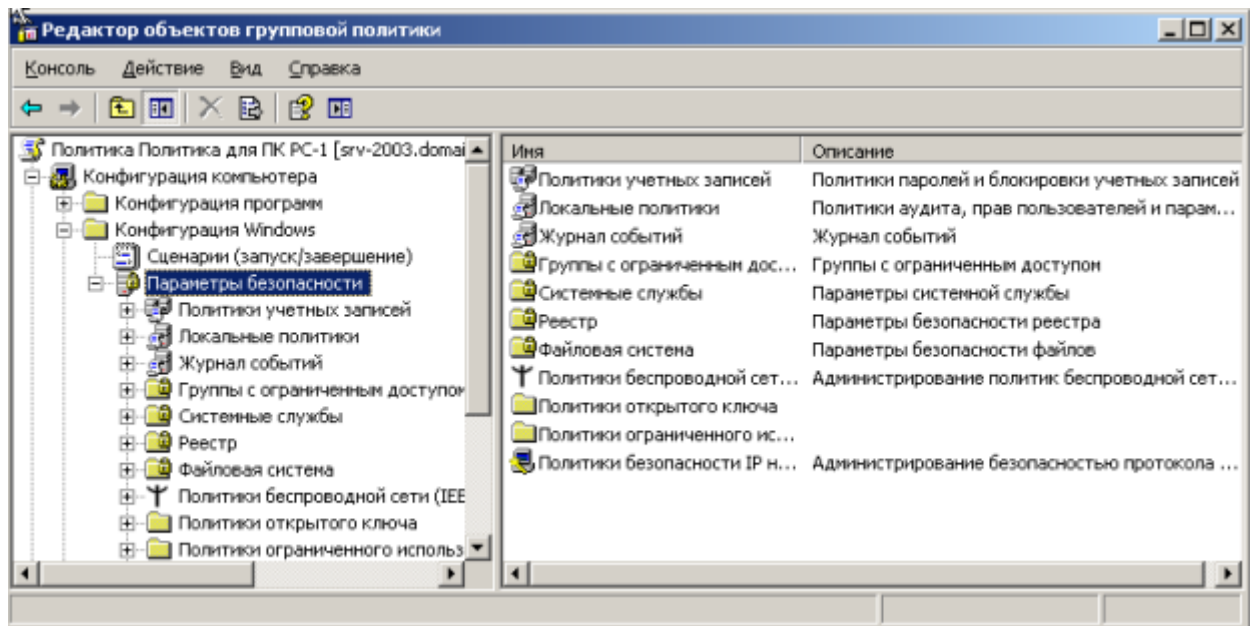


Рис. 15.17. Окно Редактор объектов групповой политики
 Далее ищем **Назначение прав пользователя-Локальный вход в систему** (рис. 15.18).

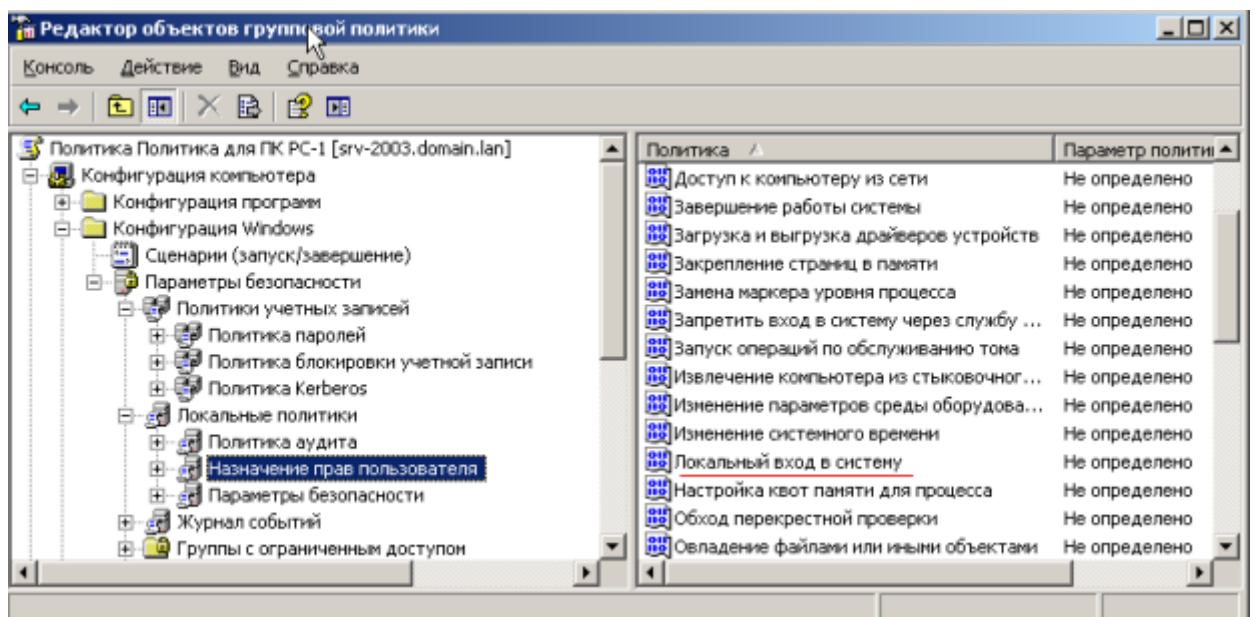


Рис. 15.18. Ищем запись Локальный вход в систему
 Далее активируем флажок, показанный на рис. 15.19.

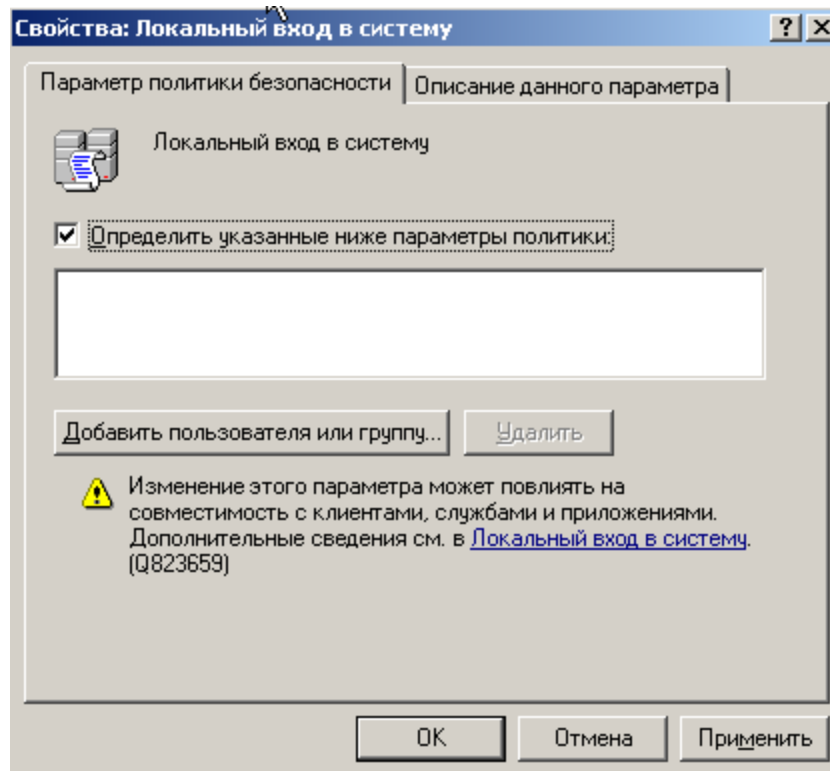


Рис. 15.19. Окно Свойства: Локальный вход в систему
Кнопкой **Добавить пользователя в группу** задаем пользователей, которым можно заходить в РС-1 (рис. 15.20).

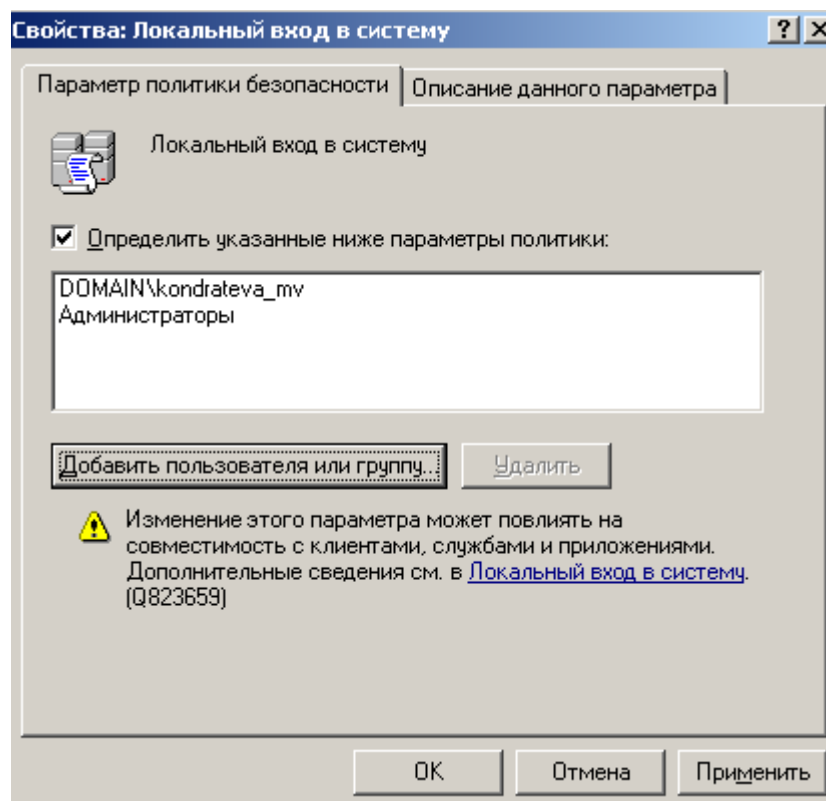


Рис. 15.20. Мы определили тех, кто может входить в РС-1
В РС-1 обязательно должны заходить администраторы, также мы разрешили вход пользователю Кондратьева М.В. Остальные, даже зная

правильный *пароль*, в этот ПК не войдут. В окне на рис. 15.21 мы можем указать, для каких ПК данную групповую политику можно применять.

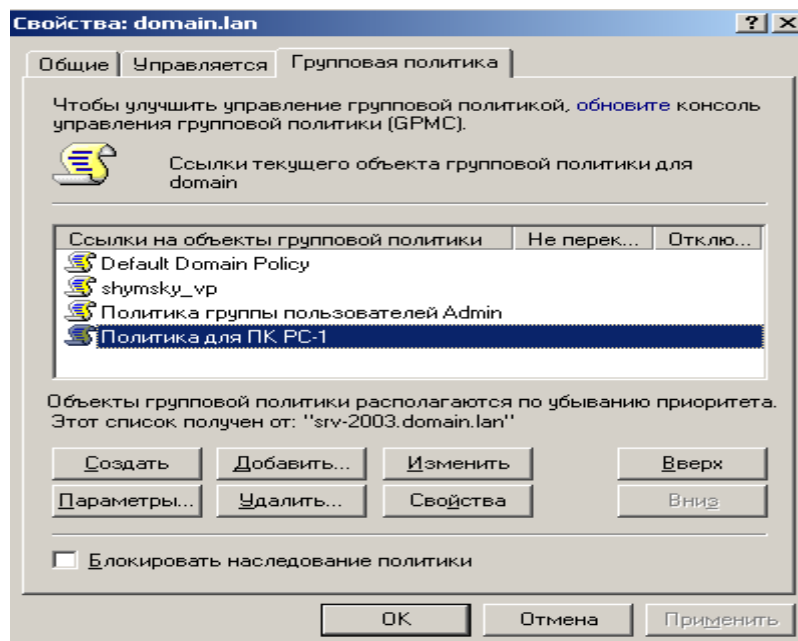


Рис. 15.21. Выделяем строчку Политика для ПК PC-1

Выполняем команды **Свойства-Безопасность** и **убираем** галочку **Применение групповой политики** (рис. 15.22).

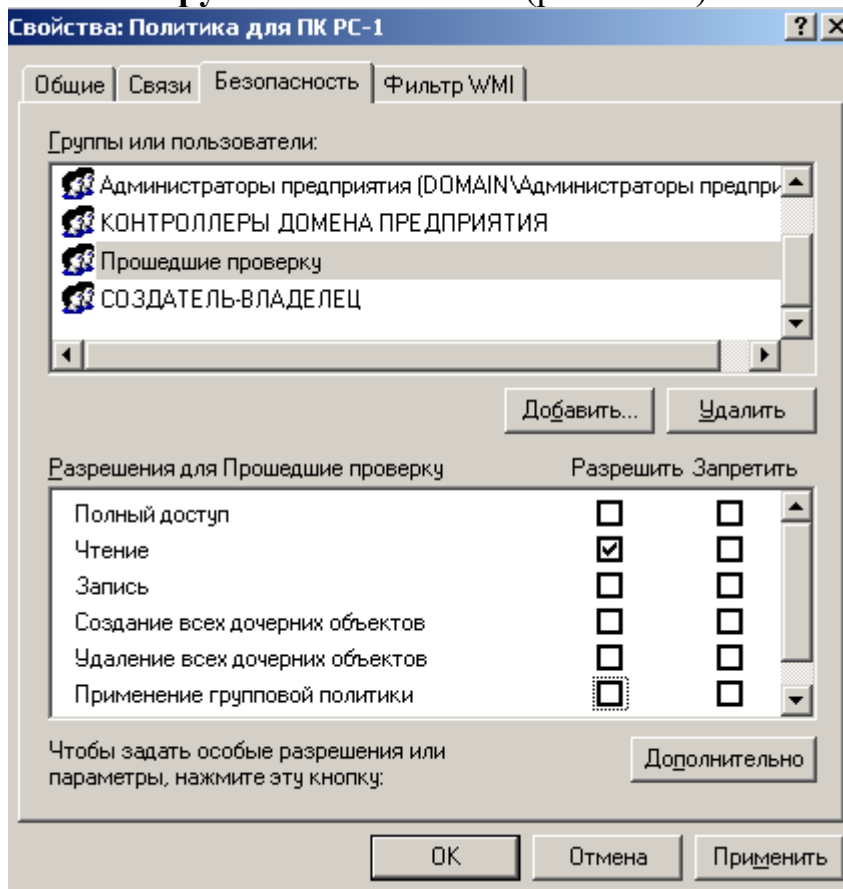


Рис. 15.22. Убираем галочку Применение групповой политики

Нажимаем **Применить**, затем – **Добавить**. Далее нажимаем на кнопку **Тип объекта** и ставим флажок **Компьютеры** (рис. 15.23).

