

**Автономная некоммерческая организация профессионального образования
«ПЕРМСКИЙ ГУМАНИТАРНО-ТЕХНОЛОГИЧЕСКИЙ КОЛЛЕДЖ»
(АНО ПО «ПГТК»)**

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ
ПО ВЫПОЛНЕНИЮ ПРАКТИЧЕСКИХ РАБОТ
ПО ДИСЦИПЛИНЕ**

ОП.02 ОПЕРАЦИОННЫЕ СИСТЕМЫ И СРЕДЫ

для специальности

09.02.11 Разработка и управление программным обеспечением
(код и наименование специальности)

Квалификация выпускника
Программист

Форма обучения
Очная

Пермь 2026

Методические рекомендации по выполнению практических работ учебной дисциплины ОП.02 ОПЕРАЦИОННЫЕ СИСТЕМЫ И СРЕДЫ составлен в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.11 Разработка и управление программным обеспечением

Данные методические рекомендации помогут организовать самостоятельную деятельность студентов на основе деятельного и компетентного подходов к обучению, что соответствует ФГОС СПО по специальности 09.02.11 Разработка и управление программным обеспечением.

Программа предназначена для студентов и преподавателей АНО ПО «ПГТК».

Автор – составитель: Могильникова Н.С., старший преподаватель.

Методические рекомендации по выполнению практических работ предназначен для оценивания достижений запланированных результатов по дисциплине ОП.02 «ОПЕРАЦИОННЫЕ СИСТЕМЫ И СРЕДЫ»

Методические рекомендации по выполнению практических работ представляет собой комплект материалов для проведения практических занятий (в форме практической подготовки) и осуществления контроля за выполнением работ.

Методические рекомендации по выполнению практических работ позволяет оценивать:

Код ОК, ПК	Уметь	Знать
ОК. 01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам; ОК. 02 Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности; ОК. 09 Пользоваться профессиональной документацией на государственном и иностранном языках.	<ul style="list-style-type: none"> • Управлять параметрами загрузки операционной системы. • Выполнять конфигурирование аппаратных устройств. • Управлять учетными записями, настраивать параметры рабочей среды пользователей. • Управлять дисками и файловыми системами, настраивать сетевые параметры, управлять разделением ресурсов в локальной сети. 	<ul style="list-style-type: none"> • Основные понятия, функции, состав и принципы работы операционных систем. • Архитектуры современных операционных систем. • Особенности построения и функционирования семейств операционных систем "Unix" и "Windows". • Принципы управления ресурсами в операционной системе. • Основные задачи администрирования и способы их выполнения в изучаемых операционных системах.

Перечень заданий для практических работ

- Использование сервисных программ поддержки интерфейсов. Настройка рабочего стола. Настройка системы с помощью Панели управления. Работа со встроенными приложениями.
- Управление памятью.
- Управление процессами с помощью команд операционной системы для работы с процессами.
- Исследование соотношения между представляемым и истинным объемом занятой дисковой памяти. Изучение влияния количества файлов на время, необходимое для их копирования.
- Работа с программой «Файл-менеджер Проводник». Работа с файловыми системами и дисками.
- Диагностика и коррекция ошибок операционной системы, контроль доступа к операционной системе.
- Установка и настройка системы. Установка параметров автоматического обновления системы. Установка новых устройств. Управление дисковыми ресурсами.
- Работа с командами в операционной системе. Использование команд работы с файлами и каталогами. Работа с дисками.
- Конфигурирование файлов. Управление процессами в операционной системе. Резервное хранение, командные файлы.
- Работа с текстовым редактором. Работа с архиватором. Работа с операционной

оболочкой.

- Изучение эмуляторов операционных систем. Установка операционной системы.

Практические занятия: знакомство с понятием «операционная система», видами операционных систем, их назначением; изучение графического интерфейса операционной системы Windows 7

Содержание работы:

Операционная система—базовый комплекс компьютерных программ, обеспечивающий управление аппаратными средствами компьютера, работу с файловой системой, ввод и вывод данных с помощью периферийных устройств, выполнение прикладных программ. При включении компьютера операционная система загружается в оперативную память раньше остальных программ и затем обеспечивает их выполнение.

Для настольных и портативных персональных компьютеров чаще всего используются операционные системы *Microsoft Windows* и *Linux*. На ПК фирмы Apple устанавливается операционная система *Mac OS*. Для КПК разработаны ОС *Windows Mobile* и *Palm OS*.

Минимальным адресуемым элементом носителя информации является **кластер**, который может включать в себя несколько секторов (объем сектора составляет 512 байтов). Размер кластера может составлять от 512 байтов до 64 Кбайт. Кластеры нумеруются в линейной последовательности (на магнитных дисках от первого кластера нулевой дорожки до последнего кластера последней дорожки).

Файловая система организует кластеры в файлы и каталоги (каталог реально является файлом, содержащим список файлов в этом каталоге). Файловая система отслеживает, какие из кластеров в настоящее время используются, какие свободны, какие помечены как неисправные.

Командный процессор - специальная программа, которая запрашивает у пользователя команды и выполняет их.

Драйверы устройств — специальные программы, которые обеспечивают управление работой устройств и согласование информационного обмена с другими устройствами, а также позволяют производить настройку некоторых их параметров. Каждому типу устройств соответствует свой драйвер.

Служебные программы. В состав операционной системы входят также служебные программы, которые позволяют обслуживать диски (проверять, сжимать, дефрагментировать и т. д.), выполнять операции с файлами (архивировать и т. д.), работать в компьютерных сетях и т. д.

Графический интерфейс. Для упрощения работы пользователя в состав современных операционных систем входят программные модули, создающие графический пользовательский интерфейс. В операционных системах с графическим интерфейсом пользователь может вводить команды с помощью диалоговых окон и элементов управления: кнопок, текстовых полей, списков, переключателей, флажков, счетчиков, ползунков и др.

Интерфейс Aero

Операционная система Windows 7 использует интерфейс Aero. Само слово Aero это аббревиатура английских слов: Authentic, Energetic, Reflective, Open (подлинный, энергичный, отражающий и открытый). Интерфейс Aero включает в себя следующий набор функций:

Aero Glass - использование эффекта отношения к заголовкам и различным окнам. При использовании этого запущенного приложения могут



следующий набор

матового стекла по панелям открытых эффекта, за окном проступать

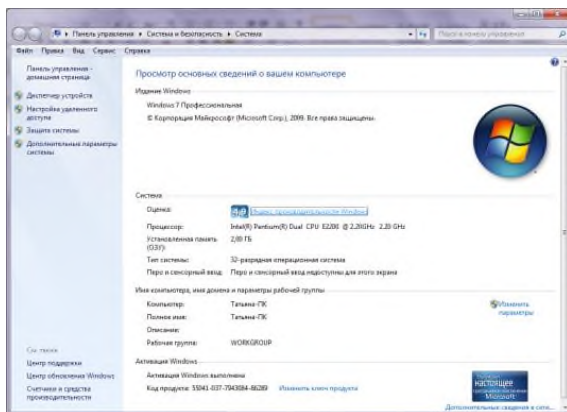
размытые очертания обоев рабочего стола или рисунок следующего открытого окна.

Активные эскизы - миниатюрные изображения открытых окон, с помощью которых возможна удобная и быстрая навигация между открытыми окнами. Активные эскизы могут показываться при работе с панелью задач или при переключении между окнами с помощью клавиатуры.

Aero Snap - при поднесении окна к краю рабочей области монитора происходит автоматическое выравнивание окна. Если окно подносится к верхнему краю, оно разворачивается во весь экран. Если поднести окно к правому или к левому краю, оно займет ровно половину экрана.

Aero Shake - функция, с помощью которой можно свернуть все неактивные приложения движением мыши. Для ее использования нужно захватить заголовок окна и немного "встряхнуть". Если еще раз проделать эту операцию, окна приложений вернуться в исходное положение (Win+Home)

Aero Peek – просмотр миниатюры открытых окон при наведении курсора в область панели задач, переключаться между окнами простым щелчком мыши, быстро сворачивать все окна и увидеть содержимое рабочего стола. Для этого достаточно поднести курсор к крайнему правому краю панели задач (WIN+Пробел).



Оценивание производительности компьютера:

Нажмите на клавиатуре сочетание клавиш **"Win+Break"**, после чего на экране появится окно **"Просмотр основных сведений о вашем компьютере"**.

Базовый индекс производительности - это показатель мощности текущей конфигурации компьютера. Максимальное значение базового индекса производительности, которое может получить ваш компьютер, равно 7,9, минимальное - 1. Для того чтобы можно было

использовать все эффекты Aero, базовый индекс производительности компьютера должен быть не менее "3".

Выбор одной из тем оформления Aero. Для этого щелкните правой кнопкой мыши по рабочему столу, выберите пункт **"Персонализация"**, после чего щелкните по одной из тем в разделе **"Темы Aero"**. Если индекс производительности выше 3, эффекты можно будет наблюдать.

Использование горячих клавиш

Эффективность работы за компьютером во многом зависит от того, насколько быстро можно задействовать тот или иной инструмент. Поэтому знание сочетаний клавиш быстрого доступа к различным функциям **Windows 7** позволит заметно сэкономить время, потраченное на выполнение различных задач, а также упростит многие действия в операционной системе. Например, для переключения между открытыми окнами удобно использовать не только привычный для многих **"Alt+Tab"**, но также и сочетание **"Win+Tab"**. Причем во втором случае можно будет наблюдать красивый трехмерный эффект скольжения окон в трехмерном пространстве - **Flip 3D**.

Стандартное переключение также содержит небольшую "изюминку" - при нажатии **"Alt+Tab"** на экране появятся миниатюры окон запущенных приложений. Эти изображения обладают тем же свойством, что и миниатюры, появляющиеся над панелью задач - при наведении указателя мыши на одну из миниатюр, на экране отображается содержимое этого окна.



Чтобы переключиться в него, достаточно щелкнуть по уменьшенному изображению правой кнопкой мыши.



Горячие клавиши помогут быстро запустить и приложения, расположенные на панели задач. Для запуска этих приложений используйте сочетания клавиш **"Win+1"**, **"Win+2"**, **"Win+3"**, **"Win+4"** и т.д., причем, номер цифры, которую вы выбираете для запуска приложения, соответствует номеру значка на панели задач. Например, в случае, показанном на рисунке, сочетание клавиш **"Win+7"** откроет окно браузера Google

Chrome.

Еще одна удобная команда предназначена для раскрытия окна на весь экран - одновременное нажатие клавиши **Win** и **"↑"** (стрелка вверх). Обратную команду очень легко запомнить - **Win** и **"↓"** (стрелка вниз) сворачивают окна. Нажатие клавиши **Win** и **"→"** (стрелка вправо) или **"Win"+"←"** (стрелка влево) позволяет быстро прикреплять окно к краю рабочей области монитора, на половину экрана. В новой версии Windows стало очень удобно открывать копию уже запущенного приложения. Для этого нужно щелкнуть кнопкой мыши по кнопке программы на панели задач, удерживая при этом нажатую клавишу **"Shift"**.

Стандартные приложения

WordPad: Интерфейс новой версии текстового процессора, интегрированного в **Windows 7**,

очень напоминает внешний вид популярного редактора **MS Word**.

Теперь он обладает наглядным "ленточным" интерфейсом **Ribbon**, с которым пользователи впервые познакомились в **MS Office 2007**.

Одно из интересных нововведений **WordPad** - интеграция с программой **Paint**. Чтобы вставить в текстовый документ графическое изображение с возможностью его дальнейшего редактирования, можно использовать сочетание клавиш **"Ctrl+D"** или нажать кнопку **"Рисунок Paint"** на панели **Ribbon**. Удобство такой интеграции состоит в том, что созданный в **Paint** рисунок не нужно сохранять - после закрытия редактора новое изображение будет автоматически вставлено в документ.

Переработанный Paint

"Ленточный" интерфейс **Ribbon** присутствует не только в новом **WordPad**, его также можно увидеть и в программе **Paint**. Во многом благодаря этому он изменился настолько, что узнать в новой программе старого знакомого практически невозможно.

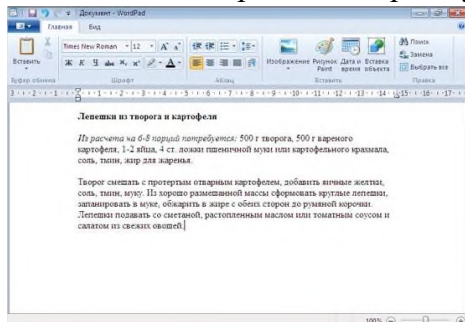


Как и в **WordPad**, тут ленту можно сворачивать, увеличивая при этом рабочую область.

"Записки" - быстрые напоминания на рабочем столе

Чтобы не забывать о важных вещах, которые необходимо сделать, многие используют клейкие листочки, прикрепляя их на видное место, недалеко от компьютера, часто просто на монитор. В таких записках обычно пишут напоминания о том, что нужно сделать, и в какое время.

Утилита "Записки" дает возможность оставлять на рабочем столе виртуальные записки с сообщениями. Просто выберите утилиту в меню



"Пуск", и новая записка тут же отобразится на рабочем столе поверх всех окон. По умолчанию записки

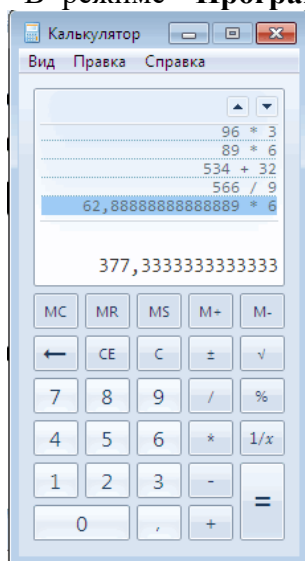
желтые, но при желании цвет "листочков" можно менять.