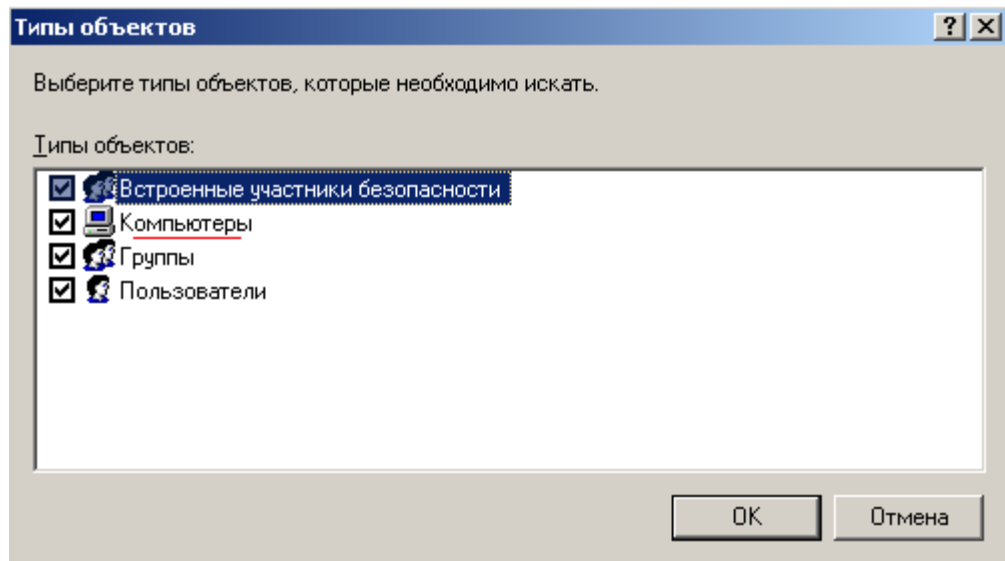


Рис. 15.23. Активируем флажок Компьютеры
Теперь кнопкой **Поиск** можно найти компьютеры из AD и задать для них сконфигурированную нами групповую политику (рис. 15.24).

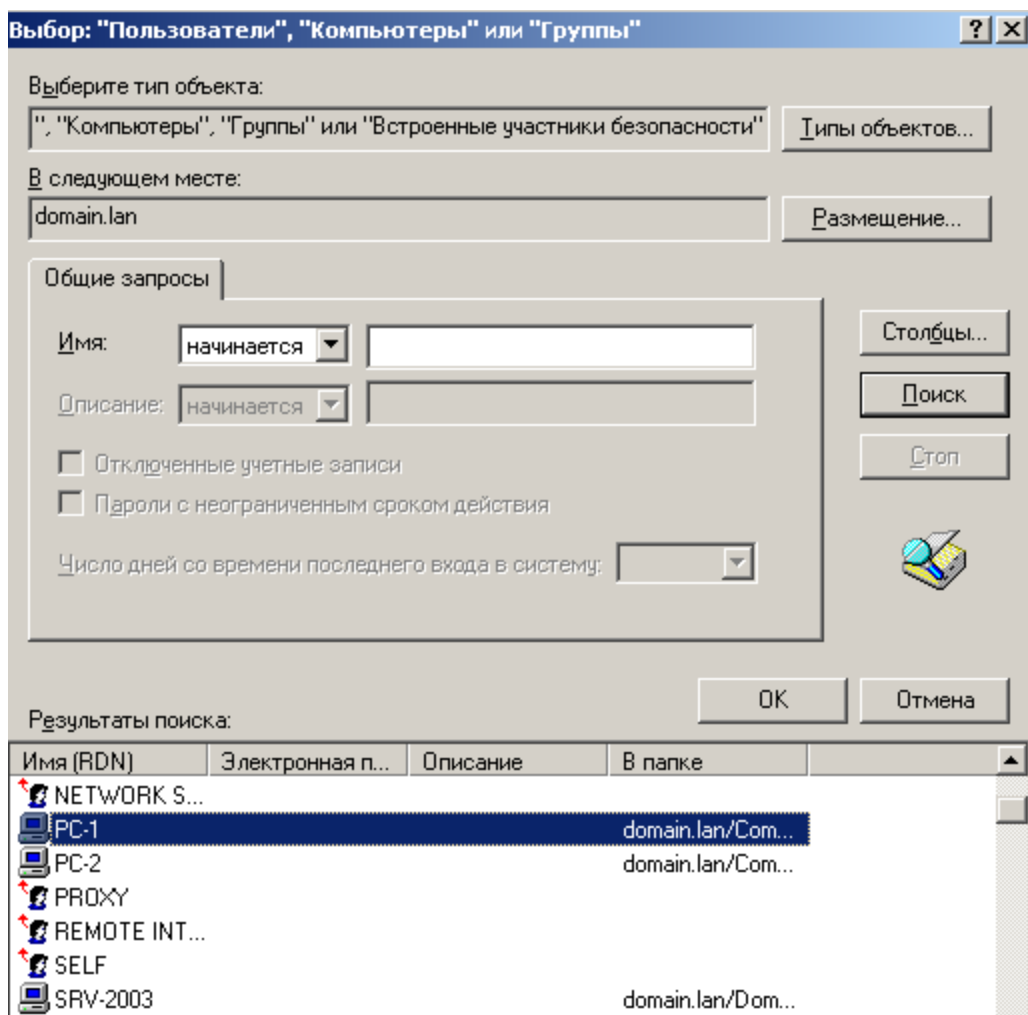


Рис. 15.24. Находим PC-1

Жмем ОК и устанавливаем флажок (рис. 15.25).

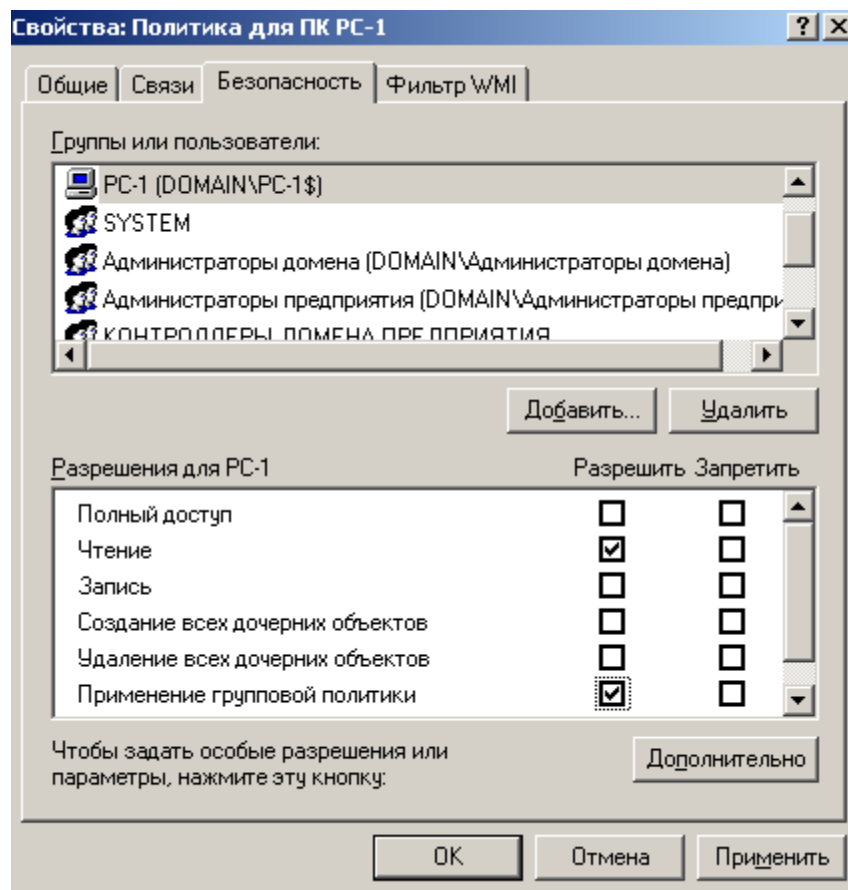


Рис. 15.25. Активируем для PC-1 флажок Применение групповой политики
Перезагрузите ПК и проверьте результат наших настроек групповой политики самостоятельно.

Краткие итоги

В этой работе мы сделали изменение групповой политики (ГП) для одного (конкретного) пользователя, а также создали групповую политику для группы пользователей. Также научились производить *конфигурирование* групповых политик для отдельных компьютеров. **Лабораторную работу дополняет скринкаст.**

Лабораторная работа №14

Архитектура ЭВМ и система команд

1. Общие положения

Для решения с помощью ЭВМ некоторой задачи должна быть разработана программа. Программа на языке ЭВМ представляет собой последовательность команд. Код каждой команды определяет выполняемую операцию, тип адресации и адрес. Выполнение программы, записанной в памяти ЭВМ, осуществляется последовательно по командам в порядке возрастания адресов команд или в порядке, определяемом командами передачи управления.

Для того чтобы получить результат выполнения программы, студент должен:

- ввести программу в память ЭВМ;
- определить, если это необходимо, содержимое ячеек ОЗУ и РОН, содержащих исходные данные, а также регистров IR и BR;
- установить в РС стартовый адрес программы;
- перевести модель в режим Работа.

Каждое из этих действий выполняется посредством интерфейса модели. Ввод программы может осуществляться как в машинных кодах непосредственно в память модели, так и в мнемокодах в **окно Текст программы** с последующим ассемблированием.

Цель настоящей лабораторной работы — знакомство с интерфейсом модели ЭВМ, методами ввода и отладки программы, действиями основных классов команд и способов адресации. Для этого необходимо ввести в память ЭВМ и выполнить в режиме Шаг некоторую последовательность команд (определенную вариантом задания) и зафиксировать все изменения на уровне программно-доступных объектов ЭВМ, происходящие при выполнении этих команд.

Команды в память учебной ЭВМ вводятся в виде шестизначных десятичных чисел (см. форматы команд на рис. 3, коды команд и способов адресации в табл. 2—4).

В настоящей лабораторной работе будем программировать ЭВМ в машинных кодах.

2. Пример 1

Дана последовательность мнемокодов, которую необходимо преобразовать в машинные коды, занести в ОЗУ ЭВМ, выполнить в режиме Шаг и зафиксировать изменение состояний программно-доступных объектов ЭВМ (табл. 1).

Таблица 9.1. Команды и коды

Последовательность	Значения				
Команды	RD#20	WR30	ADD #5	WR@30	JNZ 002
Коды	21 1 020	22 0 030	23 1 005	22 2 030	12 0 002

Введем полученные коды последовательно в ячейки ОЗУ, начиная с адреса 000. Выполняя команды в режиме Шаг, будем фиксировать изменения программно-доступных объектов (в данном случае это Асс, РС и ячейки ОЗУ 020 и 030) в табл.2.

Таблица 9.2. Содержимое регистров

РС	Асс	М(30)	М(20)	РС	Асс	М(30)	М(20)
000	000000	000000	000000	004			000025
001	000020			002			
002		000020		003	000030		
003	000025			004			000030

3. Задание 1

1. Ознакомиться с архитектурой ЭВМ (см. часть I).
2. Записать в ОЗУ "программу", состоящую из пяти команд — варианты

задания выбрать из табл. 3. Команды разместить в последовательных ячейках памяти.

3. При необходимости установить начальное значение в устройство ввода IR.

4. Определить те программно-доступные объекты ЭВМ, которые будут изменяться при выполнении этих команд.

5. Выполнить в режиме Шаг введенную последовательность команд, фиксируя изменения значений объектов, определенных в п. 4, в таблице (см. форму табл. 2).

6. Если в программе образуется цикл, необходимо просмотреть не более двух повторений каждой команды, входящей в тело цикла.

Таблица 9.3. Варианты задания 1

№	IR	Команда 1	Команда 2	Команда 3	Команда 4	Команда 5
1	000007	IN	MUL #2	WR10	WR @10	JNS 001
2	X	RD #17	SUB #9	WR16	WR @16	JNS 001
3	100029	IN	ADD #16	WR8	WR@8	JS 001
4	X	RD #2	MUL #6	WR 11	WR @11	JNZ 00
5	000016	IN	WR8	DIV #4	WR @8	JMP 002
6	X	RD #4	WR 11	RD @11	ADD #330	JS 000
7	000000	IN	WR9	RD @9	SUB#1	JS 001
8	X	RD 4	SUB #8	WR8	WR @8	JNZ .J1
9	100005	IN	ADD #12	WR 10	WR @10	JS 004
10	X	RD 4	ADD #15	WR 13	WR @13	JMP 001
11	000315	IN	SUB #308	WR11	WR @11	JMP 001
12	X	RD #988	ADD #19	WR9	WR @9	JNZ 001
13	000017	IN	WR11	ADD 11	WR @11	JMP 002
14	X	RD #5	MUL #9	WR10	WR @10	JNZ 001

4. Содержание отчета

1. Формулировка варианта задания.
2. Машинные коды команд, соответствующих варианту задания.
3. Результаты выполнения последовательности команд в форме табл. 2.