"Калькулятор": преобразование величин и другие интересные возможности

В новой версии калькулятора присутствует не два, а четыре режима работы. К стандартному виду калькулятора и интерфейсу для инженерных вычислений, в Windows 7 добавлены еще два варианта - "Программист" и "Статистика".

В режиме "Программист" калькулятор может выполнять расчеты в разных системах исчисления - двоичной, десятичной, восьмеричной, шестнадцатеричной. При переключении в режим "Статистика" калькулятор может использоваться для анализа и обработки статистических данных. В новом калькуляторе есть история предыдущих вычислений.

Новинки калькулятора: преобразование величин, подсчет времени между датами и вычисление процентов по ипотеке.

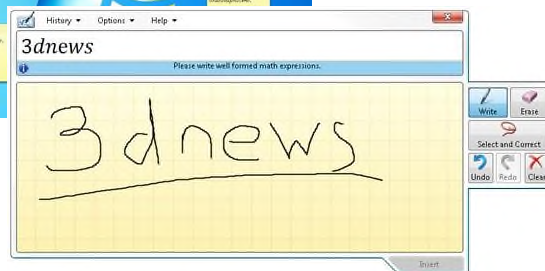


Панель ввода:

Ее



математического рукописный ввод математических выражений



предназначение - ручной ввод и распознавание математических выражений (символов, формул, всевозможных функций и уравнений). Работает панель математического ввода в режиме реального времени. В поле для ввода пользователь рисует указателем мыши или специальным пером символы, которые распознаются программой в реальном времени, буквально на лету. "Панель математического ввода" поддерживает символы для обозначения векторных выражений, тригонометрических функций, элементов логики, трехмерной аналитической геометрии, любых арифметических вычислений и т.д.

Задание 1. Оцените производительность вашего компьютера

Заполните таблицу:

Название операционной системы	
-------------------------------	--

Год создания данной операционной системы	
Индекс производительности операционной системы	
Тип процессора	
Частота процессора	
Количество оперативной памяти	
Тип операционной системы	

Используя вкладку **Счетчики и средства производительности**, заполните таблицу:

Компонент	Что оценивается	Оценка	Общая оценка

Задание 2. Используя приложение «Калькулятор», узнайте, сколько времени прошло между датами: 17 марта 1974 года по сегодняшний день. Промежуток времени укажите в днях, неделях, месяцах и годах.

Задание 3. Используя приложение «Калькулятор», определите дату следующим образом: к сегодняшней дате прибавьте 7 лет 5 месяцев 3 недели и 25 дней.

Задание 4. Вы, как молодая семья, решили купить квартиру в ипотеку. Стоимость квартиры 2млн 750 тыс. рублей, срок, на который вы оформляете займ, 15 лет. Определите ежемесячный платеж, если вы платите первоначальный взнос 300000 руб. Процентная ставки по ипотеке 16,25%.

Задание 5. Переведите 100 м² в см², 10 гектар в м², 25 см в метры, 45 дней в минуты, 30⁰С в градусы по Фаренгейту

Задание 6. В панели математического ввода введите формулу:
$$\sqrt[5]{\frac{\sqrt[2]{x-5x^2}}{(\sin 2x)^{\frac{1}{2}}}}$$

Задание 7. Откройте окно справочной системы **Windows**.

Нажмите кнопку **Пуск** (Start), затем выберите пункт меню **Справка и Поддержка**. Найдите информацию о запуске программ.

- 1) В текстовое поле *Поиск в справке* введите фразу «запуск программ».
- 2) В окне списка разделов щелкните ЛКМ на любой теме и изучите информацию.

Закройте окно **Справки и поддержки**.

Задание 8. Настройте автоматическое исчезновение панели задач:

- 1) На вкладке **Панель задач** установите флажок **Автоматически скрывать панель задач**. Нажмите **Применить**.
- 2) Проверьте, что панель автоматически исчезает с экрана и появляется при наведении указателя мыши на границу экрана.

Снимите флажок **Автоматически скрывать панель задач**. Нажмите **Применить**.

Задание 9. Создайте новую папку **Документы**

- 1) На левой панели **Проводника** щелкните на значок **Компьютер**, затем на значок **Локальный диск (C:)**. На правой панели отобразится содержимое диска.
- 2) Если отсутствует строка меню (**Файл, Правка, Вид** и т.д.) щелкните ЛКМ на кнопку **Упорядочить**, переместите мышь на пункт **Представление**, и в открывшемся списке щелкните ЛКМ по пункту **Строка меню**. Проверьте, что в проводнике появилась строка меню.
- 3) Выберите меню **Файл — Создать — Папка**. В окне **Проводника** появилась новая папка с именем Новая папка.
- 4) Введите с клавиатуры имя папки — **Новые документы** и нажмите ENTER. Вы создали новую, пустую папку.

Задание 10. Создайте еще две новые папки **Картинки** и **Таблицы**

- 1) Повторите шаги задания **10** для создания каждой папки.

Задание 11. Переместите новые папки в **Вашу папку**

О Вниманию! Перед выполнением этого задания убедитесь, что у вас на диске **С:** создана **Ваша папка** (папка с Вашим именем, в которой вы будете сохранять все документы).

Если она не создана, повторите шаги задания 10. В дальнейшем по тексту ваша личная папка будет упоминаться как **Ваша папка**.

- 1) На левой панели **Проводника** выберите **Вашу папку**, на правой панели отобразится содержимое папки — папка пуста, в ней ничего нет.
- 2) На левой панели **Проводника** подведите указатель мыши к папке **Новые документы**, нажмите ПКМ и, удерживая нажатой, потащите на значок папки **Ваша папка**.
- 3) Отпустите ПКМ и выберите **Переместить**.
- 4) Повторите шаги 1-3 для перемещения папок **Картинки** и **Таблицы**.

Задание 12. Отобразите содержимое новых папок

- 1) На левой панели **Проводника** щелкните на значок папки **Новые документы**.
- 2) На правой панели просмотрите содержимое папки — она пуста.
- 3) Повторите шаги 1 -2, чтобы увидеть содержимое папок **Картинки** и **Таблицы**.

Задание 13. Переместите файл **Страна чудес** в папку **Новые документы**

- 1) На левой панели **Проводника** выберите папку **Документы**, на правой панели отобразится содержимое папки — в ней находятся ранее созданные документы.
- 2) На правой панели **Проводника** подведите указатель мыши к документу **Страна чудес**, нажмите ПКМ и, удерживая нажатой, потащите на значок папки **Новые документы** (находящейся в **Вашей папке**).
- 3) Отпустите ПКМ и выберите **Переместить**.

Задание 14. Переместите файл **Картинка** в папку **Картинки**

- 1) Повторите шаги предыдущего задания с той разницей, что файл с картинкой находится не в папке **Документы**, а в папке **Изображения**.

Задание 15. Создайте 2 копии файла **Страна чудес** в папке **Новые документы**

- 1) На левой панели **Проводника** выберите папку **Новые документы**.
- 2) Выделите файл **Страна чудес**, щелкнув на нем ЛКМ.
- 3) Подведите указатель мышки к значку файла, нажмите ПКМ и, удерживая нажатой, потяните значок чуть ниже.
- 4) Отпустите ПКМ и выберите **Копировать**.
- 5) Повторите шаги 2-4 и создайте еще одну копию файла **Страна чудес**.

Задание 16. Создайте 2 копии файла **Картинка** в папке **Картинки**.

Задание 17. Переименуйте папку **Новые документы**.

- 1) На левой панели **Проводника** выберите папку **Новые документы**
- 2) Щелкните на папке ПКМ и выберите **Переименовать**
- 3) Введите с клавиатуры новое имя для папки — **Работа с текстом**
- 4) Нажмите ENTER

Задание 18. Переименуйте файл **Копия Страна чудес**

- 1) На правой панели **Проводника** выберите файл **Страна чудес -копия**.
- 2) Щелкните на нем ПКМ и выберите **Переименовать**.

- 3) Введите с клавиатуры новое имя для файла — **Чудо**.
- 4) Нажмите ENTER.
- 5) Повторите шаги 1-4 для переименования другой копии **Страна чудес** в **Страна**.

Задание 19. Удалите файл **Чудо**.

- 1) На левой панели **Проводника** выделите папку **Работа с текстом**.
- 2) На правой панели выделите файл **Чудо**, щелкнув на нем ЛКМ.
- 3) Нажмите на клавиатуре клавишу DELETE.
- 4) Нажмите ДА для подтверждения удаления. Файл **Чудо** переместился из папки **Работа с текстом** в папку **Корзина**.
- 5) Повторите шаги 1-4 для удаления файл **Страна**, и два файла-копии картинки из папки **Картинки**.

Задание 20. Восстановите удаленные файлы из корзины

- 1) На **Рабочем столе** выделите папку **Корзина**.
На правой панели откроется окно **Корзины (RecycleBin)** со всем ее содержимым. Там должны быть файлы: **Чудо**, **Страна**, две копии **Картинки**.
- 2) Найдите удаленный файл **Чудо**.
- 3) Щелкните на нем ПКМ и выберите команду **Восстановить**.
Файл **Чудо** исчез из **Корзины** и восстановился в ту папку, из которой был удален.
- 4) Повторите шаги 2-3 для восстановления файла **Страна**.

Задание 21. Очистите **Корзину**

- 1) Щелкните ПКМ по значку **Корзина** и выберите **Очистить корзину**.
- 2) Нажмите Да для подтверждения удаления. Все файлы и папки из корзины будут удалены. После очистки корзины восстановление удаленных файлов и папок будет невозможно!

Задание 22. Удалить ненужные файлы

- 1) Удалите самостоятельно файлы **Чудо** и **Страна**.

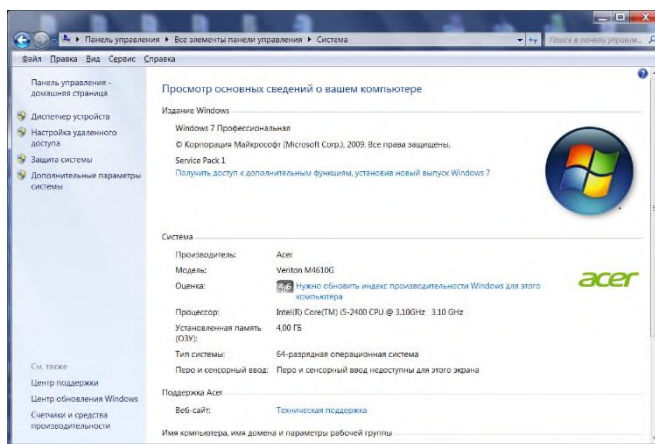
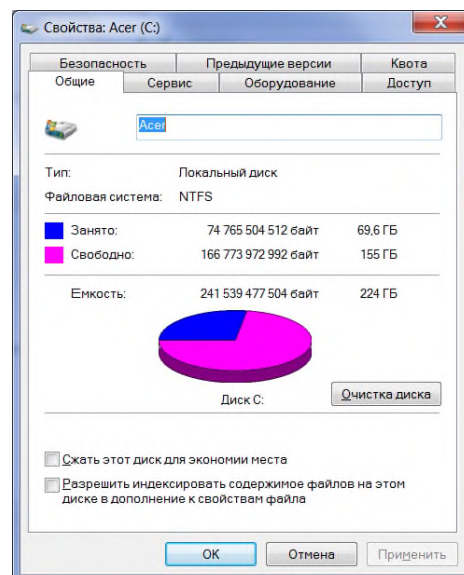
Очистите самостоятельно корзину.

Задание 23. Выписать свойства установленной операционной системы.

1. Рабочий стол - щелкнуть кнопкой мыши по ярлыку **Компьютер** на Рабочем столе (или же **Пуск-Компьютер**) - **Свойства**.
2. Ознакомится с информацией, и выписать в тетрадь:

Задание 1: Название операционной системы: (название, версия, пакет обновлений). **Тип системы:** ____-разрядная ОС. **Процессор** (указать название, тип, тактовую частоту). **Установленная память (ОЗУ)** - ____.

Задание 24. Ознакомится с характеристиками установленных устройств ПК.



1. Справа в меню в окне Система открыть Диспетчер устройств. В открытом окне перечислено оборудование, установленное на Вашем ПК.
2. Щелкнуть по треугольнику рядом с устройством. Откроется соответствующая информация.
3. Ознакомиться с конфигурацией компьютера - информацией по устройствам – и выписать в тетрадь:

Задание 2: IDEATA/ATAPI (что это?) : _____ . Видеоадаптеры (что это?) :

_____. Дисковые устройства (что

_____. Звуковые, видео и игровые устройства (что это?) : _____.

Компьютер (что это?) : _____.

Контроллеры USB (что это?) : _____.

Мониторы (что это?) : _____ . Порты это?) : _____ . Процессоры (что

_____. Сетевые адаптеры (что

_____.


Задание 25. Познакомится со свойствами дисков на примере диска C:.

1. Открыть Проводник.
2. Щелкнуть правой кнопкой мыши по - Свойства.

3. Ознакомиться с информацией, и выписать в тетрадь следующую информацию:

Задание 26. Винчестер: диск C:. Файловая система ____ . Занято - _____ байт, ____ Г байт. Свободно - ____ байт, ____ Гбайт. Емкость - _____ байт, ____ Гбайт.

Задание 27. Выполнить дефрагментацию жесткого диска D в операционной системе Window7 Pro.

1. Закрыть все работающие программы.
2. Прочитать в Википедии (ru.wikipedia.org) информацию про дефрагментацию диска.
3. Запустите программу дефрагментации диска: кнопка Пуск  - Все программы – Стандартные - Служебные - Дефрагментация диска.

4. Выделить том диска (D:).

5. Нажать кнопку Анализировать диск.

После – нажать кнопку Дефрагментация диска.