5. Контрольные вопросы

1. Из каких основных частей состоит ЭВМ и какие из них представлены в модели?
2. Что такое система команд ЭВМ?
3. Какие классы команд представлены в модели?
4. Какие действия выполняют команды передачи управления?
5. Какие способы адресации использованы в модели ЭВМ? В чем отличие между ними?
6. Какие ограничения накладываются на способ представления данных в модели ЭВМ?
7. Какие режимы работы предусмотрены в модели и в чем отличие между ними?
8. Как записать программу в машинных кодах в память модели ЭВМ?

9. Как посмотреть содержимое регистров процессора и изменить содержимое некоторых регистров?

10. Как посмотреть и, при необходимости, отредактировать содержимое ячейки памяти?

11. Как запустить выполнение программы в режиме приостановки работы после выполнения каждой команды?

12. Какие способы адресации операндов применяются в командах ЭВМ?

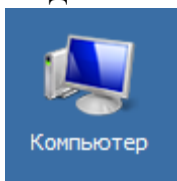
13. Какие команды относятся к классу передачи управления?

Лабораторная работа №15

Диагностика IP протокола

Применение команды Ping для проверки наличия связи компьютеров в сети

Наиболее быстрым способом проверки работоспособности локальной можно назвать системную команду *PING*, которая посылает сетевой *запрос* на заданный *IP-адрес* компьютера, получает ответ и выводит отчет на экран. Если посланный *запрос* получен обратно - *связь* физически существует, то ваша *сеть* настроена и работает корректно. Если же на экране вы увидите надпись "Превышен *интервал* ожидания *запрос*" - вы допустили ошибку либо в настройках, либо в подключении компьютеров. Перед запуском команды *Ping* необходимо посмотреть доступные компьютеры в



сети. Заходим в **Компьютер** и видим, что в нашей рабочей группе 110 имеется четыре ПК (рис. 8.1).

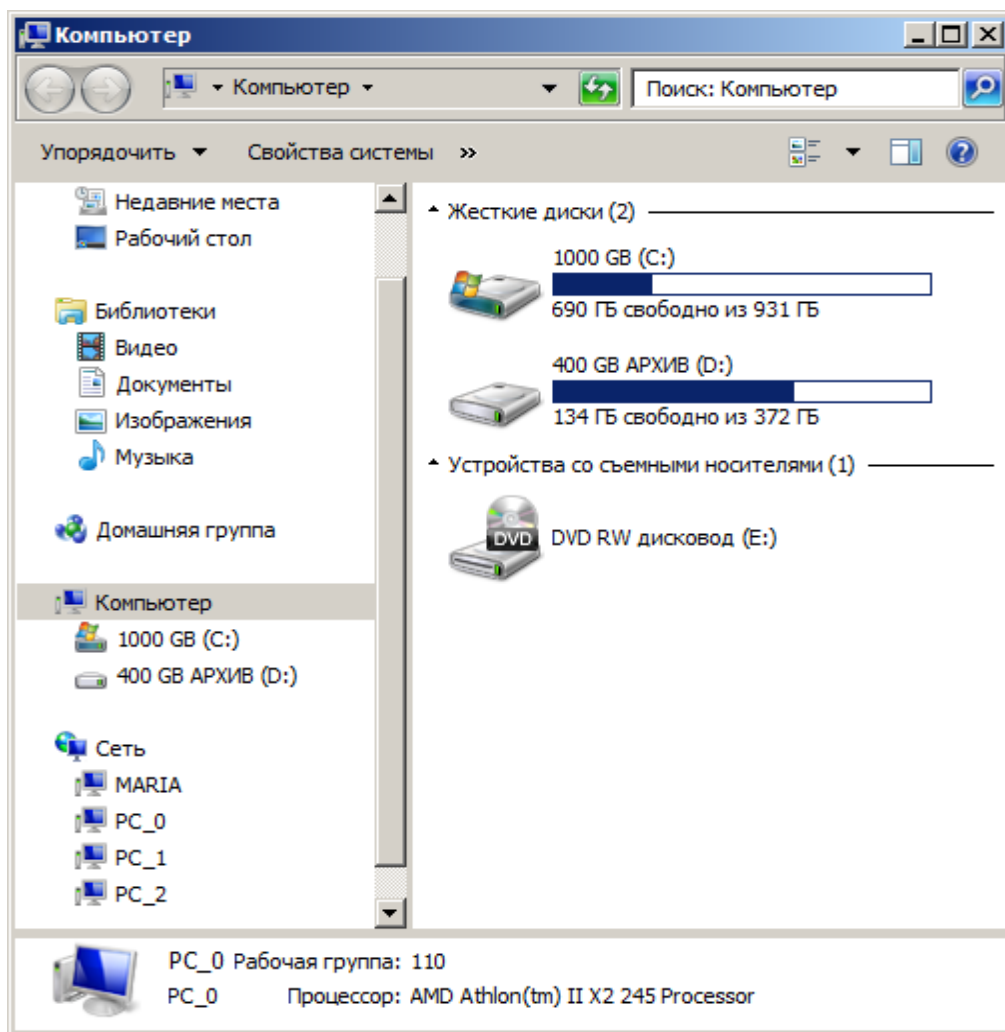


Рис. 8.1. В рабочей группе 110 мы видим 4 ПК

Для того чтобы воспользоваться командой *ping*, откройте окно командной строки командой **Пуск-Все программы-Стандартные-Командная строка** и введите там команду *ping*, укажите имя или *IP-адрес* удаленного компьютера (или его *ИМЯ"/>*) (рис. 8.2). По умолчанию утилита *ping* отправляет 4 пакета и ожидает каждый ответ в течение четырех секунд. По умолчанию команда посылает пакет 32 байта. За размером тестового пакета отображается время отклика удаленной системы (в нашем случае — меньше 1 миллисекунды"/>).

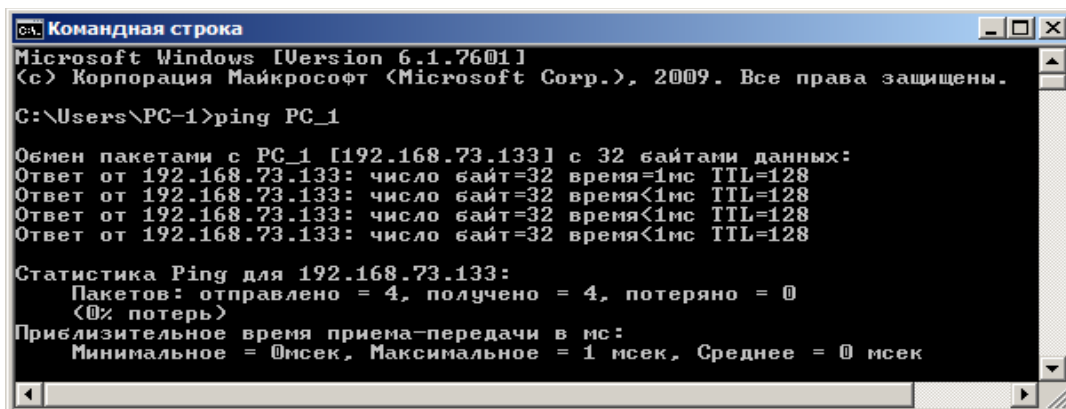


Рис. 8.2. Пингование машины PC_1 с IP-адресом 192.168.73.133

При необходимости для этой команды вы можете использовать следующие параметры:

-t. Данный *параметр* указывает на то, что производится проверка связи с указанным узлом до прекращения вручную;

-n. Текущий *параметр* определяет количество отправляемых Echo-запросов;

-f. Этот *параметр* устанавливает бит "не фрагментировать" на *ping*-пакете. По умолчанию фрагментация разрешается;

-w. Данный *параметр* позволяет настроить тайм-аут для каждого пакета в миллисекундах (по умолчанию установлено значение 4000"/>);

-a. Текущий *параметр* определяет имена узлов по адресам;

-l. При помощи этого параметра вы можете указать размер буфера отправки;

-i. Использование данного параметра позволяет вам задать срок жизни пакета;

-v. Этот *параметр* задает тип службы для IPv4 и не влияет на поле TOS в IP-заголовке;

-r. Текущий *параметр* записывает маршрут для указанного числа прыжков;

-s. Данный *параметр* позволяет отмечать время для указанного числа прыжков;

-j. Используя этот *параметр*, вы можете указать свободный выбор маршрута по списку узлов;

-k. При помощи данного параметра вы можете определить жесткий выбор маршрута по списку узлов;

-R. Текущий *параметр* позволяет использовать заголовок для проверки также и обратного маршрута только для IPv6;

-S. Данный *параметр* указывает используемый адрес источника;

-4. *Параметр* определяет принудительное использование протокола IP версии 4;

-6. *Параметр* определяет принудительное использование протокола IP версии 6.

Итак, выше было показано, что *утилита Ping* используется в том случае, когда необходимо проверить, может ли компьютер подключиться к сети TCP/IP или сетевым ресурсам. Иначе говоря, мы пингуем для того, чтобы проверить, что отправляемые пакеты доходят до получателя. ПК-отправитель отправляет Echo-запрос, а ПК-получатель, в ответ должен отправить ICMP-сообщение с ответом. Если удаленный компьютер реагирует на запрос ping, то подключение к удаленному компьютеру работает. Также, *утилита ping* ведет статистику, из которой понятно, сколько пакетов получено, а сколько потеряно. Но, это еще не все.

Применение команды Ping для анализа качества связи ПК в сети

Для тестирования качества связи запустите *Ping* со следующими параметрами: **ping.exe -l 16384 -w 500 -n 100 192.168.73.133**. Это обеспечит

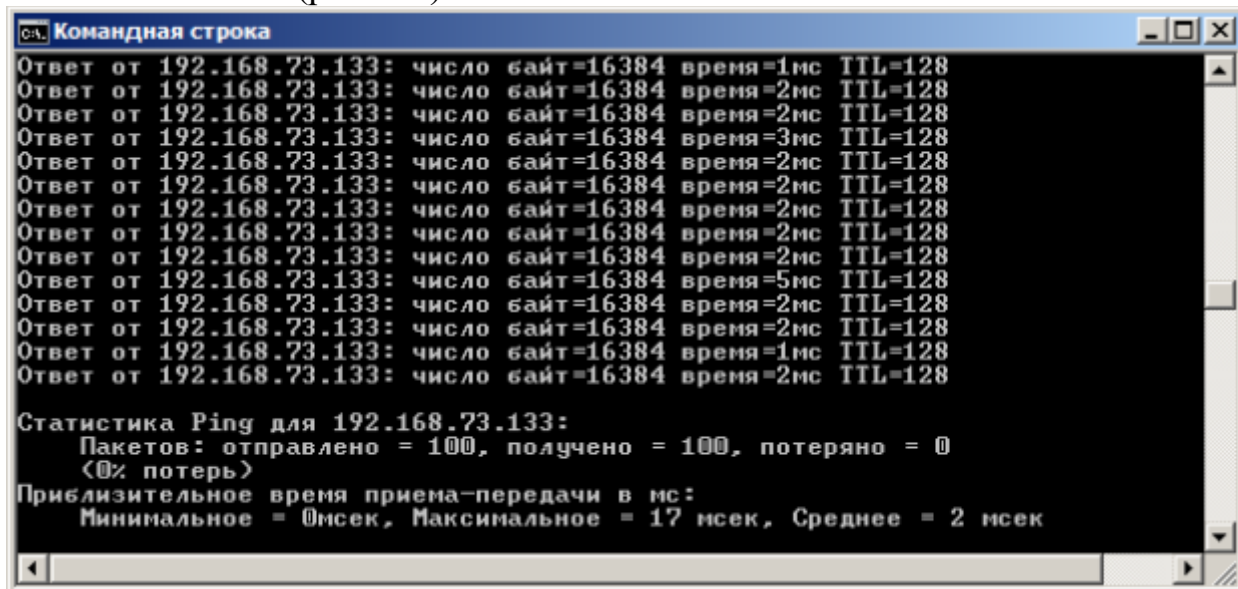
отправку 100 запросов (n) пакетами по 16 килобайт (l) на заданный IP адрес с интервалом ожидания ответа в 0,5 секунды (w). То есть:

L – размер буфера отправки.

N – число отправляемых запросов,

W – время ожидания ответа на запрос в миллисекундах,

Подождите, пока пройдут все 100 пакетов. Ответ должен будет быть приблизительно такой (рис. 8.3).



```
Сл. Командная строка
Ответ от 192.168.73.133: число байт=16384 время=1мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=3мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=5мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=1мс TTL=128
Ответ от 192.168.73.133: число байт=16384 время=2мс TTL=128

Статистика Ping для 192.168.73.133:
  Пакетов: отправлено = 100, получено = 100, потеряно = 0
  (0% потерь)
Приблизительное время приема-передачи в мс:
  Минимальное = 0мсек, Максимальное = 17 мсек, Среднее = 2 мсек
```

Рис. 8.3. Ответ на команду ping.exe с ключами

Проанализируем результат выполнения команды:

- 0% потерь – сеть работает отлично.
- Если потери информации составили не более 3%, то сеть работает хорошо.
- При потерях 3-10% дошли не все пакеты, но сеть, благодаря алгоритмам коррекции ошибок, работает удовлетворительно. Из-за необходимости повторной доставки потерянной информации снижается эффективная скорости работы сети – сеть тормозит.
- Если число потерянных пакетов превышает 10-15%, то необходимо принять меры по устранению неисправности. Качество связи ПК неудовлетворительное.