Программа анализирует состояние диска до дефрагментации - выводит результат в виде цифры – процентной части нефрагментированных файлов – файлов, записанных на диске по частям. В результате проведения дефрагментации записанные по частям файлы будут собраны в единое целое. В зависимости от размера диска и хранящихся на нем данных процесс дефрагментации может занимать время до нескольких часов.

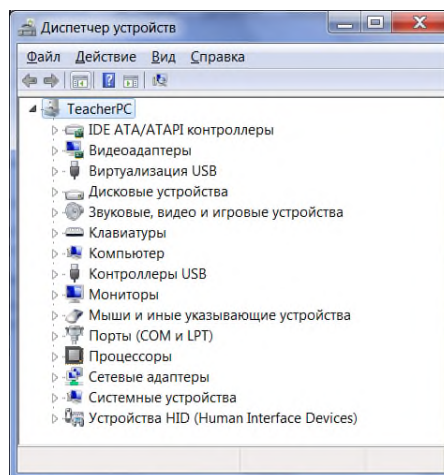
6. По завершении процедуры дефрагментации нажать кнопку Настроить расписание - настроить расписание дефрагментации – по понедельникам, в 9 утра.

Цель занятия: изучить элементы рабочего стола, панели задач, Главного меню; изучить возможности стандартных программ ОС Windows и дополнительные возможности операционной системы.

Содержание работы:

Элементы рабочего стола.

Рабочий стол – это графическая среда, на которой отображаются объекты Windows и элементы управления Windows. *Панель задач* – тоже очень важный элемент управления.

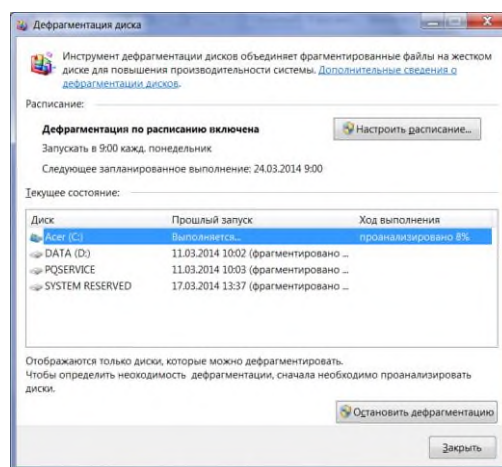


это?) :

(что это?) :
это?) :

жестких

диску C:



На Рабочем столе находятся *ярлыки* (значки, пиктограммы), которые делятся на две группы: *системные* и *пользовательские*.

Системные входят в состав операционной системы и обязательно находятся на Рабочем столе. Это следующие значки: Мой компьютер, Мои документы, Корзина, Сетевое окружение, InternetExplorer. Их, как правило, нельзя переименовать и сменить им значок.

Пользовательские ярлыки создаются пользователями ПК для быстрого доступа к файлам, папкам и быстрого открытия программ. Чаще всего пользовательский ярлык отличается чёрной стрелкой в углу.

1. *Мой компьютер*. С помощью этого значка можно просмотреть любые диски (в том числе CD, DVD, дискеты, флеш-карты и другое оборудование), разыскать на них нужные папки и файлы и поработать с ними.
2. *Корзина* – специальный объект Windows, выполняющий функцию контейнера. Она служит для временного хранения удаляемых объектов. Любой объект можно восстановить из Корзины, причём он восстановится в ту папку, из которой был удалён. Если щёлкнуть на значке Корзины *правой кнопкой мыши* и в контекстном меню выбрать команду *Свойства*, то можно настроить различные параметры. Например, флажок «Уничтожать файлы сразу после удаления, не помещая их в корзину» означает, что файлы и папки будут удаляться сразу, не попадая в Корзину, и не смогут быть восстановлены.
3. *Мои документы*— это личная папка пользователя. В ней содержатся две специализированные личные папки: «Мои рисунки» и «Моя музыка». Можно открыть доступ к личным папкам для всех пользователей, имеющих учетную запись на этом компьютере, или сделать эти папки частными; при этом файлы в них будут доступны только для данного пользователя. Windows создает личные папки для каждого пользователя компьютера. В Windows предоставляется также папка «Общие документы» для размещения файлов, доступных другим пользователям.
4. *Сетевое окружение*. Если несколько компьютеров объединены в локальную сеть, то этот значок поможет посмотреть информацию на соседних компьютерах и поработать с ней. Если компьютер автономный, то этот значок не нужен.
5. *InternetExplorer*. Это *программа – браузер*, предназначенная для работы в Интернете и просмотра Веб-страниц. Она удобна тем, что уже входит в комплект программ Windows. Есть более удобные программы – браузеры, но их нужно устанавливать дополнительно.


На Панели задач обязательно находится кнопка *Пуск*, с помощью которой открывается *Главное меню*, индикаторы звука, клавиатуры, времени, дата и другие значки, которые может настраивать пользователь. На панели задач отображаются кнопки открытых документов и программ, поэтому ОС Windows называется *многозадачной ОС*.

Главное меню содержит две части: обязательную и необязательную. Необязательная часть располагается над серой чертой, в неё добавляются нужные пользователю команды. В обязательной части находятся следующие команды:

- *Программы*. Можно открыть любую программу, установленную на компьютере.
- *Документы*. Список недавно открывавшихся документов, откуда можно быстро открыть документ или просмотреть последние открытые документы.
- *Найти*. (Поиск) Чаще всего эта команда используется для поиска нужных файлов и папок.
- *Справка и поддержка*. Вызов встроенной справочной системы ОС Windows.
- *Выполнить*. С помощью этой команды можно запустить любую программу, если известно, как называется её файл запуска.
- *Настройка*. Позволяет настроить экран, мышь, клавиатуру, принтеры, сканеры, ввести пароли для пользователей системы и т.д.

Работа с окнами.

Любое окно в операционной системе Windows имеет три кнопки управления в верхнем

правом углу: . С помощью этих кнопок можно свернуть, развернуть окно, восстановить его предыдущие размеры, закрыть его. Замечание: чтобы развернуть окно или восстановить его размеры, можно также дважды щелкнуть строку заголовка окна.

ОС Windows позволяет работать с несколькими открытыми окнами одновременно, поэтому говорят, что эта операционная система имеет *многооконный интерфейс*. Все окна свернуты на *панели задач*, откуда их можно быстро развернуть.

Расположение окон на экране относительно друг друга можно менять. Активным считается окно с выделенным заголовком или нажатой кнопкой на панели задач. Чтобы *изменить расположение* всех открытых окон:

1. Щелкните правой кнопкой мыши пустое место на панели задач.
2. Выберите команду *Окна каскадом*, *Окна слева направо* или *Окна сверху вниз*.

Замечание:

1. Окна, свернутые в кнопки на панели задач, на экране не отображаются.
2. Чтобы восстановить предыдущие размеры и положение окон, щелкните правой кнопкой пустое место на панели задач и выберите команду *Отменить Каскадом* или *Отменить Окна рядом*.
3. Чтобы свернуть все окна в кнопки на панели управления нажмите правую кнопку мыши на панели задач и выберите команду *Свернуть всё*.

Чтобы изменить размеры открытого окна наведите указатель на его левую или правую границу или на угол. Когда указатель примет форму горизонтальной двухсторонней стрелки, перетащите границу влево или вправо.

Замечание: размеры развернутого окна (в полноэкранном режиме) изменить нельзя.

Создание снимка экрана. Чтобы создать копию активного окна, нажмите клавиши ALT+PRINT SCREEN. Чтобы скопировать весь экран в том виде, как он отображается на мониторе, нажмите клавишу PRINT SCREEN. Чтобы вставить полученное изображение в документ, откройте меню Правка в окне документа и выберите команду Вставить.

Атрибуты файлов.

Кроме имени и *расширения* файла операционная система хранит для каждого файла дату его создания (изменения) и несколько величин, называемых *атрибутами* файла. Атрибуты – это дополнительные параметры, определяющие свойства файлов. Операционная система позволяет их контролировать и изменять, состояние атрибутов учитывается при выполнении операций с файлами. В ОС Windows есть следующие виды атрибутов:

1. атрибут файла «только для чтения» (read-only) предохраняет файл от изменений: для изменения или удаления файла с этим атрибутом требуется предварительно снять данный атрибут. Файлы на компакт-дисках также имеют атрибут «только для чтения», чтобы показать, что изменить эти файлы нельзя;
2. атрибуты «скрытый» (hidden) и/или «системный» (system) используются некоторыми системными файлами (например, основные файлы операционной системы MSDOS — IO.SYS и MSDOS.SYS, — имеют оба этих атрибута). Файлы с атрибутом «системный» не перемещаются программами оптимизации расположения файлов на диске (типа SpeedDisk), также обычно не копируются на сжатый диск при создании сжатого диска из обычного программами типа DriveSpace;
3. атрибут файла «архивный» (archive) устанавливается при создании файла и сбрасывается программами резервного копирования для обозначения того, что копия файла помещена в архив. Поэтому наличие атрибута «архивировать» обычно значит, что для файла не было сделано резервной копии.

Таким образом, большинство файлов имеет установленным только атрибут «архивный». Остальные атрибуты («только для чтения», «скрытый» или «системный»), как правило, не установлены.

Стандартные программы.

В операционную систему Windows входит ограниченный набор прикладных программ, с помощью которых можно решать простейшие задачи, если на ПК не установлены более мощные средства. Это *Стандартные программы*. Перечислим некоторые из них:

- *Блокнот*. Простейший текстовый редактор, который можно использовать в качестве удобного средства просмотра текстовых файлов;
- *Графический редактор Paint*. Это простейшая программа для создания и редактирования изображений. Она не соответствует современным требованиям графических программ, но является очень простой и доступной, позволяющей освоить основные приёмы работы с графикой;
- *Текстовый процессор WordPad*. Как и текстовый редактор Блокнот, служит для создания, редактирования и просмотра текстовых документов, но выполняет ещё одну важную функцию – *форматирование* документа. Под форматированием понимают применение нескольких шрифтовых наборов, методы выравнивания текста, встраивание в документ рисунков и других объектов, их обтекание текстом и т.д.;
- *Калькулятор*. Удобная программа для математических расчётов.

Замечание: мы рассмотрели наиболее популярные стандартные программы, остальные рассмотреть самостоятельно с помощью справочной системы!

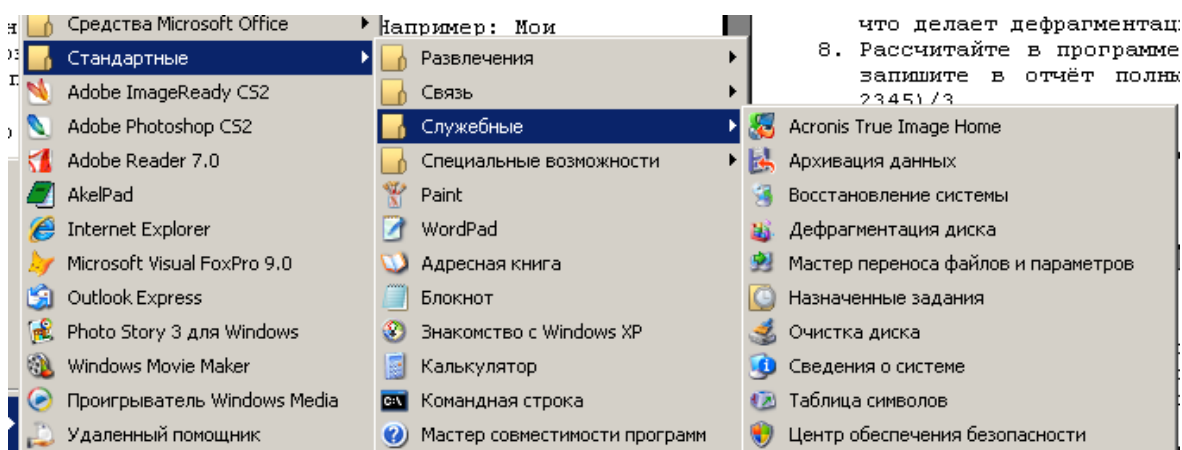
служебные программы

Служебные приложения предназначены для обслуживания персонального компьютера и самой операционной системы. Они позволяют находить и устранять дефекты файловой системы, оптимизировать настройки программного и аппаратного обеспечения, также автоматизировать некоторые рутинные операции, связанные с обслуживанием компьютера.

Эти приложения открываются командой *Пуск/Программы/Стандартные/Служебные*. Они поставляются в составе операционной системы и устанавливаются вместе с ней.

Замечание: назначение служебных программ изучить самостоятельно, пользуясь лекционным материалом и дополнительной литературой!

Рис. 4



ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Изучите ярлыки на рабочем столе.
2. Изучите элементы панели задач.
3. Выведите и скройте *Панель быстрого запуска*.
4. Откройте одновременно несколько окон. Например: Мои документы, Корзина и Мой компьютер.
5. Попробуйте подвигать их по экрану, свернуть, развернуть, измените их размеры с помощью мыши и добейтесь приблизительно одинаковых размеров.

6. Расположите открытые окна каскадом, сверху вниз, слева направо. Закройте все окна.
7. Подвигайте ярлыки на Рабочем столе, затем упорядочьте их автоматически.
8. Откройте стандартную программу *Блокнот* и напечатайте в ней какое-либо четверостишие. Сохраните в своей папке под именем *Стихотворение*. Обратите внимание, с каким значком сохранился документ.
9. Скопируйте текст стихотворения в окно программы WordPad и посмотрите, какие возможности доступны в этой программе. Сохраните в своей папке под именем Копия и обратите внимание, с каким значком сохранился документ.
10. Откройте стандартную программу *Paint* и нарисуйте в ней ёлочку. Сохраните в своей папке под именем *Ёлка*. Обратите внимание, с каким значком сохранился документ.
11. Щёлкните правой кнопкой мыши на значках файлов Стихотворение и Ёлка. Выберите в контекстном меню команду Свойства и изучите содержимое открывшегося диалогового окна. Какие атрибуты заданы для этих файлов?
12. Изучите, какие ещё Стандартные программы есть на Вашем ПК.
13. Откройте Адресную книгу в Стандартных программах и создайте в ней две папки: Друзья и Коллеги.
14. В каждой папке создайте по два контакта.
15. Попробуйте через *поиск на Панели инструментов* в Адресной книге найти добавленного Вами в Контакты друга.
16. Изучите остальные Стандартные программы.
17. В Справочной системе ОС Windows посмотрите назначение неизвестных Вам программ.

Требования к отчёту:

1. Запишите, какие значки и индикаторы есть на Панели задач?
2. Запишите, какие ярлыки на Рабочем столе являются пользовательскими, а какие системными?
3. Опишите, что нужно сделать, если папки и файлы удаляются в Корзину, но она пуста?
4. Запишите свойства файлов Стихотворение и Ёлка. (пункт 10)
5. Запишите, какие возможности форматирования текста есть в программе Блокнот.
6. Запишите, чем отличаются программы Блокнот и WordPad.
7. Найдите в Справочной системе и запишите ответ на вопрос, что делает дефрагментация диска?
8. Рассчитайте в программе Калькулятор следующее выражение и запишите в отчёт полным ответом: $(23456 + (34789 - 5432) * 2 - 2345) / 3$.
9. Запишите, какие программы представлены в пункте Стандартных программ Специальные возможности.

Контрольные вопросы.

1. Какие элементы расположены на Рабочем столе?
2. Какие ярлыки называются системными и пользовательскими?
3. Для чего нужна Панель задач?
4. Какие элементы расположены на Панели задач?
5. Возможна ли настройка Панели задач?
6. Какие операции можно выполнять с окнами?
7. Как изменить расположение нескольких открытых окон на экране?
8. Для чего нужен значок Мой компьютер?
9. Для чего нужна Корзина?
10. Зачем нужен значок Сетевое окружение?
11. Зачем нужна программа *Internet Explorer*?
12. Что такое Атрибуты файлов?
13. Какие Стандартные программы Вы знаете?
14. Для чего нужны Стандартные программы?
15. Для чего используются служебные программы ОС Windows?

Цель занятия: научиться настраивать операционное окружение операционной системы.

Содержание работы:

1. Загрузите операционную систему Windows.
2. Создайте папку Практика на рабочем столе.
3. Откройте папку Практика.
4. Создайте в папке ярлык программы Paint (обзор: C:\WINDOWS\System32\mspaint.exe)[*].
5. Создайте ярлык программы Проводник (обзор: C:\WINDOWS\explorer.exe)
6. Перетягиванием перенесите (переместите) ярлык программы Калькулятор с рабочего стола в папку Практика.
7. Произведите любые арифметические и тригонометрические вычисления и закройте Калькулятор.
8. Загрузите графический редактор Paint через созданный ярлык.
9. Загрузите текстовый редактор WordPad через меню Пуск.
10. Переключитесь в окно Paint.
11. Вызовите диспетчер (список) задач и снимите (завершите) задачу WordPad без сохранения данных (завершение выполнения "зависших" программ).
12. Закройте окно Paint, используя системное меню окна.
13. Откройте свойства рабочего стола (экрана). Используя контекстную подсказку, изучите параметры настройки экрана.
14. Выберите любой фоновый рисунок из списка и нажмите кнопку Применить.
15. Измените оформление, выбрав любую цветовую схему из списка и нажмите кнопку ОК.
16. Верните прежнюю цветовую схему, выбрав Стандартная из списка и нажмите кнопку ОК.
17. Откройте свойства панели задач и изучите параметры (используя контекстную подсказку).

Цель занятия: познакомиться с модулями операционной системы; получить практические навыки управления загрузками операционной системы.

Содержание работы:

Наиболее общим подходом к структуризации операционной системы является разделение всех ее модулей на две группы:

1. Ядро □ это модули, выполняющие основные функции ОС.
2. Вспомогательные модули, выполняющие вспомогательные функции ОС. Одним из определяющих свойств ядра является работа в привилегированном режиме.

Модули ядра выполняют следующие базовые функции ОС:

- Управление процессами.—
- Управление системой прерываний.—
- Управление памятью.—
- Управление устройствами ввода-вывода.—

Ядро является движущей силой всех вычислительных процессов в компьютерной системе, и крах ядра равносителен краху всей системы, без него ОС является полностью неработоспособной и не сможет выполнить ни одну из своих функций. Поэтому разработчики операционной системы уделяют особое внимание надежности кодов ядра, в результате процесс их отладки может растягиваться на многие месяцы.

Например, в отличие от Windows ядро в Linux ведёт вполне самостоятельную жизнь и никак не связано с графической оболочкой: можно без проблем заменить одну оболочку на другую, не нарушая при этом никаких: внутренних связей.

Вспомогательные модули ОС выполняют вспомогательные функции ОС (полезные, но менее обязательные, чем функции ядра).

Примеры вспомогательных модулей:

Программа дефрагментации диска.—

Текстовый редактор простой.—

Вспомогательные модули ОС оформляются либо в виде приложений, либо в виде библиотек процедур. Вспомогательные модули ОС подразделяются на следующие группы:

- утилиты □ это сервисные программы, решающие задачи управления и сопровождения компьютерной системы: обслуживание дисков и файлов;
- программы предоставления пользователю дополнительных услуг пользовательского интерфейса (калькулятор, игры);
- системные программы (компоновщики, отладчики и т.д.);
- библиотеки различного назначения, упрощающие разработку приложений (библиотека математических функций).

Как и обычные приложения, для выполнения своих задач утилиты, обрабатывающие программы и библиотеки ОС, обращаются к функциям ядра посредством системных вызовов.

Функции, выполняемые модулями ядра, являются наиболее часто используемыми функциями операционной системы, поэтому скорость их выполнения

определяет производительность всей системы в целом. Для обеспечения высокой скорости работы ОС все модули ядра или большая их часть постоянно находятся в оперативной памяти, то есть являются резидентными.

Вспомогательные модули обычно загружаются в оперативную память только на время выполнения своих функций, то есть являются транзитными.

Такая организация ОС экономит оперативную память компьютера.

Загрузка операционной системы.

Начальной загрузкой называется сложный и многошаговый процесс запуска компьютера. Загрузочная последовательность □ это последовательность действий, которые должен выполнить компьютер для запуска операционной системы.

Файлы, составляющие ОС, хранятся на диске, поэтому система называется дисковой операционной (ДОС). Известно, что для их выполнения программы - и, следовательно, файлы ОС - должны находиться в оперативной памяти (ОЗУ). Однако чтобы произвести запись ОС в ОЗУ, необходимо выполнить программу загрузку, которой сразу после включения компьютера в ОЗУ нет. Выход из этой ситуации состоит в последовательной, поэтапной загрузке ОС в оперативную память.

Первый этап загрузки ОС. В системном блоке компьютера находится постоянное запоминающее устройство (ПЗУ, постоянная память, и т.д.), в котором содержатся программы тестирования блоков компьютера и первого этапа загрузки ОС. Они начинают выполняться с первым импульсом тока при включении компьютера. На этом этапе процессор обращается к диску, и проверяет наличие на определенном месте (в начале диска) очень небольшой программы - загрузчика. Если эта программа обнаружена, то она считывается в ОЗУ и ей передается управление.

Второй этап загрузки ОС. Программа - загрузчик, в свою очередь, ищет на диске базовый модуль ОС, переписывает его память и передает ему управление.

Третий этап загрузки ОС. В состав базового модуля входит основной загрузчик, который ищет остальные модули ОС и считывает их в ОЗУ. После окончания загрузки ОС управление передается командному процессору и на экране появляется приглашение системы к вводу команды пользователя. Заметим, что в оперативной памяти во время работы компьютера обязательно должен находиться базовый модуль ОС и командный процессор. Следовательно, нет необходимости загружать в оперативную память все файлы ОС одновременно. Драйверы устройств и утилиты могут подгружаться в ОЗУ по мере необходимости, что позволяет уменьшать обязательный объем оперативной памяти, отводимый под системное программное обеспечение.

Операционная система - это базовый комплекс компьютерных программ, обеспечивающий интерфейс с пользователем, управление аппаратными средствами

компьютера, работу с файлами, ввод и вывод данных, а также выполнение прикладных программ и утилит.

Рассмотрим управление запуском на примере Windows 7. Запуск программы Настройка системы

Чтобы запустить программу «Настройка системы», откройте меню Пуск, введите в поисковую строку msconfig и нажмите Ввод.

Также можно воспользоваться сочетанием клавиш Windows + R (Выполнить), ввести msconfig и нажать ОК.

Вкладка Общие

На вкладке Общие можно выбрать один из трех вариантов запуска операционной системы:

Обычный запуск

В этом режиме Windows 7 запускается обычным способом. «Обычный запуск» используется, когда нет проблем с загрузкой ОС или после устранения неполадок.

Диагностический запуск

В режиме диагностического запуска вместе с Windows запускаются только основные службы и драйверы, необходимые для функционирования операционной системы и компьютера. Если при включенном диагностическом запуске проблема не исчезает, то скорее всего, повреждены основные файлы и драйверы Windows. Если при включенном диагностическом запуске проблем нет, то нужно воспользоваться режимом Выборочный запуск.

Выборочный запуск

В этом режиме запуск Windows 7 производится с использованием основных служб и драйверов, а также других служб и автоматически загружаемых программ, выбранных пользователем. Доступны три дополнительных параметра:

Загружать системные службы ☐ если этот параметр включен, то операционная система загружается со стандартным набором служб, необходимых для её работы.

Загружать элементы автозагрузки ☐ если этот параметр включен, то вместе с операционной системой запускаются программы, отмеченные флажками на

вкладке Автозагрузка.

Использовать оригинальную конфигурацию загрузки ☐ этот параметр по умолчанию включен и затенен (отображается серым цветом). Данный параметр восстанавливает изначальные настройки запуска Windows 7 в случае внесения изменений на вкладке Загрузка.

Выборочный запуск нужно использовать, если диагностический запуск прошел без ошибок. Поочередно включайте дополнительные службы и программы, и отслеживайте работу системы до тех пор, пока не выявите причину ошибок.

Вкладка Загрузка

На вкладке Загрузка находятся детальные настройки параметров запуска Windows 7.

Операционная система по умолчанию

Если на компьютере установлено несколько операционных систем, можно назначить любую из них загружаемой по умолчанию. Чтобы сделать это, выделите нужную операционную систему и нажмите кнопку Использовать по умолчанию.

Также можно установить произвольное время задержки меню мультизагрузки, установив время в секундах в поле Таймаут.

Чтобы удалить операционную систему из списка, выделите её и нажмите кнопку Удалить.

Безопасный режим

Безопасный режим ☐ это режим работы операционной системы с ограниченным набором служб, устройств и драйверов, необходимых для функционирования компьютера.

Службы Windows, запускающиеся в безопасном режиме:

Журнал событий Windows

Поддержка самонастраивающихся устройств Plug and Play
Удаленный вызов процедур (RPC)
Службы криптографии
Защитник Windows
Инструментарий управления Windows (WMI)

Цель занятия: научиться выполнять конфигурирование аппаратных устройств.

Содержание работы:

Everest — программа для просмотра информации об аппаратной и программной конфигурации компьютера, разработанная компанией Lavalys. Являлась последовательницей AIDA32, заменена на рынке программой AIDA64.

Возможности программы:

- Программа анализирует конфигурацию компьютера и выдаёт подробную информацию;
- об установленных в системе устройствах — процессорах, системных платах, видеокартах, аудиокартах, модулях памяти и т. д.;
- их характеристиках: тактовая частота, напряжение питания, размер кэшей, и т. д.;
- поддерживаемых ими наборах команд и режимах работы;
- их производителях;
- установленном программном обеспечении;
- конфигурации операционной системы;
- установленных драйверах;
- автоматически загружаемых программах;
- запущенных процессах;
- имеющихся лицензиях;

AIDA32 — в прошлом популярная бесплатная программа для просмотра информации об аппаратной и программной конфигурации компьютера. Её развитием стали платные программы Everest (2004-2010) и AIDA64 (с 2010).

Программа анализирует конфигурацию компьютера и выдаёт подробную информацию:

- Об установленных в системе устройствах — процессорах, системных платах, видеокартах, аудиокартах, модулях памяти и т. д.
- их характеристиках: тактовая частота, напряжение питания, размер кэшей, и т. д.
- температура центрального процессора (CPU) (выбор (нажать "+") на панели слева: "Компьютер"—>"Датчик", Температуры, ЦП)
- поддерживаемых ими наборах команд и режимах работы
- их производителях
- установленном программном обеспечении
- конфигурации операционной системы
- установленных драйверах
- автоматически загружаемых программах
- запущенных процессах
- имеющихся лицензиях
- В программе есть тест записи в память и чтения из памяти с возможностью сравнения их результатов с эталонными.

SpeedFan — компьютерная программа, следящая за показателями датчиков материнской платы, видеокарты и жёстких дисков, с возможностью регулирования скорости вращения установленных кулеров.

Отображаемые параметры

- Температура центрального микропроцессора (CPU) (выбор в меню: "Показатели", Temp2 в первой тройке датчиков температуры)
- Скорость вентилятора кулера центрального микропроцессора (CPU) (выбор в меню: "Показатели", Fan2 в первой тройке датчиков скорости вентиляторов)
- График температуры центрального микропроцессора (CPU) (выбор в меню: "Графики"—>"температуры"—>"Temp2" в первой тройке датчиков температуры)
- График скорости вентилятора кулера центрального микропроцессора (CPU) (выбор в меню: "Графики"—>"скорости вентиляторов"—>"Fan2" в первой тройке датчиков скорости вентиляторов)

Регулируемые параметры

- Управляет скоростями вращения установленных кулеров. Может регулировать вращение, как автоматически так и вручную. В последних версиях добавлена поддержка технологии S.M.A.R.T.