Далее: как видим, время отклика удаленной системы среднее 2 мсек, а максимальное 17 мсек. Анализируя отклик по миллисекундам, надо иметь ввиду следующее. По стандарту, нормальное время отклика 16-килобайтного пакета для 100-мегабитной сети - 3-8 мс. Для 10-мегабитной - 30-80 мс. Получается, что у нас сеть работает на скорости порядка 100 мбит/сек.

Использование утилиты PathPing

Pathping это утилита, которая позволяет обнаружить потери пакетов на маршруте между вашим компьютером и заданным адресом IP. Потери пакетов могут сильно повлиять на работу сети, например, когда вы играете в видеоигру. Иначе говоря, утилита PathPing отправляет многочисленные сообщения с Echo-запросом каждому маршрутизатору, который находится между исходным пунктом и пунктом назначения, после чего, на основании

пакетов, полученных от каждого из них, вычисляет процентное соотношение пакетов, возвращаемых в каждом прыжке. Поскольку утилита PathPing показывает степень потери пакетов на каждом маршрутизаторе или узле, то с ее помощью вы можете точно определить маршрутизаторы и узлы, на которых возникают сетевые проблемы. Пример использования данной команды приведен на рис. 4.

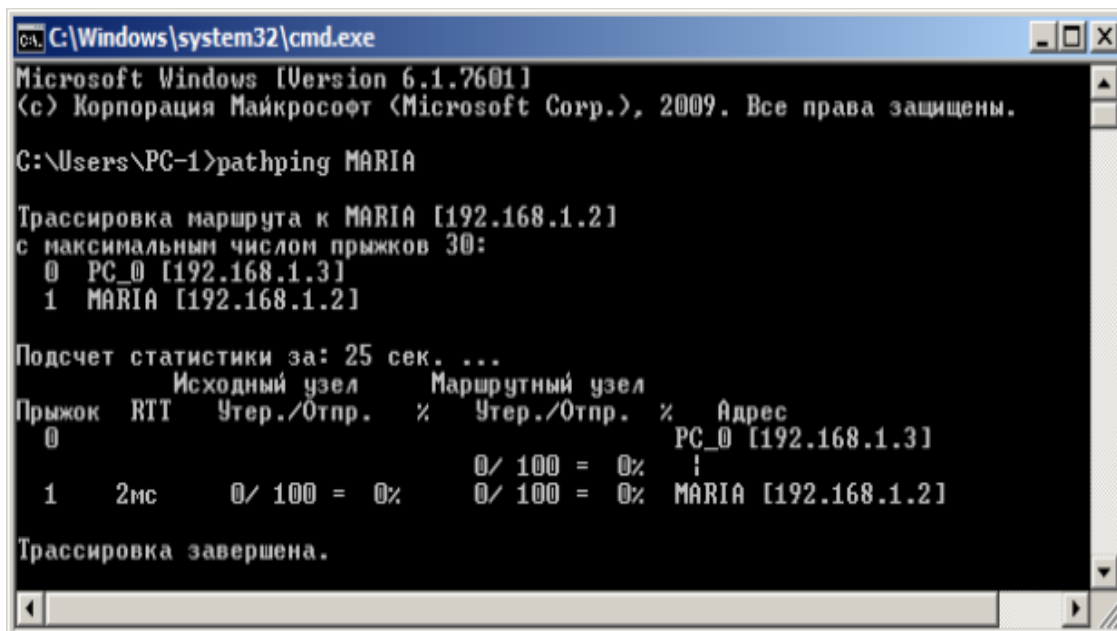


Рис. 8.4. Поиск потерь пакетов на маршруте от ПК PC_0 до ПК MARIA. Итак, в строке поиска наберем **CMD**, чтобы вызвать командную строку (рис. 8.5).

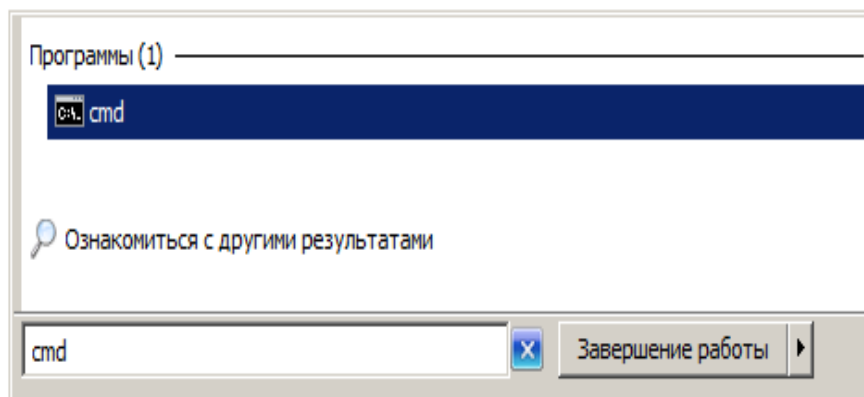


Рис. 8.5. Один из способов вызова командной строки в ОС Windows 7. Далее произведет трассировку маршрута от нашего ПК до поискового сервера Яндекс (рис. 8.6).

```

C:\Windows\system32\cmd.exe
C:\Users\PC-1>pathping yandex.ru
Трассировка маршрута к yandex.ru [213.180.204.11]
с максимальным числом прыжков 30:
 0 PC_0 [192.168.1.3]
 1 192.168.1.1
 2 lo0-at66-2.natm.ru [213.148.173.214]
 3 at66-ats66-L3-giga-core.natm.ru [213.148.163.81]
 4 ATS3-TGE1-8-TTS-TGE1-4.natm.ru [78.81.0.37]
 5 GWay-TGE0-2.natm.ru [78.81.0.254]
 6 ge-0-1-0-v1988-10g.M320-1-NOUG.nwtelecom.ru [212.48.214.53]
 7 ae1-30g.MX960-1-MMT.nwtelecom.ru [212.48.198.246]
 8 as13238-yandex.gateway.nwtelecom.ru [212.48.214.102]
 9 * * *
Подсчет статистики за: 200 сек. ...
Исходный узел Маршрутный узел
Прыжок RTT Утер./Отпр. % Утер./Отпр. % Адрес
 0 PC_0 [192.168.1.3]
 1 1мс 0/100 = 0% 0/100 = 0% 192.168.1.1
 2 1мс 0/100 = 0% 0/100 = 0% lo0-at66-2.natm.ru [213.148.1
4]
 3 2мс 0/100 = 0% 0/100 = 0% at66-ats66-L3-giga-core.natm.
13.148.163.81]
 4 1мс 0/100 = 0% 0/100 = 0% ATS3-TGE1-8-TTS-TGE1-4.natm.r
.81.0.37]
 5 3мс 0/100 = 0% 0/100 = 0% GWay-TGE0-2.natm.ru [78.81.0.
6]
 6 2мс 0/100 = 0% 0/100 = 0% ge-0-1-0-v1988-10g.M320-1-NOU
elecom.ru [212.48.214.53]
 7 11мс 0/100 = 0% 0/100 = 0% ae1-30g.MX960-1-MMT.nwtelecom
212.48.198.246]
 8 --- 100/100 =100% 0/100 = 0% as13238-yandex.gateway.nwtele
u [212.48.214.102]
Трассировка завершена.

```

Рис. 8.6. Пример использования утилиты Pathping

Проанализируем результат:

- Первый блок информации представляет собой трассировку. Вы можете пропустить его и перейти ко второму блоку информации, в котором будет указано процентное отношение потерь пакетов.

- Если пакеты не терялись на данном маршруте подключения, то вы увидите 0% потерь пакетов. Если вы увидите значения, отличающиеся от 0%, это означает, что на пути к нашим серверам были потери пакетов. Потери выше 1% начиная с первого шага, могут указывать на некорректную работу узлов сети или маршрутизаторов. Если эти устройства вам доступны, то нужно попробовать обновить их программное обеспечение или полностью заменить их. Иначе, о потерях, возникших после первого шага и до последнего шага, следует сообщить вашему Интернет провайдеру.

Примечание

Если последние строки указывают на 100% потерь, то это не является показателем проблемы, а происходит потому, что сервера защищены от нежелательного трафика и атак.

С данной командой вы можете использовать следующие параметры:

- g. Данный *параметр* определяет использование параметра свободной маршрутизации в IP-заголовке с набором промежуточных мест назначения для сообщений с Echo-запросом, который указывается в списке компьютеров.

- h. Данный *параметр* задает максимальное количество переходов на пути при поиске конечного объекта;

- i. Этот *параметр* указывает *IP-адрес* источника;
- n. Текущий *параметр* предотвращает попытки сопоставления *IP-адресов* промежуточных маршрутизаторов с их именами, что существенно ускоряет *вывод* результатов;
- r. Используя данный *параметр*, вы можете задать *время ожидания* между последовательными проверками связи, где значением *по умолчанию* указано 250 миллисекунд;
- q. При помощи текущего параметра вы можете указать количество сообщений с Echo-запросом, отправленных каждому маршрутизатору пути (*по умолчанию* - 100);
- w. Данный *параметр* определяет *время ожидания* для получения Echo-ответов протокола *ICMP* или *ICMP-сообщений* об истечении времени в миллисекундах, которые соответствуют данному сообщению Echo-запроса. *Значение по умолчанию* 4 секунды;
- 4. *Параметр* определяет принудительное использование протокола *IP* версии 4;
- 6. *Параметр* определяет принудительное использование протокола *IP* версии 6.

Другие команды командной строки. Отображение параметров *TCP/IP*-протокола командой `Ipconfig`

Команда **IPCONFIG** используется для отображения текущих настроек протокола *TCP/IP* и для обновления некоторых параметров, задаваемых при автоматическом конфигурировании сетевых интерфейсов при использовании протокола *DHCP*. Предположим, что у нас имеется *сеть*, изображенная на рис. 8.7.

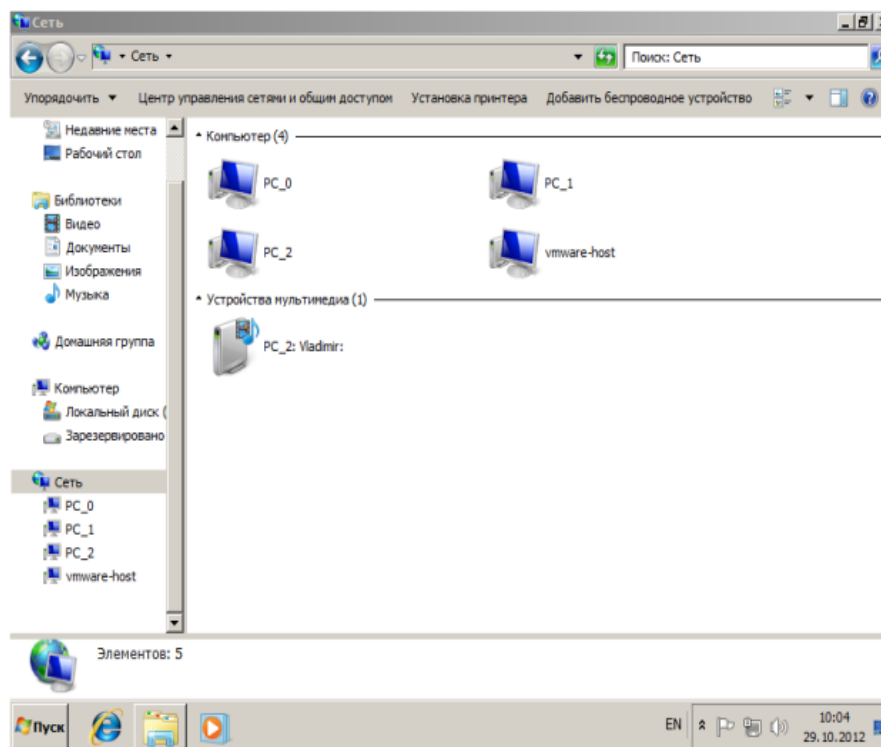


Рис. 8.7. Небольшая локальная сеть
Выполним команду командой `Ipconfig` на *PC_2* (рис. 8.8).

```

Администратор: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\Uladimir>ipconfig

Настройка протокола IP для Windows

Ethernet adapter Подключение по локальной сети:

    DNS-суффикс подключения . . . . . : localdomain
    Локальный IPv6-адрес канала . . . . : fe80::1170:c16a:c226:283c%11
    IPv4-адрес . . . . . : 192.168.73.133
    Маска подсети . . . . . : 255.255.255.0
    Основной шлюз . . . . . : 192.168.73.2

Туннельный адаптер isatap.localdomain:

    Состояние среды . . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . : localdomain

Туннельный адаптер Подключение по локальной сети*:

    DNS-суффикс подключения . . . . . :
    IPv6-адрес . . . . . : 2001::0:5ef5:79fd:2437:299d:3f57:b67a
    Локальный IPv6-адрес канала . . . . : fe80::2437:299d:3f57:b67a%13
    Основной шлюз . . . . . :

```

Рис. 8.8. Отображение параметров TCP/IP-протокола командой Ipconfig
Из отчета мы видим такую информацию:

- DNS-суффикс подключения - localdomain (из настроек сетевого подключения)
- Локальный IPv6-адрес канала - локальный IPv6 адрес, если используется адресация IPv6
- IPv4-адрес - используемый для данного адаптера IPv4 – адрес
- Маска подсети - 255.255.255.0
- Основной шлюз - IP-адрес маршрутизатора, используемого в качестве шлюза по умолчанию.

Примечание

Туннельный адаптер isatap.localdomain это эмуляция IPV6 в сетях IPV4. ISATAP (Intra-Site Automatic Tunnel Addressing Protocol) — Протокол автоматической внутрисайтовой адресации туннелей, позволяющий передавать между сетями IPv6 пакеты через сети IPv4

Ключи команды:

/all *Отображение* полной информации по всем адаптерам.