Используя программу Everest получите и выпишите в тетрадь нижеперечисленные сведения об аппаратной части вашего ПК:

1. Марка\модель центрального процессора (ЦП)
2. Наборы инструкций ЦП
3. Тактовая частота ЦП
4. Объём кэш-памяти первого, второго и третьего уровней
5. Производителя и модель системной платы, дату выпуска
6. Все свойства шины FSB;
7. Вся физическую информацию о системной плате
8. Все свойства модулей оперативной памяти
9. Найти и выписать все данные о видеоадаптере
10. Все данные о BIOS
11. Данные о мониторе (Модель, тип, дата выпуска, Соотношение сторон, максимальное разрешение)
12. Звуковой адаптер (Производителя, марку, модель, тип шины)
13. Жёсткие диски (Производитель, модель, интерфейс подключения, количество цилиндров, головок, секторов в треке, байт в секторе, размер буфера, неформатированная ёмкость)
14. Сетевые адаптеры (Производитель, модель, аппаратный адрес, максимальная пропускная способность)
15. Основные данные о клавиатуре и мыши
16. Температура устройств (Температура процессора, жёсткого диска, количество оборотов вентилятора на процессоре)
17. Произвести тест оперативной и кэш-памяти, результаты выписать в тетрадь

Контрольные вопросы:

1. Перечислите возможности программы Everest
2. Для чего предназначена программа SpeedFan?

Цель занятия: познакомиться с основами администрирования учетных записей операционной системы Windows.

Содержание работы:

Для входа в систему Windows 10 нужно иметь заранее созданную учетную запись. Если рабочая станция включена в рабочую группу, а не в домен, эта учетная запись должна храниться на этой станции. Чтобы войти в домен, необходима учетная запись в этом домене (или в доверенном домене).

Учетная запись содержит не только имя и другие данные, идентифицирующие пользователя, но и определяет, каким образом пользователь регистрируется, когда он

может входить в систему, какие ресурсы ему доступны и каков его уровень доступа к этим ресурсам. Другими словами, учетная запись определяет все аспекты доступа к компьютеру и в сеть. Кроме того, каждая учетная запись содержит пароль, обеспечивающий ее безопасное использование.

Права пользователя определяются не только учетной записью, но и членством в группах. **Группа** – это совокупность пользователей, выполняющих сходную работу и имеющих примерно одинаковые потребности в ресурсах.

Windows XP поддерживает два типа групп.

– **Глобальные группы.** Глобальная группа может содержать учетные записи пользователей только того домена, где она создана. Права доступа к ресурсам других доменов глобальные группы получают в рамках доверительных отношений между доменами.

– **Локальные группы.** Локальная группа может иметь в своем составе как индивидуальных пользователей, так и глобальные группы. Это позволяет объединять в одной локальной группе пользователей из различных доменов и управлять ими коллективно. Права локальной группы распространяются только на тот домен, где она создана.

В подавляющем большинстве группы используются, чтобы упростить контроль за доступом к общим ресурсам. Сначала создается группа и задается ей право доступа к конкретному ресурсу, а затем включается сюда пользователь, которому этот ресурс необходим. Позже, когда потребуются изменить уровень доступа к ресурсу (например, наложить ограничения на его изменение), будет достаточно модифицировать права группы, и все ее члены наследуют новые права на общий ресурс. Это значительно легче, чем изменять привилегии каждой индивидуальной учетной записи в группе.

Доверительные отношения

Доверительным называют особые логические отношения между доменами, при которых один домен доверяет пользователям другого домена. Домен-доверитель открывает пользователям доверенного домена доступ к своим ресурсам. Если, например, домен А является доменом-доверителем, а домен Б – доверенным доменом, то пользователь домена Б может работать и с ресурсами домена А.

Доверительные отношения могут быть как односторонними, так и двусторонними. То, что домен А доверяет домену Б, вовсе не означает, что домен Б доверяет домену А. Для установления двусторонних доверительных отношений необходимо явно указать, что домен А доверяет домену Б.

Встроенные группы

Windows XP содержит целый ряд встроенных учетных записей и групп. Одна из них – учетная запись администратора, которая создается при установке операционной системы. Учетную запись администратора нельзя удалить или отключить, ее можно только переименовать. В число встроенных входит и учетная запись гостя, которую также нельзя удалить, но можно переименовать или отключить. Эта запись применяется для регистрации в компьютере без использования специально созданной учётной записи. Она не требует ввода пароля и по умолчанию заблокирована.

Администраторы (Administrators). Члены группы “Администраторы” получают практически неограниченный доступ к ресурсам домена, сервера или рабочей станции, где находится группа, и полную власть над ними. Они могут также использовать любые файлы и каталоги раздела FAT, однако полного доступа к каталогам файловой системы NTFS автоматически не получают. Право доступа им должен предоставить владелец файла или каталога, в противном случае даже администратор не может использовать этот ресурс. Правда, за администратором остается право присвоить себе любой ресурс и тем самым получить к нему полный доступ.

Операторы архива (BackupOperators). Эта локальная группа создана специально для архивирования файлов. Пользователи, входящие в группу “Операторы архива”,

получают право проводить резервное копирование и восстановление файлов, локально входить в систему и завершать ее работу. Однако операторы архива лишены возможности изменять параметры безопасности и выполнять другие административные задачи.

Опытные пользователи (PowerUsers). Опытные пользователи – это пользователи, которым предоставлены некоторые административные привилегии. Они могут создавать новые учетные записи и вносить изменения в те, которые создали сами. Кроме того, они могут включать учетные записи в группы пользователей, гостей и опытных пользователей, а также разрешать или запрещать совместное использование файлов и принтеров на локальных рабочих станциях и серверах.

Пользователи (Users). Это самая многочисленная группа. Члены этой группы могут выполнять большинство пользовательских функций, например, запускать приложения, пользоваться локальным или сетевым принтером, завершать работу системы или блокировать рабочую станцию. Они могут также создавать локальные группы и регулировать состав их членов.

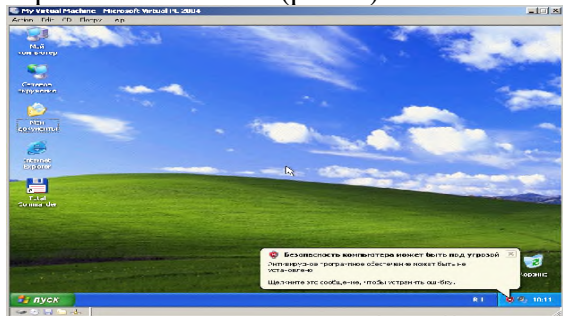
Гости (Guests). Члены этой группы обладают ограниченными правами на доступ к ресурсам системы и могут завершать работу системы. В отличие от пользователей гостям запрещен локальный вход на сервер, хотя они могут регистрироваться на нем через сеть.

Репликаторы (Replicator). Эта специальная группа создана, чтобы упростить тиражирование файлов и каталогов.

Главное в концепции учетных записей и групп состоит в том, что они упрощают администрирование ресурсов и доступ к ним. Вместо того чтобы задавать конкретные права каждому пользователю, можно создать группу с этими правами, а затем по мере необходимости включать в нее пользователей. Важно понимать, что пользователь может быть членом нескольких групп и что одна группа способна включать в себя другую.

Оснастка “Локальные пользователи и группы”

Перед началом работы произвести запуск виртуальной машины *Oracle VM VirtualBox*. Для этого воспользуйтесь либо ярлыком программы **Oracle VM VirtualBox** на рабочем столе. После запуска программы выбрать необходимый образ виртуальной машины и нажать кнопку **Start**. В результате этого будет запущен процесс эмуляции персональной ЭВМ (рис. 2).



Назначение и состав оснастки “Локальные пользователи и группы”

Оснастка “Локальные пользователи и группы” – это основной инструмент консоли управления MMC, с помощью которого выполняется управление локальными учётными записями пользователей и групп – как на локальном, так и на удалённом компьютере. Запускать оснастку может любой пользователь, однако выполнять администрирование учётных записей могут только администраторы и члены группы “Опытные пользователи”.

Запустить оснастку можно добавив её стандартным способом в консоли управления, либо введя команду “*lusrmgr.msc*” в окне запуска программ (**Пуск > Выполнить...**). На экране появится окно, представленное на рис 3.

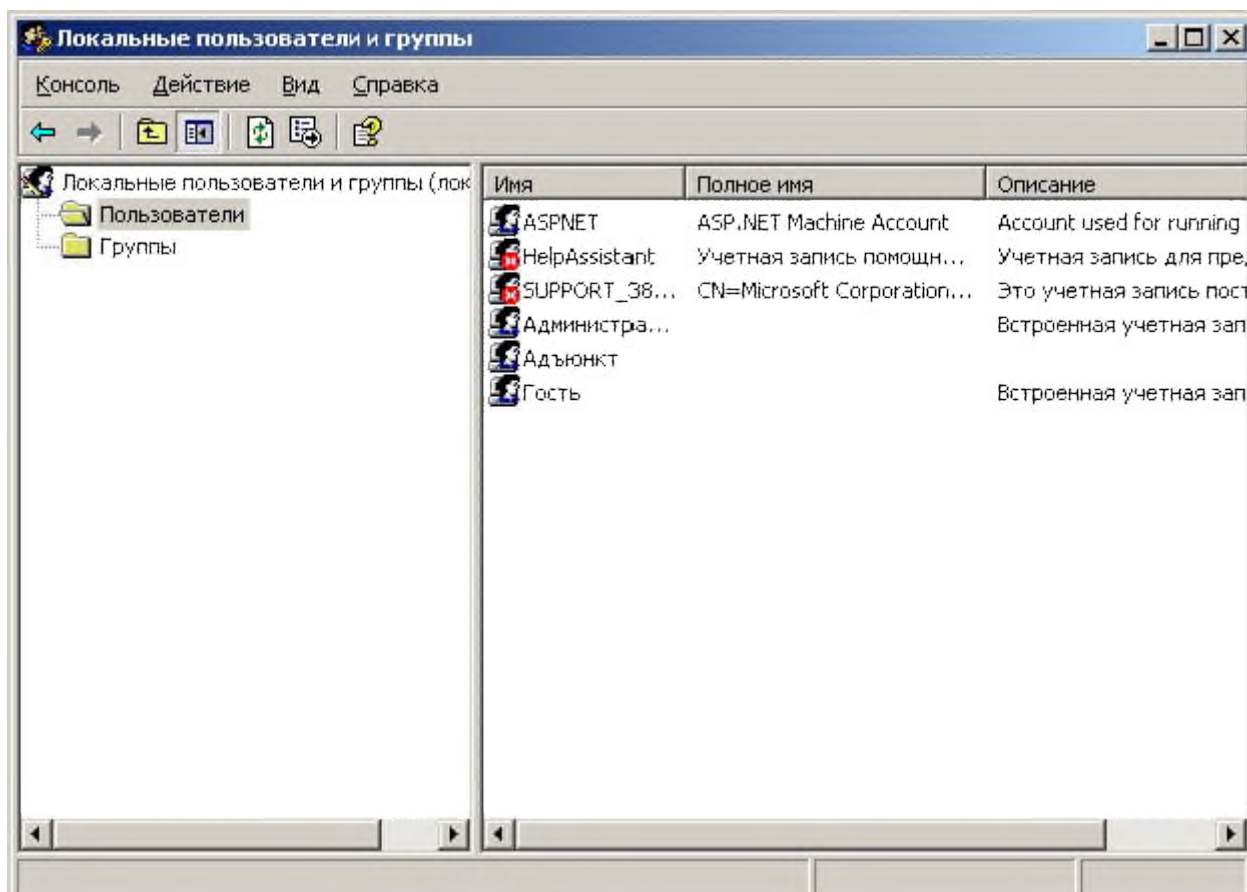


Рис. 3. Оснастка “Локальные пользователи и группы”

Оснастка “Локальные пользователи и группы” позволяет выполнять следующие административные функции:

- создание и изменение учетных записей пользователей;
- создание и изменение групп;
- определение политики учетных записей;
- предоставление прав пользователям;
- настройка аудита пользователей;
- создание и изменение доверительных отношений.

Выбор пользователей

Чтобы выбрать одну запись, необходимо просто щелкнуть на ней левой кнопкой “мыши”. Если же нужно выделить несколько учетных записей (что позволит внести изменения сразу во все выделенные записи), необходимо щелкнуть на первой из них, затем нажать клавишу <Ctrl> и, не отпуская ее, поочередно щелкнуть на остальных

Создание учетных записей

Создавая учетную запись, необходимо ввести такие общие данные, как имя пользователя и пароль, а также задать разрешения и определить другие ограничения на вход в систему.

Ввод общих данных.

Чтобы начать создание новой учетной записи, выберите каталог **Пользователи (Users)**, войдите в меню **Действие (Action)** и щелчком на пункте **Новый пользователь (New User)** вызовите диалоговое окно, изображенное на рис. 4. В графе **Пользователь (Username)** введите имя учетной записи, которое пользователь должен будет указать при регистрации на компьютере или в домене.

Рис. 4. Ввод имени пользователя и пароля

В графе **Полное имя (FullName)** укажите реальное имя владельца учетной записи, а в графе **Описание (Description)** – дополнительные данные в произвольной форме (например, должность владельца или местоположение его офиса). Теперь нужно назначить пароль, который пользователь должен будет указать вместе с именем учетной записи при входе в систему. Это делается в два этапа. Сначала введите его в графе **Пароль (Password)**, а затем – в графе **Подтверждение (ConfirmPassword)**.

Под текстовыми полями диалогового окна **Новый пользователь (NewUser)** находятся четыре переключателя, определяющие общие свойства учетной записи. “**Потребовать смену пароля при следующем входе в систему**” (**User**

MustChangePasswordAtNextLogon). Пометьте эту кнопку флажком, если хотите, чтобы пользователь сменил свой пароль. При очередном его входе в систему Windows XP потребует назначить новый пароль.

“**Запретить смену пароля пользователем**” (**UserCannotChangePassword**).

Если пометить флажком эту кнопку, пользователь не сможет менять свой пароль самостоятельно.

“**Срок действия пароля не ограничен**” (**PasswordNeverExpires**). Пометьте эту кнопку флажком, если хотите, чтобы пароль пользователя имел неограниченный срок действия. Ни вам, ни пользователю это не мешает; если возникнет такая необходимость, сменить пароль.

“**Отключить учетную запись**” (**AccountDisabled**). Пометьте эту кнопку флажком, если хотите заблокировать учетную запись. Например, можно создать учетную запись-шаблон, которую будет применяться для создания других учетных записей, а чтобы ею не воспользовался недоброжелатель – отключить ее.

Лучше заблокировать и те учетные записи, которые не будут использоваться в течение некоторого времени.

Включение в группы.

Кроме ввода общих данных необходимо также указать, членом каких групп станет пользователь. Для этого двойным щелчком мыши по названию учетной записи вызовите окно **Свойства (Properties)**. Во вкладке **Член групп (MemberOf)**, представленная на рис. 5, указано, в какие группы на данный момент включена учетная запись.

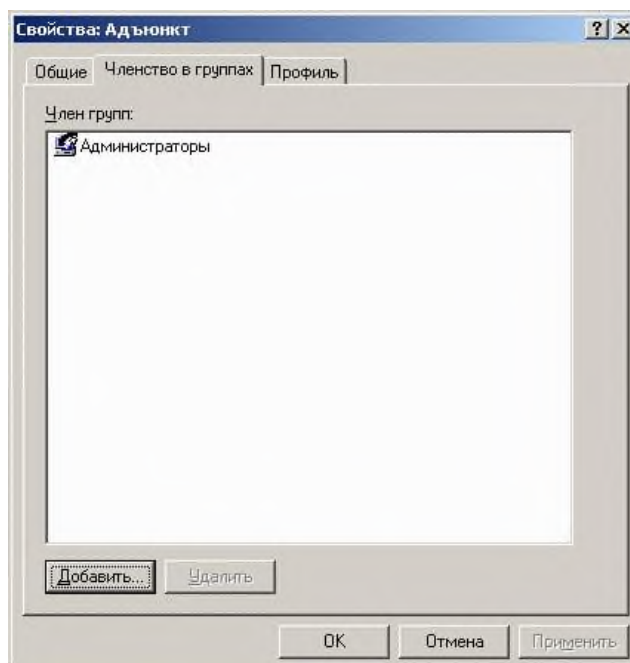


Рис.5. Вкладка **Членство в группах** свойств Учетной записи

Чтобы включить учетную запись в любую из учетных групп щелкните на кнопке **Добавить...** (Add...). Таким образом будет вызвано окно **Выбор: Группы** (SelectUsersofGroups), представленное на рис. 6.

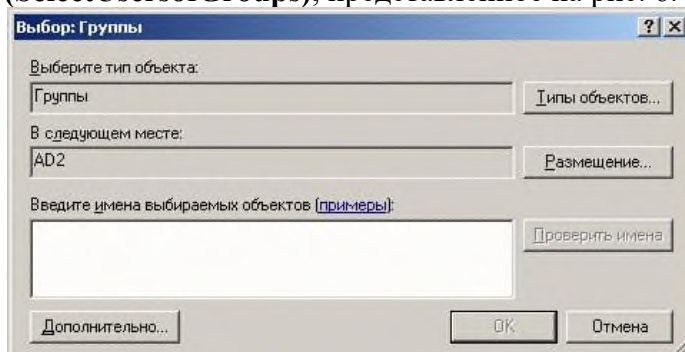


Рис.6. Окно **Выбор: Группы**

В поле **Введите имена выбираемых объектов (Enterthenamestoselect)** необходимо ввести название группы, в которую необходимо добавить учетную запись пользователя. Для упрощения ввода необходимо нажать на кнопку **Дополнительно...** (Advanced...), и в появившемся окне (рис. 7) произвести поиск требуемых названий, нажав кнопку **Поиск (Find)**.

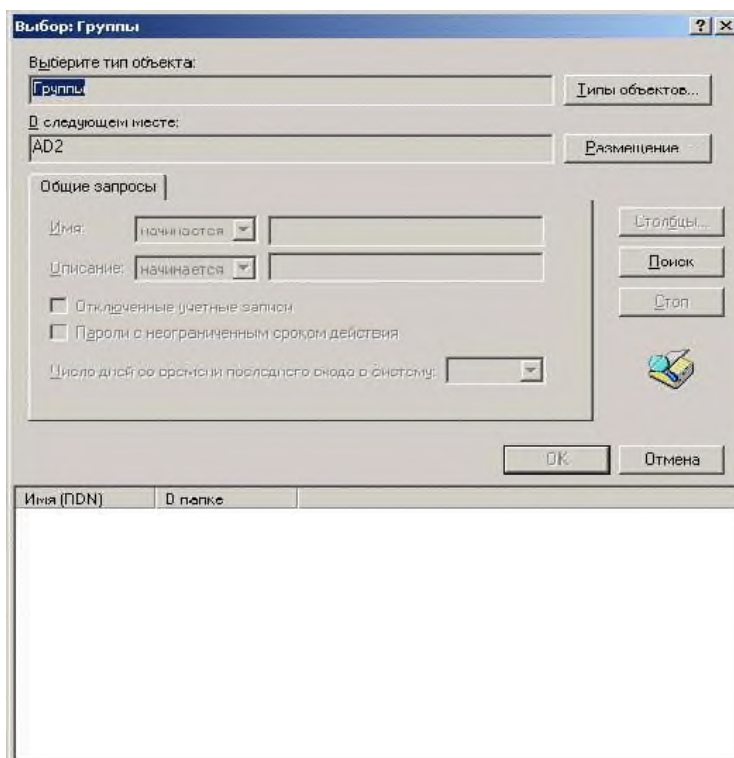


Рис.7. Дополнительное окно **Выбор: Группы**

Если нужно исключить пользователя из какой-либо группы, выделите имя группы в поле **Член групп (MemberOf)** и щелкните на кнопке **Удалить (Remove)**.

После того как вы включили учетную запись во все необходимые группы, щелкните на кнопке **ОК**.

Возможен и обратный способ – не выбирать группу для добавления в них учетной записи, а выбор учетной записи для ее добавления в группу. Для этого необходимо вызвать окно **Свойства группы (Properties)**, а затем выполнить действия, аналогичные вышеуказанному.

Выбор профиля.

На следующем этапе создания учетной записи нужно указать профиль пользователя, основной каталог и сценарий входа в систему. Эти параметры указываются во вкладке **Профиль (Profile)**, представленная на рис. 8.

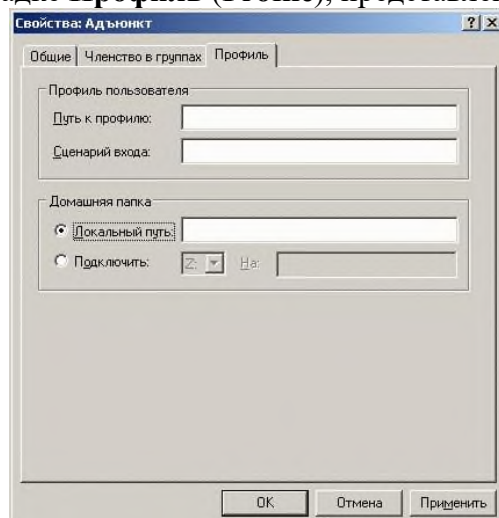


Рис.8. Вкладка **Профиль** свойств Учетной записи

Профилем называют специальный файл, в котором Windows XP сохраняет установленные пользователем параметры настройки: свойства рабочего стола, цветовую и звуковую схемы и некоторые другие. Кроме того, на каждом компьютере имеется

стандартный профиль пользователя, который используется по умолчанию, если при входе пользователя в систему не указан другой профиль.

После этого стандартный профиль становится пользовательским.

Для пользователя, который может подключаться к сети с различных узлов, создается перемещаемый профиль. Его нужно сохранять на сетевом сервере, где он будет доступен пользователю при каждом входе в систему. Благодаря этому пользователь, где бы он ни находился, сразу попадет в собственную рабочую среду.

Чтобы указать профиль, который будет использоваться вместе с учетной записью, введите в графе **Путь (UserProfilePath)** раздела **Профили пользователей (Userprofiles)** полное составное имя профиля. Например, вы можете воспользоваться стандартом UNC (UniversalNameConvention – универсальное соглашение об именовании) и указать путь в виде `\\server\profiles\billg.usr`. Если вы имеете дело с несколькими учетными записями, можете заменить имя пользователя на переменную среды. Допустим, вы работаете с шестью различными учетными записями и поэтому не можете указать единый файл для всех профилей. В этом случае введите примерно такую строку: `\\server\profiles\%username%.usr`. Диспетчер пользователей при входе пользователя в систему подставит вместо переменной `%username%` его имя и найдет (или создаст) индивидуальный профиль данного пользователя.

В текстовой графе **Сценарий входа (LogonScriptName)** укажите имя файла, содержащего сценарий входа в систему для данной учетной записи. Вы можете ввести здесь имя пакетного (.BAT), командного (.CMD) или исполняемого (.EXE) файла. Сценарии входа в систему обычно хранятся в подкаталоге `\\SYSTEM32\Repl\Import\Scripts`.

Раздел **Домашняя папка (HomeDirectory)** диалоговой вкладки **Профиль** позволяет указать начальный каталог, в который пользователь попадает сразу после регистрации. Как правило, в качестве основного каталога пользователя назначается тот, где хранятся его личные файлы.

Основной каталог может располагаться как на локальном компьютере, так и на сервере. Чтобы указать локальный каталог, щелкните на кнопке **Локальный путь (LocalPath)** и введите в расположенном рядом поле имя локального диска и каталога, например, `C:\Users\Fredf`.

Чтобы указать в качестве основного каталога общий сетевой каталог, щелкните на кнопке **Подключить (Connect)**, а затем выберите в ниспадающем меню буквенное обозначение диска и введите в соседнем текстовом поле имя каталога в формате UNC, например, `\\Server\Users\Fredf`.

Если вы работаете с несколькими учетными записями, можете использовать вместо имени каталога переменную среды `%username%`. В процессе регистрации Диспетчер пользователей заменит ее на имя пользователя.

Создание групп.

Диспетчер пользователей позволяет вам создавать и изменять не только учетные записи пользователей, но и группы, в которые затем можно добавить существующие учетные записи.

Создание локальной группы. Для создания локальной группы выберите каталог **Группы (Groups)**, откройте меню **Действие (Action)** и щелчком на пункте **Создать группу (NewGroups)** вызовите диалоговое окно, изображенное на рис. 9.

В текстовом поле **Имя группы (GroupName)** укажите имя, которое вы хотите присвоить новой группе. В нем можно использовать любые буквы верхнего и нижнего регистров, а также все символы, за исключением следующих:

`" / \ [] : ; | = , + * ? < >`

В текстовой графе **Описание (Description)** вы можете охарактеризовать создаваемую группу, хотя делать это не обязательно.

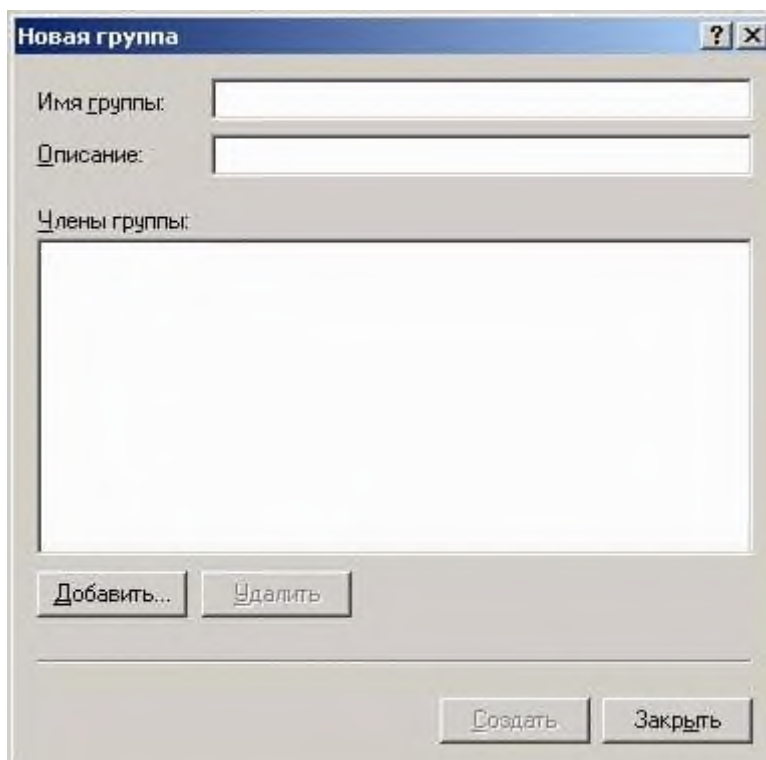


Рис. 9. Диалоговое окно **Новая группа**

Чтобы включить пользователей в группу, щелчком на кнопке **Добавить...**

(**Add...**), так же вызывающее диалоговое окно **Выбор: Пользователи** (**SelectUsersofGroups**), показанное при создании группы (рис. 6).