/release [адаптер] *Отправка сообщения DHCPRELEASE серверу DHCP для освобождения текущей конфигурации DHCP и удаления конфигурации IP-адресов для всех адаптеров (если адаптер не задан) или для заданного адаптера. Этот ключ отключает протокол TCP/IP для адаптеров, настроенных для автоматического получения IP-адресов.*

/renew [адаптер] *Обновление IP-адреса для определённого адаптера или если адаптер не задан, то для всех. Доступно только при настроенном автоматическом получении IP-адресов.*

/flushdns *Очищение DNS кэша.*

/registerdns *Обновление всех зарезервированных адресов DHCP и перерегистрация имен DNS.*

/displaydns *Отображение содержимого кэша DNS.*

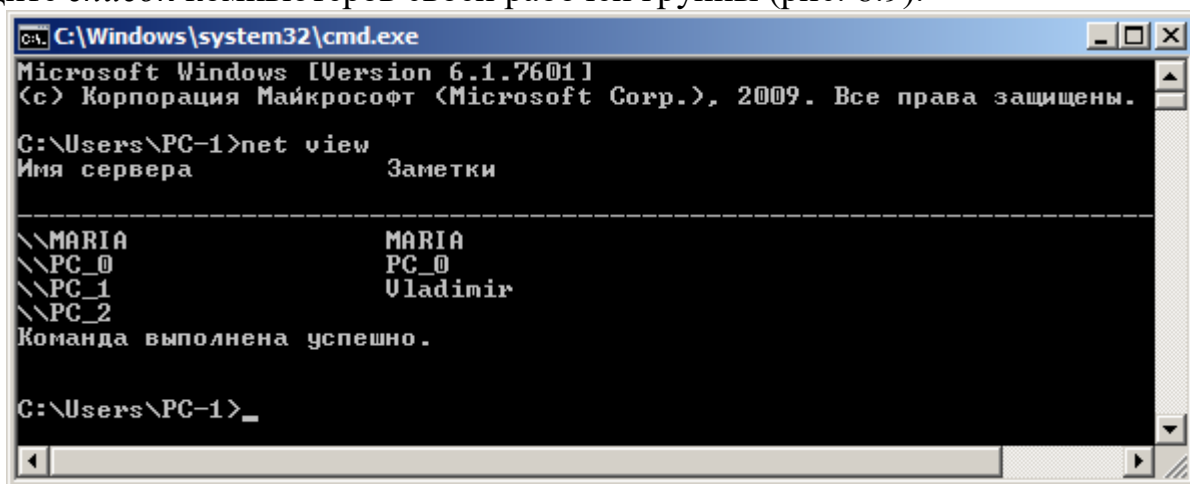
`/showclassid адаптер` Отображение кода класса *DHCP* для указанного адаптера. Доступно только при настроенном автоматическим получением *IP*-адресов.

`/setclassid адаптер [код_класса]` Изменение кода класса *DHCP*. Доступно только при настроенном автоматическим получением *IP*-адресов.

`/?` Справка. *TCP/IP*: значения *IP* адреса, маски и шлюза.

Команда вывода списка компьютеров рабочей группы Net view

В командной строке введите команду **net view**, и вы увидите список компьютеров своей рабочей группы (рис. 8.9).



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

C:\Users\PC-1>net view
Имя сервера          Заметки
-----
\\MARIA                MARIA
\\PC_0                 PC_0
\\PC_1                 Vladimir
\\PC_2
Команда выполнена успешно.

C:\Users\PC-1>
```

Рис. 8.9. В рабочей группе имеется 4 ПК

Трассировка

Tracert — это служебная компьютерная программа, предназначенная для определения маршрутов следования данных в сетях *TCP/IP*. Программа *tracert* выполняет отправку данных указанному узлу сети, при этом отображая сведения о всех промежуточных маршрутизаторах, через которые прошли данные на пути к целевому узлу. В случае проблем при доставке данных до какого-либо узла программа позволяет определить, на каком именно участке сети возникли неполадки.

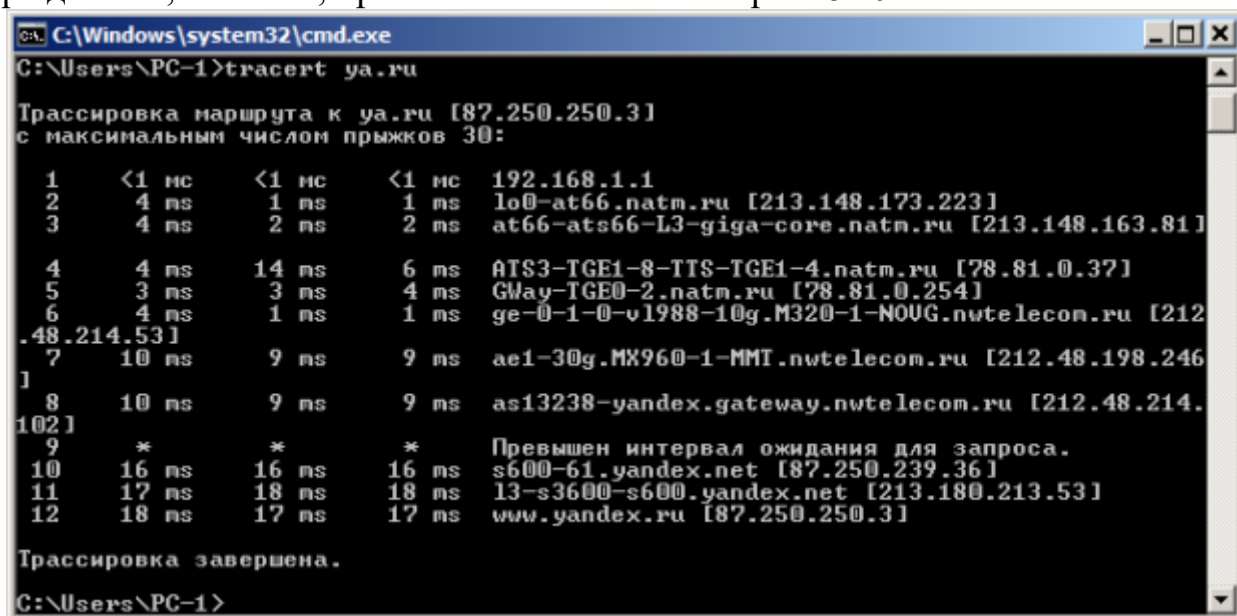
Запуск программы производится из командной строки. Для этого вы должны войти в неё. Для операционной системы *Windows 7* существует несколько способов запуска командной строки:

3. Сочетание клавиш Win (кнопка с логотипом Windows) + R (должны быть нажаты одновременно) — В графе "Открыть" написать "cmd" и нажать Ок.

4. Пуск — Все программы — Стандартные — Командная строка.

В открывшемся окне мы напишем **tracert ya.ru**. Принцип действия этой программы схож с принципом действия программы *ping*. Команда отправляет на сервер данные и при этом фиксирует все промежуточные маршрутизаторы, через которые проходят эти данные на пути к серверу (целевому узлу). Если при доставке данных до одного из узлов происходит проблема, программа определяет участок сети, на котором возникли неполадки. Время отклика показывает загруженность канала. А вот

если вместо времени отклика вы видите надпись "**Превышен интервал ожидания для запроса**", это значит, что на данном узле связи происходит потеря данных, а значит, проблема именно в нем – рис. 8.10.



```

C:\Windows\system32\cmd.exe
C:\Users\PC-1>tracert ya.ru

Трассировка маршрута к ya.ru [87.250.250.3]
с максимальным числом прыжков 30:

  1    <1 ms    <1 ms    <1 ms    192.168.1.1
  2     4 ms     1 ms     1 ms     lo0-at66.natm.ru [213.148.173.223]
  3     4 ms     2 ms     2 ms     at66-ats66-L3-giga-core.natm.ru [213.148.163.81]

  4     4 ms    14 ms     6 ms     AT$3-TGE1-8-TTS-TGE1-4.natm.ru [78.81.0.37]
  5     3 ms     3 ms     4 ms     GWay-TGE0-2.natm.ru [78.81.0.254]
  6     4 ms     1 ms     1 ms     ge-0-1-0-v1988-10g.M320-1-NOUG.nwtelecom.ru [212.48.214.53]
  7    10 ms     9 ms     9 ms     ae1-30g.MX960-1-MMT.nwtelecom.ru [212.48.198.246]
  8    10 ms     9 ms     9 ms     as13238-yandex.gateway.nwtelecom.ru [212.48.214.102]
  9     *        *        *        Превышен интервал ожидания для запроса.
 10    16 ms    16 ms    16 ms    s600-61.yandex.net [87.250.239.36]
 11    17 ms    18 ms    18 ms    13-s3600-s600.yandex.net [213.180.213.53]
 12    18 ms    17 ms    17 ms    www.yandex.ru [87.250.250.3]