Права доступа и их присвоение

В Windows XP при защите папок, файлов, принтеров и прочих ресурсов очень важную роль играют разрешения, тесно связанные с учетными записями.

Задавать разрешения можно двумя различными способами. Во-первых, каждому объекту можно назначить собственный набор разрешений (их часто называют *объектными разрешениями*), которые регулируют локальный доступ к ресурсу.

Кроме локальных объектных разрешений вы можете задать и *разрешения на общий ресурс*, действие которых распространяется на ресурсы совместного использования, размещенные на компьютере. Эти разрешения определяют уровень доступа удаленных пользователей к такому ресурсу по сети.

Права определяют возможность пользователя (или группы) выполнять конкретные действия. Войдя в систему, пользователь может выполнять только операции, определенные правами его учетной записи, которые либо были назначены ей непосредственно, либо получены через членство в группе. Ниже приведен список конкретных прав пользователей.

– **Архивирование файлов и каталогов (Backupfilesanddirectories)**. Это право позволяет выполнять резервное копирование файлов и каталогов на определенном компьютере и является преимущественным по сравнению с разрешениями на эти файлы и каталоги. Обратите внимание, что данное право не дает пользователю возможности просматривать содержимое файлов и каталогов, если только у него нет явного права на это.

– **Восстановление файлов и каталогов (Restorefilesanddirectories)**.

Имея такое право, пользователь может восстанавливать файлы с резервных копий. Обратите внимание: право на архивирование не означает, что пользователь может восстанавливать файлы, и наоборот.

– **Вход в качестве пакетного задания (Logonas a batchjob)**. Это право дает пользователю возможность входить в систему в качестве объекта пакетной очереди.

– **Вход в качестве службы (Logon as service).** Это право дает пользователю возможность выполнять функции безопасности и предназначено специально для того, чтобы процессы могли регистрироваться в системе как службы.

– **Добавление рабочих станций к домену (Addworkstationstodomain).**

Имея такое право, пользователь может добавлять в домен рабочие станции, благодаря чему они начинают признавать учетные записи и глобальные группы домена (что открывает возможность регистрации с этих рабочих станций). Задать это право вы можете, только работая с Диспетчером пользователей для доменов UserManagerforDomain.

– **Доступ к компьютеру из сети (Access this computer from network).** Это право дает пользователю возможность подключаться к компьютеру через сеть.

– **Завершение работы системы (Shutdownthesystem).** Это право дает пользователю возможность завершать работу системы.

– **Загрузка и выгрузка драйверов устройств (Loadandunloaddevice drivers).** Обладая этим правом, пользователь может загружать и выгружать драйверы устройств. Когда вы управляете доменом, это право относится к основному и резервному (резервным) контроллерам домена. За пределами домена это право применимо только к компьютеру, на котором оно задано.

– **Закрепление страниц в памяти (Lockpagesinmemory).** Имея такое право, пользователь может закреплять страницы в памяти, предотвращая тем самым их сброс на диск.

– **Замена маркера уровня процесса (Replace a processleveltoken).** Благодаря этому праву пользователь получает возможность изменять маркер доступа процесса.

– **Извлечение компьютера из стыковочного узла (Removecomputer fromdockingstation).** Определяет, может ли пользователь отключать переносной компьютер от стыковочного узла, не входя в систему. Если эта политика включена, пользователь должен войти в систему перед тем, как отключать компьютер от стыковочного узла. Если эта политика отключена, пользователь может отсоединять компьютер от стыковочного узла, не входя в систему.

– **Изменение параметров среды оборудования (Modifyfirmwareenvironment values).** Это право разрешает пользователю изменять переменные системной среды.

– **Изменение системного времени (Changethesystemtime).** Имея такое право, пользователь может настраивать системные часы.

– **Локальный вход в систему (Logonlocally).** Это право разрешает пользователю локально входить в систему. По соображениям безопасности вы едва ли захотите, чтобы пользователь локально регистрировался на сервере или на основном либо резервных контроллерах домена, хотя такая возможность тоже не исключена.

– **Настройка квот памяти для процесса (Increasequotas).** Определяет, какие учетные записи могут использовать процесс, обладающий разрешением «Запись свойства» для доступа к другому процессу, с целью увеличить назначенную последнему квоту ресурсов процессора. Данное право пользователя определено в объекте групповой политики стандартного контроллера домена, а также в локальной политике безопасности рабочих станций и серверов.

– **Обход перекрестной проверки (Bypassstraversechecking).** Пользователь, наделенный этим правом, может перемещаться по дереву каталогов даже в тех случаях, когда у него нет права на обход какого-либо каталога. Право на обход перекрестной проверки не заменяет прав на владение и разрешений. Если у пользователя нет необходимых разрешений на просмотр содержимого каталога, он его и не увидит, хотя пройти через этот каталог сможет.

– **Овладение файлами или иными объектами (Takeownershipoffilesorotherobjects).** Получив такое право, пользователь может овладевать файлами, каталогами и принтерами. Когда вы управляете доменом, это право

относится к основному и резервному (резервным) контроллерам домена. За пределами домена оно применимо только к компьютеру, на котором задано.

- **Отладка программ (Debugprograms).** Это право дает пользователю возможность отлаживать программы.
- **Отказ в доступе к компьютеру из сети (Denyaccesssthiscomputer fromthenetwork).** Определяет, каким пользователям запрещается доступ к данному компьютеру через сеть. Эта политика отменяет политику Доступ к компьютеру из сети, если учетная запись пользователя контролируется обеими политиками.
- **Отказ во входе в качестве пакетного задания (Denylogonas a batch job).** Определяет, какие учетные записи запрещается использовать при входе в систему в качестве пакетного задания. Эта политика отменяет политику Вход в качестве пакетного задания, если учетная запись пользователя контролируется обеими политиками.
- **Отказ во входе в качестве службы (Denylogonas a service).** Определяет, каким учетным записям запрещается регистрировать процесс в качестве службы. Эта политика отменяет политику Вход в качестве службы, если учетная запись пользователя контролируется обеими политиками.
- **Отклонить локальный вход (Denylogonlocally).** Определяет, каким пользователям запрещается вход в систему на данном компьютере. Эта политика отменяет политику Локальный вход в систему, если учетная запись пользователя контролируется обеими политиками.
- **Принудительное удаленное завершение (Forceshutdownfrom a remotesystem).** Это право предусматривает для пользователя возможность принудительно завершать работу системы с удаленного узла (например, по удаленному подключению).
- **Профилирование загрузки системы (Profilesystemperformance).** Пользователь, наделенный таким правом, имеет возможность профилировать общую производительность системы.
- **Профилирование одного процесса (Profilesingleprocess).** Это право позволяет пользователю профилировать отдельный процесс.
- **Работа в режиме операционной системы (Act as part of the operating system).** Это право дает пользователю возможность выступать в качестве доверенной части операционной системы и автоматически предоставляется некоторым подсистемам.
- **Разрешать вход в систему через службу терминалов (Denylogon throughterminalservices).** Определяет, каким пользователям и группам разрешается входить в систему в качестве клиента служб терминалов.
- **Разрешение доверия к учетным записям при делегировании (Enablecomputer and user accounts to be trusted for delegation).** Определяет, какие пользователи могут устанавливать атрибут Доверен для делегирования для объекта «Пользователь» или «Компьютер». Пользователь или объект, наделенный данной привилегией, должен иметь право записи во флаги управления учетной записью объекта «Пользователь» или «Компьютер». Серверный процесс, который работает на компьютере (или в контексте пользователя), доверенном для делегирования, может получать доступ к ресурсам другого компьютера, используя делегированные учетные данные клиента, при условии, что для учетной записи клиента не установлен флаг управления **Учетная запись не может быть делегирована**. Данное право пользователя определено в объекте групповой политики стандартного контроллера домена, а также в локальной политике безопасности рабочих станций и серверов.
- **Синхронизация данных службы каталогов (Synchronizedirectoryservicedata).** Определяет, какие пользователи и группы имеют полномочия синхронизировать данные, относящиеся к службе каталогов. Эта процедура также называется синхронизацией ActiveDirectory.
- **Создание журналов безопасности (Generatesecurityaudits).** Благодаря этому праву пользователь имеет возможность генерировать записи журнала безопасности.

– **Создание маркерного объекта (Create a tokenobject).** Имея такое право, пользователь может создавать маркеры безопасного доступа.

Создание постоянных объектов совместного использования (Create permanentsharedobjects). Это право позволяет пользователю создавать постоянные объекты совместного использования, такие, как экземпляры устройств.

– **Создание страничного файла (Create a pagefile).** Это право дает пользователю возможность создавать файл подкачки.

– **Увеличение приоритета диспетчирования (Increaseschedulingpriority).**

Это право дает пользователю возможность повысить приоритет выполнения процесса.

– **Управление аудитом и журналом безопасности (Manageauditingandsecuritylog).** Пользователь, имеющий такое право, может управлять аудитом файлов, каталогов и других объектов.

В большинстве случаев вы управляете пользовательскими правами, добавляя пользователей в группу, права которой уже определены. Но права можно присваивать как группе, так и каждой учетной записи отдельно.

Чтобы задать права, необходимо запустить диспетчер **Локальных параметров безопасности (LocalSecuritySettings)**, представленного на рис. 10.

Для этого необходимо открыть меню **Пуск (Start)**, выберите пункт **Выполнить...** и введите «secpol.msc». Последовательно выбрав разделы **Локальные политики (LocalPolicies)**, **Назначения прав пользователя (UserRightsAssignment)**, выводим полный список локальных политик безопасности со списком определенных для них параметров (Пользователей, Групп, Встроенных участников безопасности).

Двойным щелчком мыши на любой политике безопасности вызывается диалоговое окно **Свойств (Properties)**, пример которого представлен на рис. 8.

Для добавления Пользователей, Групп, Встроенных участников безопасности в список параметров безопасности используется кнопка **Добавить пользователя или группу (AddUsersorGroups)**, вызывающая стандартное диалоговое окно, представленное ранее на рисунке 6.

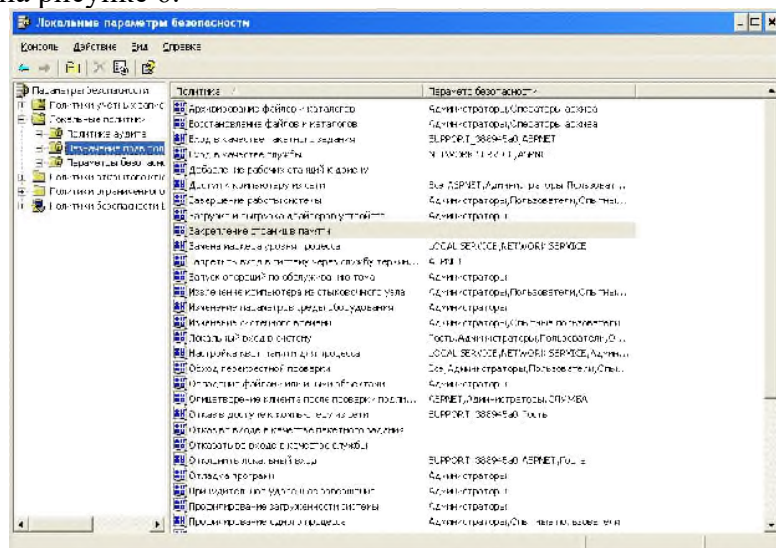


Рис. 7. Диалоговое окно Политика прав пользователей

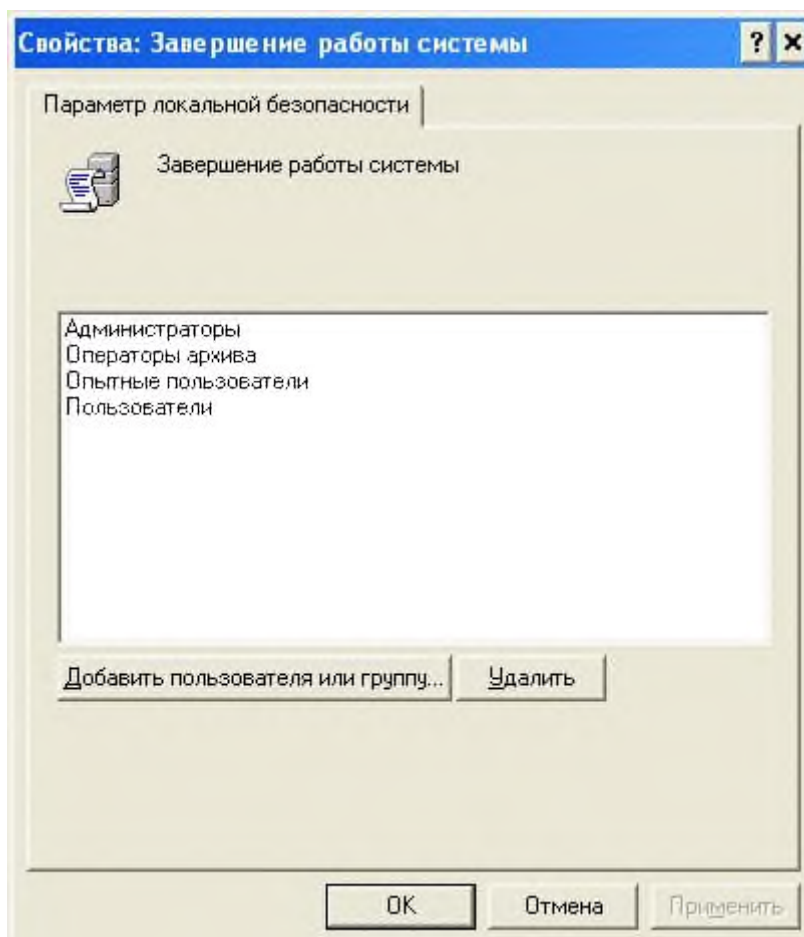


Рис. 8. Диалоговое окно Добавление пользователей и групп

Завершение работы

Закрывать все работающие программы. При завершении работы ММС сохранить настройки консоли.