Трассировка завершена.
C:\Users\PC-1>

```

Рис. 8.10. Пример трассировки домена ya.ru

Параметры команды tracert:

-d не определять доменные имена маршрутизаторов

-h <значение> установить максимальное количество переходов

-w <значение> установить максимальное время ожидания ответа (в миллисекундах)

Итак, трассировка маршрута помогает определить проблемный узел. Если данные проходят нормально и "стопорятся" на самом пункте назначения, то проблема действительно с сайтом. Если трассировка маршрута прекращается на середине пути, то проблема в одном из промежуточных маршрутизаторов. Если прохождение пакетов прекращается в пределах сети вашего провайдера — то и проблему нужно решать "на местном уровне". Попутно хочется отметить, что программа работает только в направлении от источника пакетов и является весьма грубым инструментом для выявления неполадок в сети.

Краткие итоги

В лабораторной работе мы рассмотрели применение команды Ping для проверки наличия связи компьютеров в сети и для анализа качества связи ПК, научились пользоваться командами PathPing, Ipconfig, Net view и Tracert. Работу дополняет скринкаст.

В готовой лабораторной работе, оформить скриншоты команд: Ping, PathPing, Ipconfig, Net view и Tracert.

Лабораторная работа №16

Исследование работы АЛУ и устройства управления процессора

Введение

Во многих случаях знание операторов языка высокого уровня, структуры данных и способов их обработки является достаточным для создания различных полезных приложений. Однако по-настоящему решать проблемы, связанные с управлением различной, особенно нестандартной, аппаратурой (программирование „по железу“) невозможно без знания ассемблера. Не случайно практически все компиляторы языков высокого уровня содержат средства связи своих модулей с модулями на ассемблере либо поддерживают выход на ассемблерный уровень программирования.

Однако проводить начальное обучение программированию на низком уровне с рассмотрением механизмов взаимодействия устройств на реальном языке, например x86 на персональной ЭВМ, не всегда удобно. В этом случае между пользователем и аппаратурой ЭВМ присутствует операционная система (ОС), которая существенно ограничивает желания пользователя экспериментировать с аппаратными средствами. Для преодоления этих ограничений необходимо обладать глубокими знаниями как ОС, так и аппаратных средств ЭВМ.

Используемая программная модель учебной ЭВМ отражает все основные особенности систем команд и структур современных простых ЭВМ, включает в себя, помимо процессора и памяти, модели нескольких типичных внешних устройств. Модель позволяет изучить основы программирования на низком уровне, вопросы взаимодействия различных уровней памяти в составе ЭВМ и способы взаимодействия процессора с внешними устройствами.

Задание

Архитектура ЭВМ и система команд

Цель работы – знакомство с интерфейсом модели ЭВМ, методами ввода и отладки программы, действиями основных классов команд и способов адресации.

Теоретические положения:

Модель содержит процессор, оперативную (ОЗУ) и сверхоперативную память, устройство ввода и устройство вывода.

Процессор состоит из устройства управления, арифметико-логического устройства (АЛУ), десяти регистров общего назначения (РОН) и системных регистров (CR, PC, SP и др.). Доступ ко всем регистрам и флагам процессора обеспечивается через окно Процессор.

Регистры Асс, DR, IR, OR, CR и все ячейки ОЗУ и РОН имеют длину 6 десятичных разрядов, регистры PC, SP, RA и RB – 3 разряда. В окне Процессор отражаются текущие значения регистров и флагов, причем в состоянии Останов все регистры, включая регистры блока РОН, и флаги (кроме флага I) доступны для непосредственного редактирования.

Сверхоперативная память с прямой адресацией содержит десять регистров общего назначения R0-R9. Доступ к ним осуществляется через регистры RAR и RDR.

АЛУ выполняет одну из арифметических операций, определяемую кодом операции (COP), над содержимым аккумулятора (Асс) и регистра

операнда (DR). Результат операции всегда помещается в Асс. При завершении выполнения операции арифметическое устройство вырабатывает сигналы признаков результата: $Z = 1$, если результат равен нулю; $S = 1$, если результат отрицателен; $OV = 1$, если при выполнении операции произошло переполнение разрядной сетки. В случаях, когда эти условия не выполняются, соответствующие сигналы имеют нулевое значение.

В модели ЭВМ предусмотрены внешние устройства двух типов: первый – регистры IR и OR, которые могут обмениваться с аккумулятором с помощью безадресных команд IN (Асс := IR) и OUT (OR := Асс); второй – набор моделей внешних устройств, которые могут подключаться к системе и взаимодействовать с ней в соответствии с заложенными в моделях алгоритмами.

Устройство управления осуществляет выборку команд из ОЗУ в последовательности, определяемой естественным порядком выполнения команд (т. е. в порядке возрастания адресов команд в ОЗУ) или командами передачи управления; выборку из ОЗУ операндов, задаваемых адресами команды; инициирование выполнения операции, предписанной командой; останов или переход к выполнению следующей команды.

Задание:

	IR	Команда 1	Команда 2	Команда 3	Команда 4	Команда 5
7	1	00038	01 0 000	25 1 006	22 0 003	22 3 003
						14 0 001

Команды и коды:

Последовательность	Значения				
Адреса	000	001	002	003	004
Коды	01 0 000	25 1 006	22 0 003	22 3 003	14 0 001
Команды	IN	MUL #006	WR 003	WR [003]	JNS 1

Результаты выполнения:

PC	Асс	M(3)	PC	Асс	M(3)
001	000381		01		
002	002286		02	002286	
003		002286	03		002286
004			04		

001			01	0	
002	013716		02	0	013716
003		013716	03	0	013716
004	000381		04	0	00381

Лабораторная работа №17

ОТЛАДЧИК DEBUG КАК СРЕДСТВО ДЛЯ ОЗНАКОМЛЕНИЯ С АРХИТЕКТУРОЙ INTEL 8086

Цель: научиться использовать программу DEBUG для исследования работы виртуального режима процессора Intel.

Технические средства: персональный компьютер, оснащенный операционной системой DOS или WINDOWS, отладчик двоичного кода DEBUG.

3.1. Краткая теоретическая часть

Изучая работу процессора персонального компьютера, удобно использовать программу DEBUG, встроенную в командную оболочку операционной системы WINDOWS. Эта программа представляет собой шестнадцатирядный отладчик кода программ.

Запуск программы DEBUG

Запустить программу DEBUG можно из командной строки или непосредственно из папки, в которой она находится. Чтобы запустить программу из командной строки, выберите команду из меню ПУСК – ВЫПОЛНИТЬ или нажмите комбинацию клавиш WIN + R. В открывшемся окне напечатайте слово

«debug» и нажмите клавишу ENTER или щелкните кнопку ОК.

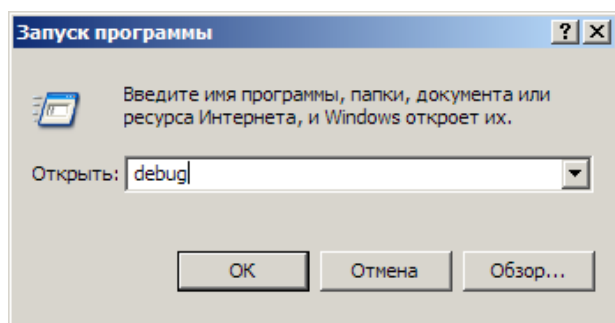


Рис. 1. Запуск программы Debug

В качестве приглашения выступает знак «минус».

Использование программы DEBUG

Теперь можно вводить команды программы, например, для вывода справки нужно ввести знак вопроса «?» и нажать клавишу «Enter».

Команды можно вводить как в верхнем, так и в нижнем регистре. Все числовые значения являются шестнадцатеричными-

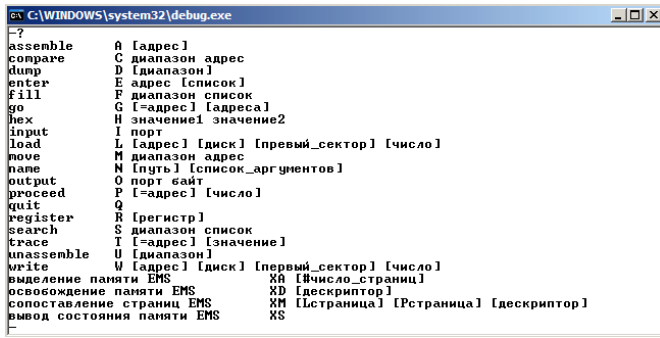


Рис. 2. Вывод справки

Чтобы вывести дамп памяти с адреса 0B2B:0100 до адреса 0B2B:0200 требуется ввести команду «D 100 200».

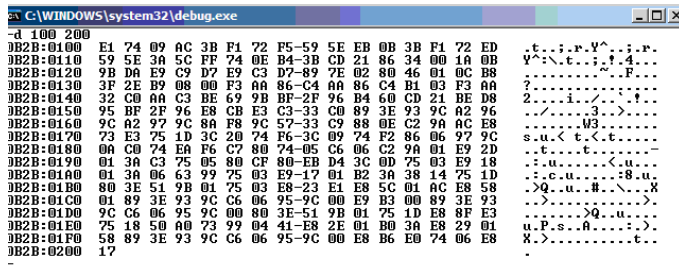


Рис. 3. Приветствие командной строки отладчика Debug

Для запуска программы на выполнение есть несколько путей:

1. Запустить программу командой «Go». В этом случае в командный интерпретатор удастся вернуться только после завершения всей программы.
2. Использовать команду «Trace». Она позволит выполнять последовательно каждую ассемблерную команду.
3. Использовать команду «Proceed». Так же, как и «Trace», выполняет по одной инструкции, но выполнение инструкций CALL, LOOP, INT или повторяемой строковой инструкции с префиксами REPnn происходит как выполнение одной команды.

Например, для выполнения одной инструкции, находящейся в памяти по адресу XXXX:0100, следует изменить значение регистра IP на 100 командой «R IP» и выполнить команду «T».

Таблица 3.1

Команды отладчика DEBUG

Команд	Описание
A (Assemble)	Транслирование команд ассемблера в машинный код; адрес по умолчанию – CS:0100h. A [<адрес_начала_кода>]
C (Compare)	Сравнение содержимого двух областей памяти; по умолчанию используется DS. В команде указывается либо длина участков, либо диапазон адресов. C <начальный_адрес_1> L<длина> <начальный_адрес_2> C <начальный_адрес_1> <конечный_адрес_1> <начальный_адрес_2>

D (Display/ Dump)	Вывод содержимого области памяти в шестнадцатеричном и ASCII-форматах. По умолчанию используется DS; можно указывать длину или диапазон. D [<начальный_адрес> [L<длина>]] D [начальный_адрес конечный_адрес]
E (Enter)	Ввод в память данных или инструкции машинного кода; по умолчанию используется DS. E [<адрес> [<инструкции/данные>]]

Продолжение таблицы 3.1

Команд	Описание
F (Fill)	Заполнение области памяти данными из списка; по умолчанию используется DS. Использовать можно как длину, так и диапазон. F <начальный_адрес_1> L<длина> '<данные>' F <начальный_адрес> <конечный_адрес> '<данные>'
G (Go)	Выполнение отлаженной программы на машинном языке до указанной точки останова; по умолчанию используется CS. При этом убедитесь, что IP содержит корректный адрес. G [=<начальный_адрес>] <адрес_останова> [<адрес_останова> ...]
H (Hexadecimal)	Вычисление суммы и разности двух шестнадцатеричных величин. H <величина_1> <величина_2>
I (Input)	Считывание и вывод одного байта из порта. I <адрес_порта>
L (Load)	Загрузка файла или данных из секторов диска в память; по умолчанию – CS:100h. Файл можно указать с помощью команды N или аргумента при запуске debug.exe. L [<адрес_в_памяти_для_загрузки>] L [<адрес_в_памяти_для_загрузки> [<номер_диска> <начальный_сектор> <количество_секторов>]]
M (Move)	Копирование содержимого ячеек памяти; по умолчанию используется DS. Можно указывать как длину, так и диапазон. M <начальный_адрес> L<длина> <адрес_назначения> M <начальный_адрес> <конечный_адрес> <адрес_назначения>
N (Name)	Указание имени файла для команд L и W. N <имя_файла>
O (Output)	Отсылка байта в порт. O <адрес_порта> <байт>

Окончание таблицы 3.1

Команд	Описание
P (Proceed)	Выполнение инструкций CALL, LOOP, INT или повторяемой строковой инструкции с префиксами REPnn, переходя к следующей инструкции. P [=<адрес_начала>] [<количество_инструкций>]
Q (Quit)	Завершение работы debug.exe
R (Register)	Вывод содержимого регистров и следующей инструкции. R <имя_регистра>
S (Search)	Поиск в памяти символов из списка; по умолчанию используется DS. Можно указывать как длину, так и диапазон. S <начальный_адрес> L<длина> ' <данные>' S <начальный_адрес> <конечный_адрес> ' <данные>'
T (Trace)	Пошаговое выполнение программы. Как и в команде P, по умолчанию используется пара CS:IP. Замечу, что для выполнения прерываний лучше пользоваться командой P. T [=<адрес_начала>] [<количество_выполняемых_команд>]
U (Unassemble)	Дизассемблирование машинного кода; по умолчанию используется пара CS:IP. К сожалению, debug.exe некорректно дизассемблирует специфические команды процессоров 80286+, хотя они все равно выполняются корректно. U [<начальный_адрес>] U [<начальный_адрес_конечный_адрес>]
W (Write)	Запись файла из debug.exe; необходимо обязательно задать имя файла командой N, если он не был загружен. А программы записываются только в виде файлов .COM! Число байт записываемой информации должно содержаться в регистре CX. W [<адрес> [<номер_диска> <начальный_сектор> <количество_секторов>]]

3.2. Практическая часть

Задание 1. При помощи отладчика Debug заполнить таблицу 3.2.

Таблица 3.2

Отчет выполнения работы с программой DEBUG

1. Вывести на экран содержимое регистров	
2. Вывести на экран 139_{10} , байт памяти, начиная с $100H$	
3. Присвоить имя программе	
4. Записать на диск программу, состоящую из кода, находящегося по адресу $2B3_{16}$ размером 134_{10} байт, и данных, находящихся по адресу 340_{16} размером 200_{10} байт	

3.3. Контрольные вопросы

1. Используя отладчик DEBUG в режиме виртуальной адресации, можно ли повредить операционную систему?
2. Содержимое каких регистров можно узнать при помощи отладчика DEBUG?

Лабораторная работа №18

Принципы работы кэш-памяти

Цель настоящей лабораторной работы — проверить работу различных алгоритмов замещения при различных режимах записи.

1. Задание 7

В качестве задания предлагается некоторая короткая "программа" (табл. 14), которую необходимо выполнить с подключенной кэш-памятью (размером 4 и 8 ячеек) в шаговом режиме для следующих двух вариантов алгоритмов замещения (табл.13).

Таблица 14 (окончание)

№ варианта	Номера команд программы						
	1	2	3	4	5	6	7
7	RD #6	CALL 006	WR11	WRR2	PUSH R2	RET	JMP 002
8	RD#8	WRR2	WR @R2+	PUSH R2	POP R3	WR -@R3	CALL 003
9	RD #13	WR14	WR@14	WR@13	ADD 13	CALL 006	RET
10	RD #42	SUB #54	WR16	WR@16	WRR1	ADD @R1+	PUSH R1
11	RD #10	WRR5	ADD R5	WRR6	CALL 005	PUSH R6	RET
12	JMP 006	RD #76	WR 14	WRR2	PUSH R2	RET	CALL 001

Не следует рассматривать заданную последовательность команд как фрагмент программы¹. Некоторые конструкции, например, последовательность команд PUSH R6, RET в общем случае не возвращает программу в точку вызова подпрограммы. Такие группы команд введены в задание для того, чтобы обратить внимание студентов на особенности функционирования стека.

2. Порядок выполнения работы

1. Ввести в модель учебной ЭВМ текст своего варианта программы (см.табл. 14), ассемблировать его и сохранить на диске в виде txt-файла.

2. Установить параметры кэш-памяти размером 4 ячейки, выбрать режим

записи и алгоритм замещения в соответствии с первой строкой своего варианта из табл. 13.

3. В шаговом режиме выполнить программу, фиксируя после каждого шага состояние кэш-памяти.

4. Для одной из команд записи (WR) перейти в режим Такт и отметить, в каких микрокомандах происходит изменение кэш-памяти.

5. Для кэш-памяти размером 8 ячеек установить параметры в соответствии со второй строкой своего варианта из табл. 13 и выполнить

программу в шаговом режиме еще раз, фиксируя последовательность номеров замещаемых ячеек кэш-памяти.

3.Содержание отчета

1. Вариант задания — текст программы и режимы кэш-памяти.
2. Последовательность состояний кэш-памяти размером 4 ячейки при однократном выполнении программы (команды 1—7).
3. Последовательность микрокоманд при выполнении команды т. с отметкой тех микрокоманд, в которых возможна модификация кэш-памяти.
4. Для варианта кэш-памяти размером 8 ячеек — последовательность номеров замещаемых ячеек кэш-памяти для второго варианта параметров кэш памяти при двукратном выполнении программы (команды 1—7).

4.Контрольные вопросы

1. В чем смысл включения кэш-памяти в состав ЭВМ?
2. Как работает кэш-память в режиме обратной записи? Сквозной записи?
3. Как зависит эффективность работы ЭВМ от размера кэш-памяти?
4. В какую ячейку кэш-памяти будет помещаться очередное слово, если свободные ячейки отсутствуют?
5. Какие алгоритмы замещения ячеек кэш-памяти вам известны?

Лабораторная работа №19

Программирование внешних устройств

Целью этой лабораторной работы является изучение способов организации взаимодействия процессора и внешних устройств (ВУ) в составе ЭВМ.

Выше отмечалось, что связь процессора и ВУ может осуществляться в синхронном или асинхронном режиме. Синхронный режим используется для ВУ, всегда готовых к обмену. В нашей модели такими ВУ являются дисплей и тоногенератор — процессор может обращаться к этим ВУ, не анализируя их состояние (правда дисплей блокирует прием данных после ввода 128 символов, формируя флаг ошибки).

Асинхронный обмен предполагает анализ процессором состояния ВУ, которое определяет готовность ВУ выдать или принять данные или факт осуществления некоторого события, контролируемого системой. К таким устройствам в нашей модели можно отнести клавиатуру и блок таймеров.

Анализ состояния ВУ может осуществляться процессором двумя способами:

- в программно-управляемом режиме;
- в режиме прерывания.

В первом случае предполагается программное обращение процессора к регистру состояния ВУ с последующим анализом значения соответствующего разряда слова состояния. Такое обращение следует предусмотреть в программе с некоторой периодичностью, независимо от

фактического наступления контролируемого события (например, нажатие клавиши).

Во втором случае при возникновении контролируемого события ВУ формирует процессору запрос на прерывание программы, по которому процессор и осуществляет связь с ВУ.

1. Задание 6

Свой вариант задания (табл. 12) требуется выполнить двумя способами — сначала в режиме программного контроля, далее модифицировать программу таким образом, чтобы события обрабатывались в режиме прерывания программы. Поскольку "фоновая" (основная) задача для этого случая в заданиях отсутствует, роль ее может сыграть "пустой цикл":

M: NOP

NOP

JMP M

Таблица 9.12. Варианты задания 6

№ варианта	Задание	Используемые ВУ	Пояснения
1	Ввод пятиразрядных чисел в ячейки ОЗУ	Клавиатура	Программа должна обеспечивать ввод последовательности ASCII-кодов десятичных цифр (не длиннее пяти), перекодировку в "8421", упаковку в десятичное число (первый введенный символ — старшая цифра) и размещение в ячейке ОЗУ. ASCII-коды не-цифр игнорировать
2	Программа ввода символов с клавиатуры с выводом на дисплей	Клавиатура, дисплей, таймер	Очистка буфера клавиатуры после ввода 50 символов или каждые 10 с
3	Вывод на дисплей трех текстов, хранящихся в памяти, с задержкой	Дисплей, таймер	Первый текст выводится сразу при запуске программы, второй — через 15 с, третий — через 20 с после второго
4	Вывод на дисплей одного из трех текстовых сообщений, в зависимости от нажатой клавиши	Клавиатура, дисплей	<1> — вывод на дисплей первого текстового сообщения, <2> — второго, <3> — третьего, остальные символы — нет реакции
5	Выбирать из потока ASCII-кодов только цифры и выводить их на дисплей	Клавиатура, дисплей, тоногенератор	Вывод каждой цифры сопровождается коротким звуковым сигналом
6	Выводить на дисплей каждый введенный с клавиатуры символ, причем цифру выводить "в трех экземплярах"	Клавиатура, дисплей, тоногенератор	Вывод каждой цифры сопровождается троекратным звуковым сигналом
7	Селективный ввод символов с клавиатуры	Клавиатура, дисплей	Все русские буквы, встречающиеся в строке ввода — в верхнюю часть экрана дисплея (строки 1—4), все цифры — в нижнюю часть экрана (строки 5—8), остальные символы не выводить
8	Вывод содержимого заданного участка памяти на дисплей посылкой с заданным промежутком времени между выводами символов	Дисплей, таймер	Остаток от деления на 256 трех младших разрядов ячейки памяти рассматривается как ASCII-код символа. Начальный адрес памяти, длина массива вывода и промежуток времени — параметры подпрограммы
9	Программа ввода символов с клавиатуры с выводом на дисплей	Клавиатура, дисплей	Очистка буфера клавиатуры после ввода 35 символов

10	Выводить на дисплей каждый введенный с клавиатуры символ, причем заглавную русскую букву выводить "в двух экземплярах"	Клавиатура, дисплей, таймер	Очистка буфера клавиатуры после ввода 48 символов, очистка экрана каждые 15 с
11	Вывод на дисплей содержимого группы ячеек памяти в числовой форме (адрес и длина группы — параметры подпрограммы)	Дисплей, таймер	Содержимое ячейки распаковывается (с учетом знака), каждая цифра преобразуется в соответствующий ASCII-код и выдается на дисплей. При переходе к выводу содержимого очередной ячейки формируется задержка 10 с
12	Определить промежуток времени между двумя последовательными нажатиями клавиш	Клавиатура, таймер	Результат выдается на ОР. (Учитывая инерционность модели, нажатия не следует производить слишком быстро.)

2. Задания повышенной сложности

1. Разработать программу-тест на скорость ввода символов с клавиатуры. По звуковому сигналу включается клавиатура и таймер на T секунд. Можно начинать ввод символов, причем каждый символ отображается на дисплее, ведется подсчет количества введенных символов (после каждых 50 дается команда на очистку буфера клавиатуры, после 128 — очищается дисплей).

Переполнение таймера выключает клавиатуру и включает сигнал завершения ввода (можно тон этого сигнала сопоставить с количеством введенных символов). Параметр Γ вводится из ИР. Результат S — средняя скорость ввода (символ/с) выдается на ОР. Учитывая, что модель учебной ЭВМ оперирует только целыми числами, можно выдавать результат в формате 5x60 символов/мин.

2. Разработать программу-тест на степень запоминания текста. Три различных варианта текста выводятся последовательно на дисплей на T_1 секунде промежутками T_2 секунд. Далее эти тексты (то, что запомнилось) вводятся с клавиатуры (в режиме ввода строки) и программно сравниваются с исходными текстами. Выдается количество (процент) ошибок.

3. Разработать программу-калькулятор. Осуществлять ввод из буфера клавиатуры последовательности цифр, упаковку (см. задание 1 в табл. 12).

Разделители — знаки бинарных арифметических операций и =. Результат переводится в ASCII-коды и выводится на дисплей.

3. Порядок выполнения работы

1. Запустить программную модель учебной ЭВМ и подключить к ней определенные в задании внешние устройства (меню Внешние устройства |Менеджер ВУ).

2. Написать и отладить программу, предусмотренную заданием, с использованием программного анализа флагов готовности ВУ. Продемонстрировать работающую программу преподавателю.

3. Изменить отлаженную в п. 2 программу таким образом, чтобы процессор реагировал на готовность ВУ с помощью подсистемы прерывания. Продемонстрировать работу измененной программы преподавателю.

4. Содержание отчета

1. Текст программы с программным анализом флагов готовности ВУ.
2. Текст программы с обработчиком прерывания.

5. Контрольные вопросы

1. При каких условиях устанавливается и сбрасывается флаг готовности клавиатуры Rd?
2. Возможно ли в блоке таймеров организовать работу всех трех таймеров с разной тактовой частотой?
3. Как при получении запроса на прерывание от блока таймеров определить номер таймера, достигшего состояния 99 999 (00 000)?
4. Какой текст окажется на экране дисплея, если после нажатия в окне обозревателя дисплея кнопки Очистить и загрузки по адресу CR (11) константы #10 вывести по адресу DR (10) последовательно пять ASCII-кодов русских букв А, Б, В, Г, Д?
5. В какой области памяти модели ЭВМ могут располагаться программы —обработчики прерываний?
6. Какие изменения в работе отлаженной вами второй программы произойдут, если завершить обработчик прерываний командой RET, а не IRET?

Таблица 13. Пояснения к вариантам задания 7

Номера вариантов	Режим записи	Алгоритм замещения
1, 7, 11	Сквозная	СЗ, без учета бита записи
	Обратная	О, с учетом бита записи
2, 5, 9	Сквозная	БИ, без учета бита записи
	Обратная	О, с учетом бита записи
3, 6, 12	Сквозная	О, без учета бита записи
	Обратная	СЗ, с учетом бита записи
4, 8, 10	Сквозная	БИ, без учета бита записи
	Обратная	БИ, с учетом бита записи

Таблица 9.14. Варианты задания 7

№ варианта	Номера команд программы						
	1	2	3	4	5	6	7
1	RD #12	WR 10	WR @10	ADD 12	WR R0	SUB 10	PUSH R0
2	RD #65	WRR2	MOV R4,R2	WR 14	PUSH R2	POP R3	CALL 002
3	RD #16	SUB #5	WR 9	WR @9	WR R3	PUSH R3	POP R4
4	RD #99	WR R6	MOV R7,R6	ADD R7	PUSH R7	CALL 006	POP R8
5	RD #11	WR R2	WR -@R2	PUSH R2	CALL 005	POP R3	RET
6	RD #19	SUB #10	WR9	ADD #3	WR @9	CALL 006	POP R4

Лабораторная работа №20

Программирование разветвляющегося процесса

Для реализации алгоритмов, пути в которых зависят от исходных данных, используют команды условной передачи управления.

1. Пример 2

В качестве примера рассмотрим программу вычисления функции

$$y = \begin{cases} (x-11)^2 - 125, & \text{при } x \geq 16, \\ \frac{x^2 + 72x - 6400}{-168}, & \text{при } x < 16, \end{cases}$$

причем x вводится с устройства ввода ИР, результат y выводится на ОР. Граф-схема алгоритма решения задачи показана на рис. 1.

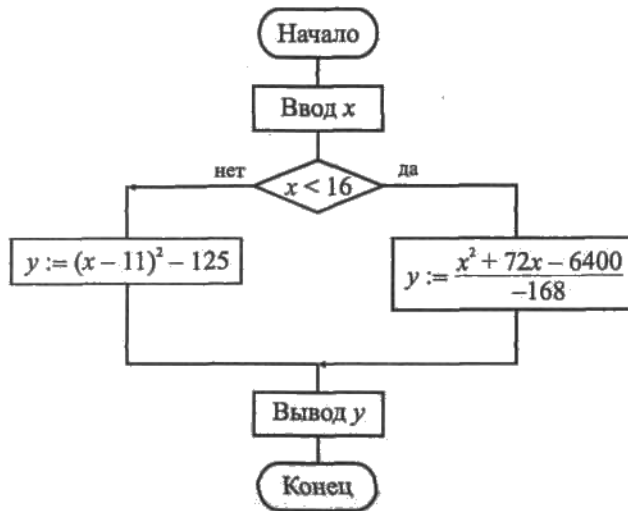


Рис. 9.1. Граф-схема алгоритма

В данной лабораторной работе используются двухсловные команды с непосредственной адресацией, позволяющие оперировать отрицательными числами и числами по модулю, превышающие 999, в качестве непосредственного операнда.

Оценив размер программы примерно в 20—25 команд, отведем для области данных ячейки ОЗУ, начиная с адреса 030. Составленная программа с комментариями представлена в виде табл. 4.

Таблица 9.4. Программа

Адрес	Команда		Примечание
	Мнемокод	Код	
000	IN	01 0 000	Ввод x
001	WR 30	22 0 030	Размещение x в ОЗУ(ОЗО)
002	SUB #16	24 1 016	Сравнение с границей — ($x-16$)
003	JS 010	13 0 010	Переход по отрицательной разности
004	RD 30	21 0 030	Вычисления по первой формуле
005	SUB #11	24 1 011	
006	WR 31	22 0 031	
007	MUL 31	25 0 031	
008	SUB #125	24 1 125	
009	JMP 020	10 0 020	Переход на вывод результата
010	RD 30	21 0 030	Вычисления по второй формуле
011	MUL 30	25 0 030	
012	WR 31	22 0 031	
013	RD 30	21 0 030	
014	MUL #72	25 1 072	
015	ADD 31	23 0 031	
016	ADI	43 0 000	
	106400		
017		106400	
018	DIVI	46 0 000	
	100168		
019		100168	
020	OUT	02 0 000	Вывод результата
021	HLT	09 0 000	Стоп

2. Задание 2

1. Разработать программу вычисления и вывода значения функции:

$$y = \begin{cases} F_i(x), & \text{при } x \geq a, \\ F_j(x), & \text{при } x < a, \end{cases}$$

для вводимого из \mathbb{R} значения аргумента x . Функции и допустимые пределы изменения аргумента приведены в табл. 5, варианты заданий — в табл. 6.

2. Исходя из допустимых пределов изменения аргумента функций (табл. 5)

и значения параметра a для своего варианта задания (табл. 6) выделить на числовой оси Ox области, в которых функция y вычисляется по представленной в п. 1 формуле, и недопустимые значения аргумента. На недопустимых значениях аргумента программа должна выдавать на OK максимальное отрицательное число: 199 999.

3. Ввести текст программы в окно Текст программы, при этом возможен набор и редактирование текста непосредственно в окне Текст программы или загрузка текста из файла, подготовленного в другом редакторе.

4. Ассемблировать текст программы, при необходимости исправить синтаксические ошибки.

5. Отладить программу. Для этого:

а) записать в \mathbb{R} значение аргумента $x > a$ (в области допустимых значений);

б) записать в PC стартовый адрес программы;

в) проверить правильность выполнения программы (т. е. правильность результата и адреса останова) в автоматическом режиме. В случае наличия ошибки выполнить пп. 5, г и 5, д; иначе перейти к п. 5, е;

г) записать в PC стартовый адрес программы;

д) наблюдая выполнение программы в режиме Шаг, найти команду, являющуюся причиной ошибки; исправить ее; выполнить пп. 5, а — 5, в;

е) записать в \mathbb{R} значение аргумента $x < a$ (в области допустимых значений); выполнить пп. 5, б и 5, в;

ж) записать в \mathbb{R} недопустимое значение аргумента x и выполнить пп. 5, б и 5, в.

6. Для выбранного допустимого значения аргумента x наблюдать выполнение отлаженной программы в режиме Шаг и записать в форме табл. 2 содержимое регистров ЭВМ перед выполнением каждой команды.

Таблица 9.5. Функции

k	$F_k(x)$	k	$F_k(x)$
1	$\frac{x+17}{1-x}; 2 \leq x \leq 12$	5	$\frac{(x+2)^2}{15}; 50 \leq x \leq 75$
2	$\frac{(x+3)^2}{x}; 1 \leq x \leq 50$	6	$\frac{2x^2+7}{x}; 1 \leq x \leq 30$
3	$\frac{1000}{x+10}; -50 \leq x \leq -15$	7	$\frac{x^2+2x}{10}; -50 \leq x \leq 50$
4	$(x+3)^3; -20 \leq x \leq 20$	8	$\frac{8100}{x^2}; 1 \leq x \leq 90$

Таблица 9.6. Варианты задания 2

Номер варианта	i	j	a	Номер варианта	i	j	a
1	2	1	12	8	8	6	30
2	4	3	-20	9	2	6	25
3	8	4	15	10	5	7	50
4	6	1	12	11	2	4	18
5	5	2	50	12	8	1	12
6	7	3	15	13	7	6	25
7	6	2	11	14	1	4	5

3. Содержание отчета

Отчет о лабораторной работе должен содержать следующие разделы:

1. Формулировка варианта задания.
2. Граф-схема алгоритма решения задачи.
3. Размещение данных в ОЗУ.
4. Программа в форме табл. 4.
5. Последовательность состояний регистров ЭВМ при выполнении программы в режиме Шаг для одного значения аргумента.

6. Результаты выполнения программы для нескольких значений аргумента, выбранных самостоятельно.

4. Контрольные вопросы

1. Как работает механизм косвенной адресации?
2. Какая ячейка будет адресована в команде с косвенной адресацией через ячейку 043, если содержимое этой ячейки равно 102 347?
3. Как работают команды передачи управления?
4. Что входит в понятие "отладка программы"?

Лабораторная работа №21

Программирование цикла с переадресацией

При решении задач, связанных с обработкой массивов, возникает необходимость изменения исполнительного адреса при повторном выполнении некоторых команд. Эта задача может быть решена путем использования косвенной адресации.

1. Пример

Разработать программу вычисления суммы элементов массива чисел. Исходными данными в этой задаче являются: n — количество суммируемых чисел и C_1, C_2, \dots, C_n — массив суммируемых чисел. Заметим, что должно выполняться условие $n > 1$, т. к. алгоритм предусматривает, по крайней мере, одно суммирование. Кроме того, предполагается, что суммируемые числа записаны в ОЗУ подряд, т. е. в ячейки памяти с последовательными адресами. Результатом является сумма S .

Составим программу для вычисления суммы со следующими конкретными параметрами: число элементов массива — 10, элементы массива расположены в ячейках ОЗУ по адресам 040, 041, 042, ..., 049. Используемые для решения задачи промежуточные переменные имеют следующий смысл: A_i — адрес числа C_i , $i \in \{1, 2, \dots, 10\}$; $ОЗУ(A_i)$ — число по адресу A_i , S — текущая сумма; k — счетчик цикла, определяющий число повторений тела цикла.

Распределение памяти таково. Программу разместим в ячейках ОЗУ, начиная с адреса 000, примерная оценка объема программы — 20 команд; промежуточные переменные: A_i — в ячейке ОЗУ с адресом 030, k — по адресу 031, S — по адресу 032. ГСА программы показана на рис. 2, текст программы с комментариями приведен в табл. 7.

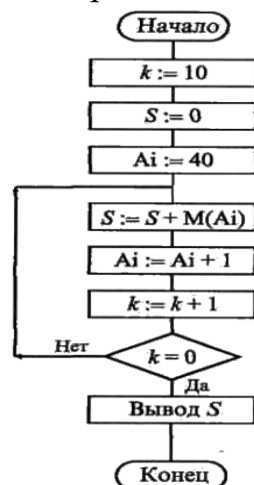


Рис. 9.2. Граф-схема алгоритма для примера 3

Таблица 9.7. Текст программы примера 3

Адрес	Команда	Примечание
000	RD #40	Загрузка начального адреса массива 040
001	WR 30	в ячейку 030

Таблица 9.7 (окончание)

Адрес	Команда	Примечание
002	RD #10	Загрузка параметра цикла $k = 10$ в ячейку 031
003	WR 31	
004	RD #0	Загрузка начального значения суммы $S = 0$
005	WR 32	в ячейку 032
006	M1: RD 32	Добавление
007	ADD @30	к текущей сумме
008	WR 32	очередного элемента массива
009	RD30	Модификация текущего
010	ADD #1	адреса массива
011	WR 30	(переход к следующему адресу)
012	RD 31	Уменьшение счетчика
013	SUB #1	(параметра цикла)
014	WR 31	на 1
015	JNZ M1	Проверка параметра цикла и переход при $k \neq 0$
016	RD 32	Вывод
017	OUT	результата
018	HLT	Стоп

2. Задание 3

1. Написать программу определения заданной характеристики последовательности чисел C_1, C_2, \dots, C_n . Варианты заданий приведены в табл. 8.

2. Записать программу в мнемосодах, введя ее в поле окна Текст программы.

3. Сохранить набранную программу в виде текстового файла и произвести ассемблирование мнемосокодов.

4. Загрузить в ОЗУ необходимые константы и исходные данные.

5. Отладить программу.

Таблица 9.8. Варианты задания 3

Номер варианта	Характеристика последовательности чисел C_1, C_2, \dots, C_n
1	Количество четных чисел
2	Номер минимального числа
3	Произведение всех чисел
4	Номер первого отрицательного числа
5	Количество чисел, равных C_1
6	Количество отрицательных чисел
7	Максимальное отрицательное число
8	Номер первого положительного числа
9	Минимальное положительное число
10	Номер максимального числа
11	Количество нечетных чисел
12	Количество чисел, меньших C_1
13	Разность сумм четных и нечетных элементов массивов
14	Отношение сумм четных и нечетных элементов массивов

Примечание. Под четными (нечетными) элементами массивов понимаются элементы массивов, имеющие четные (нечетные) индексы. Четные числа — элементы массивов, делящиеся без остатка на 2.

3. Содержание отчета

1. Формулировка варианта задания.
2. Граф-схема алгоритма решения задачи.
3. Распределение памяти (размещение в ОЗУ переменных, программы и необходимых констант).
4. Программа.
5. Значения исходных данных и результата выполнения программы.

4. Контрольные вопросы

1. Как организовать цикл в программе?
2. Что такое параметр цикла?
3. Как поведет себя программа, приведенная в табл. 7, если в ней будет отсутствовать команда WR 31 по адресу 014?
4. Как поведет себя программа, приведенная в табл.7, если метка M1 будет поставлена по адресу 005? 007?

Лабораторная работа №22

Исследование функциональных узлов ЭВМ комбинационного и последовательного типа

Ранее упоминалась ИМС арифметико-логического устройства (АЛУ) 74181 (К155ИПЗ) в связи с возможностью использования ее в качестве четырехразрядного сумматора. Там же указывалось, что эта ИМС обеспечивает 32 режима работы АЛУ в зависимости от состояния управляющих сигналов на входах M, SO...S3. Показанная на рис. 9.53 схема на базе этой ИМС позволяет оперативно реализовать все упоминавшиеся режимы.

Возможные режимы задаются с помощью переключателей O, 1, 2, 3 для подачи сигналов 0 ("земля") или 1 (+5 В) на входы управления SO, SI, S2, S3. В положении переключателя M, показанном на рис. 9.53 (сигнал 0 на входе M), выполняются 16 арифметических операций (16 комбинаций сигналов SO...S3) с учетом переноса по входу Sp (переключатель C в показанном на рис. 9.53 положении) или без учета переноса (сигнал 0 на входе Sp переключателя C). При переводе ключа M в другое положение (на входе M сигнал 1) выполняются 16 логических операций, задаваемых теми же переключателями O... 3.

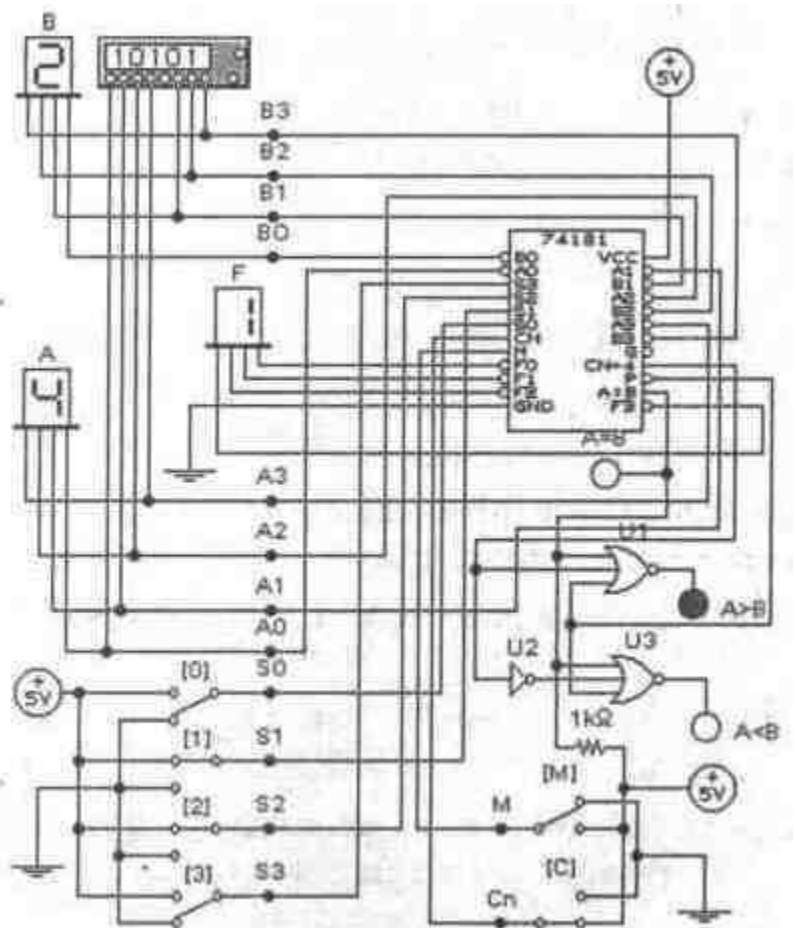


Рис. 9.53. АЛУ на ИМС 74181

Значения четырехразрядных операндов A и B задаются с помощью генератора слова и в шестнадцатеричном коде отображаются на алфавитно-цифровых индикаторах. На выходах $F0...F3$ результат суммирования отображается индикатором F . При коде 1111 на этих выходах и при равенстве операндов выход $A=B$ переводится в единичное состояние. Поскольку этот выход представляет собой каскад с открытым коллектором, то на него подается питание $+5$ В через резистор 1 кОм. Выход $A=B$ совместно с выходом переноса $CN+4$ и выходом P подтверждения переноса используются для формирования признаков $A>B$ и $A<B$ с помощью дополнительных логических элементов $U1, U2, U3$.

Изменяя состояния сигналов на управляющих входах, можно промоделировать большинство функций АЛУ, используемых в микропроцессорах. Приведем перечень этих функций.

Логические функции (на входе M сигнал 1); выполняются поразрядно, переносы не учитываются.

Код 0000 на входах $S3, S2, S1, S0$; при этом выполняется логическая функция A' — данные со входов A передаются на выходы F с инверсией, может быть использована в команде SMA (здесь и далее используется мнемоника команд микропроцессоров семейства 80xx фирмы Intel).

0001 — $(A+B)'$ — поразрядная операция ИЛИ с инверсией над операндами А и В;
 0010 — $A'B$ — операция И инвертированного операнда А и операнда В;
 00Н — 0 — нет операции;
 0100 — $(AB)'$ — операция И с инверсией;
 0101 — B' — инверсия операнда В;
 0НО — $A\Phi B$ — операция Исключающее ИЛИ, команда XRA;
 0111 — AB' — операция И над операндами А и инверсией В;
 1000 — $A'+B$ — операция ИЛИ над инверсией А и операндом В;
 1001 — $(A+B)'$ — операция ИЛИ с инверсией;
 1010 — В — передача на выход операнда В;
 1011 — AB — операция И, команда ANA;
 1100—1;
 1101 — $A+B'$ — операция ИЛИ над инверсией В и операндом А;
 1110 — $A+B$ — операция ИЛИ, команда ORA;
 1111 — А — передача на выход операнда А.

Арифметические операции ($M=0$) без переноса ($Cp=1$) и с переносом ($Cp=0$, данные приводятся в круглых скобках):

0000 — А — передача на выход операнда ($A+1$ — суммирование операнда с 1 переноса, команда инкремента).
 0001 — $A+B$ — операция суммирования без учета переноса, команда ADD ($(A+B)+1$ — суммирование с учетом переноса, команда ADC);
 0010 — $A+B'$ — операция суммирования операнда А с инверсией операнда В без учета переноса ($(A+B')+1$ — то же, но с учетом переноса);
 00Н — -1 (0);
 0100 — $A+AB'$ ($A+(AB)'+1$). Далее мы от комментариев воздержимся в надежде, что из вышеизложенного все и так очевидно;
 0101 — $(A+B)+AB'$ ($(A+B)+AB'+1$);
 0НО — $A-B-1$, команда SBB ($A-B$, команда SUB);
 01Н— $AB'-1$ ($(AB)'$);
 1000 — $A+AB(A+B+1)$;
 1001 — $A+B$, команда ADD ($A+B+1$);
 1010 — $(A+B')+AB$ ($(A+B')+AB+1$);
 10Н- $AB-1$ (AB);
 1100 — $A+A(A+A+1)$;
 1101 — $(A+B)+A$ ($(A+B)+A+1$);
 1110 — $(A+B')+A$ ($(A+B')+A+1$);
 1111- $A-1$ (A).

Контрольные задания

1. Проведите моделирование всех перечисленных выше режимов АЛУ (рис. 9.53), предварительно составив неповторяющиеся комбинации на выходе генератора слова.
2. Дополните операции без комментариев описанием выполняемых ими функций.

3. Проанализируйте систему команд микропроцессора 18080 (КР580ИК80) и возможность использования в них логических функций и арифметических операций ИМС 74181.