Выйти из системы.

Практическое занятие «Настройка параметров рабочей среды пользователя»

Цель занятия: научиться настраивать параметры рабочей среды пользователя.

Содержание работы:

Упражнение 1

Получите навык работы с дисками. Освойте программы обслуживания дисков такие как: форматирование, дефрагментация и проверка диска на наличие ошибок, выполнив следующее:

1. возьмите дискету 3,5" емкостью 1,44 Мбайт и отформатируйте ее;
2. скопируйте на flash-носитель несколько рабочих объектов (папок, файлов, ярлыков, приложений) из вашей папки;
3. в произвольном порядке удалите с flash-носителя половину объектов из тех, которые были скопированы в п.2 данного упражнения. После удаления

просмотрите содержимое  . Находятся ли удаленные файлы в  ?

4. проверьте поверхность flash-носителя и информацию, содержащуюся на ней, программой *ScanDisk*. Получите итоговую информацию о сканировании;
5. выполните дефрагментацию flash-носителя;
6. получите информацию о flash-носителе ¹;

7. вставьте в дисковод дискету, получите информацию о ней, после чего все с нее удалите. Опять получите информацию о дискете.

Упражнение 2

Продолжите работу по закреплению навыка работы с операциями над файловой системой *Windows* и с поисковой системой *Windows*, освоите работу в программе–архиваторе *WinRAR*, выполнив следующее:

1. в вашей папке создайте папку с именем «**Архив**». После чего скопируйте в эту папку несколько файлов с расширениями **txt**, **doc**, **bmp**. Общее количество файлов должно быть 15;

2. работая в программе–архиваторе *WinRAR*, создайте архивный файл с названием «**Архив№1**», содержащий скопированные файлы с расширением **doc**. Перед упаковкой файлы выделите при помощи шаблона выделения. Посмотрите общий объем скопированных файлов до их архивации и после упаковки их в архив. На сколько изменился объем упакованных файлов?

3. работая в программе–архиваторе *WinRAR*, создайте архивный файл с названием «**Архив№2**», содержащий скопированные файлы с расширением **txt**. Перед упаковкой выделите данные объекты при помощи шаблона выделения, установите пароль, а затем упакуйте. Посмотрите общий объем скопированных файлов до их архивации и после упаковки их в архив. На сколько изменился объем упакованных файлов?

4. добавьте недостающие файлы **bmp** в созданный архив «**Архив№1**». Перед упаковкой файлы выделите при помощи шаблона выделения;

5. просмотрите содержимое некоторых файлов архива «**Архив№1**»;

6. работая в программе–архиваторе *WinRAR*, удалите из архива «**Архив№1**» несколько файлов;

7. создайте самораспаковывающийся архив для вашей папки;

8. завершите работу программы–архиватора *WinRAR*. При помощи программы *Проводник* найдите в файловой системе созданные вами архивы и откройте их для просмотра;

9. удалите найденные файлы с расширениями **txt**, **doc**, **bmp** в папке «**Архив**»;

10. работая в программе *Проводник*, извлеките файлы из вашего архива «**Архив№1**» во вновь созданную вложенную подпапку «**Extra**» в папке «**Архив**». После чего удалите извлеченные файлы из папки «**Extra**»;

11. работая в программе–архиваторе *WinRAR*, извлеките файлы из вашего архива «**Архив№1**» в папку «**Extra**»;

12. работая в программе *Проводник*, извлеките файлы без подтверждения из вашего архива «**Архив№2**». Что произойдет при попытке просмотра файлов, запакованных с использованием пароля? После чего удалите извлеченные файлы;

13. работая в программе–архиваторе *WinRAR*, извлеките файлы без подтверждения из вашего архива «**Архив№2**»;

14. найдите с помощью справочной системы архиватора *WinRAR* ответ на вопрос «**Как извлечь файлы?**». Ответ скопируйте и поместите в текстовый документ **Извлечение.txt**, предварительно созданный в папке «**Архив**»;

15. работая в программе *Проводник*, найдите на диске 15 файлов с расширением **bmp** и упакуйте их в многотомный архив с именем «**Архив№3**», так чтобы части тома могли поместиться на дискетах;

16. оставьте первый том созданного многотомного архива, а остальные удалите;

17. просмотрите содержимое первого тома архива «**Архив№3**». Сколько файлов находится в данном томе архива?

18. восстановите оставшиеся тома архива «Arhiv№3» и распакуйте полностью многотомный архив «Arhiv№3»;
19. выйдите из программы–архиватора *WinRAR*.

Упражнение 3

Научитесь видеть вирусы самостоятельно, выполнив следующее:

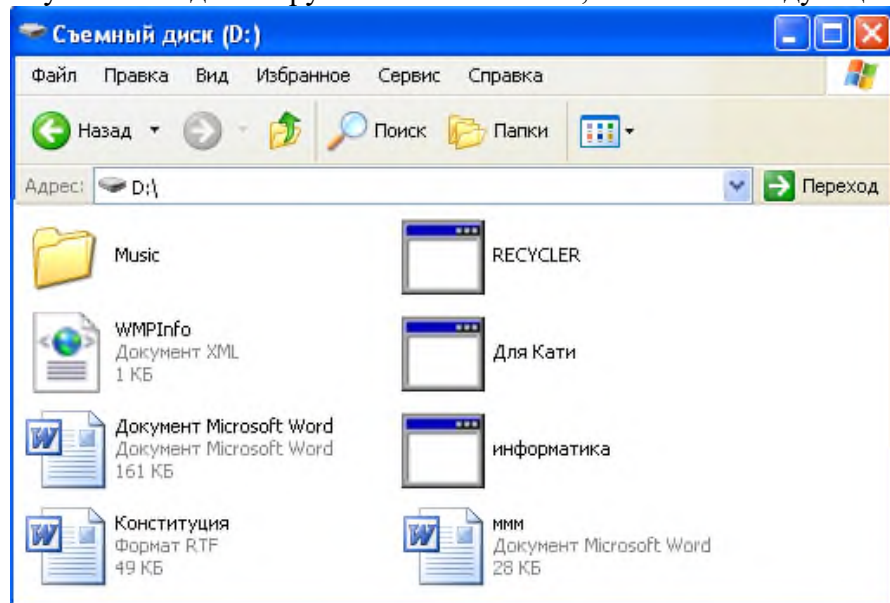


Рисунок 9. Пример окна «Съемный диск (D:)»

1. сколько вирусов изображено на рисунке 9. Почему вы так решили?
2. какие виды вирусов были найдены в ходе проверки flash–носителя антивирусной программой *NOD32* на рисунке 10. Какие действия они оказывают? Какие файлы заразили вирусы, какие расширения у данных файлов?
3. Ответьте на вопрос: «Есть ли шанс у студента, проверявшего свой flash–носитель на вирусы, продемонстрировать выполненную домашнюю работу по информатике?». Ответ свой поясните.

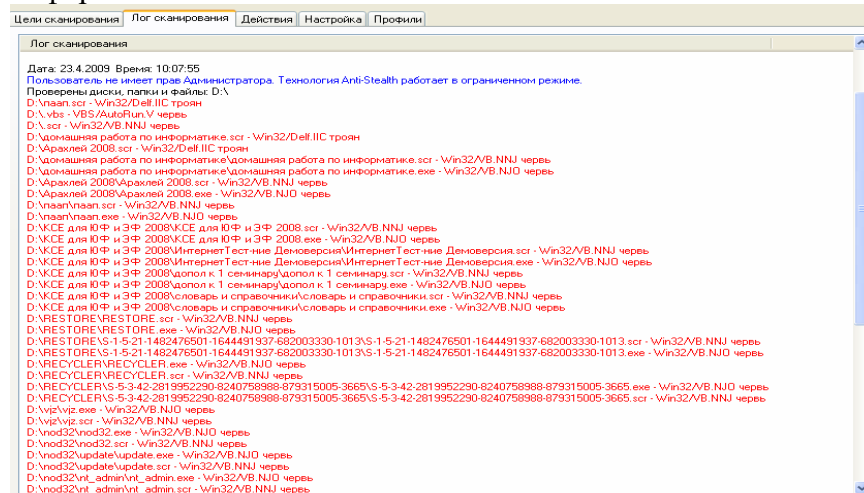


Рисунок 10. Пример части окна антивирусной программы *NOD32* в момент проверки flash–носителя на вирусы

Упражнение 4

Научитесь проверять свой flash–носитель (компьютер) на наличие вирусов, продолжите работу по обнаружению вирусов в ручную, выполнив следующее:

1. проверьте ваш flash–носитель на наличие вируса. В случае обнаружения вируса не применяйте никаких мер по его удалению (лечению). Откройте свой flash–носитель и скажите, видите вы или нет вирусы, если видите то, проговорите какие файлы были заражены. Какой вид вируса мог это сделать?

2. если в *Windows* нет настройки, чтобы отображались скрытые файлы и папки, то сделайте так, чтобы они отображались. Видите ли вы еще какие-нибудь вирусы, если видите, то какие?
3. если вирусы были обнаружены, то проверьте ваш flash-носитель снова, только на этот раз избавьте свой носитель от вирусов;
4. обновите вашу (домашнюю) версию антивирусной программы;
5. проверьте свой компьютер (домашний) на вирусы.

Практическое занятие «Работа с аппаратными и программными средствами защиты памяти»

Цель занятия: Практическое знакомство с управлением вводом/выводом в операционных системах Windows и кэширования операций ввода/вывода.

Теоретическая часть

Содержание работы:

Краткие теоретические сведения:

Необходимость обеспечить программам возможность осуществлять обмен данными с внешними устройствами и при этом не включать в каждую двоичную программу соответствующий двоичный код, осуществляющий собственно управление устройствами ввода/вывода, привела разработчиков к созданию системного программного обеспечения и, в частности, самих операционных систем.

Программирование задач управления вводом/выводом является наиболее сложным и трудоемким, требующим очень высокой квалификации. Поэтому код, позволяющий осуществлять операции ввода/вывода, стали оформлять в виде системных библиотечных процедур; потом его стали включать не в системы программирования, а в операционную систему с тем, чтобы в каждую отдельно взятую программу его не вставлять, а только позволить обращаться к такому коду. Системы программирования стали генерировать обращения к этому системному коду ввода/вывода и осуществлять только подготовку к собственно операциям ввода/вывода, то есть автоматизировать преобразование данных к соответствующему формату, понятному устройствам, избавляя прикладных программистов от этой сложной и трудоемкой работы. Другими словами, системы программирования вставляют в машинный код необходимые библиотечные подпрограммы ввода/вывода и обращения к тем системным программным модулям, которые, собственно, и управляют операциями обмена между оперативной памятью и внешними устройствами.

Таким образом, управление вводом/выводом — это одна из основных функций любой ОС. Одним из средств управления вводом/выводом, а также инструментом управления памятью является диспетчер задач Windows, он отображает приложения, процессы и службы, которые в текущий момент запущены на компьютере. С его помощью можно контролировать производительность компьютера или завершать работу приложений, которые не отвечают.

При наличии подключения к сети можно также просматривать состояние сети и параметры ее работы. Если к компьютеру подключились несколько пользователей, можно увидеть их имена, какие задачи они выполняют, а также отправить им сообщение.

Также управлять процессами можно и «вручную» при помощи командной строки. Команды Windows для работы с процессами:

- at - запуск программ в заданное время
- Schtasks - настраивает выполнение команд по расписанию

Start - запускает определенную программу или команду в отдельном окне.

- Taskkill - завершает процесс
- Tasklist - выводит информацию о работающих процессах
Для получения более подробной информации, можно использовать центр справки и поддержки или команду help (например: help at)
- command.com - запуск командной оболочки MS-DOS
- cmd.exe - запуск командной оболочки Windows

Ход работы:

Задание 1. Работа с Диспетчером задач Windows 7.

1. Запустите ранее установленную ОС Windows 7.
2. Запуск диспетчера задач можно осуществить двумя способами:
 - 1) Нажатием сочетания клавиш Ctrl+Alt+Del. При использовании данной команды не стоит пренебрегать последовательностью клавиш. Появится меню, в котором курсором следует выбрать пункт «Диспетчер задач».
 - 2) Переведите курсор на область с показаниями системной даты и времени и нажмите правый клик, будет выведено меню, в котором следует выбрать «Диспетчер задач».
3. Будет выведено окно как на рис. 21.

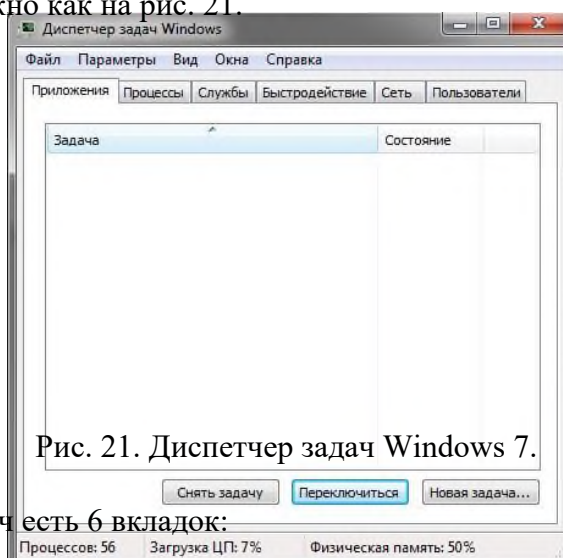


Рис. 21. Диспетчер задач Windows 7.

4. В диспетчере задач есть 6 вкладок:

- 1) Приложения
- 2) Процессы
- 3) Службы
- 4) Быстродействие
- 5) Сеть
- 6) Пользователи

Вкладка «Приложения» отображает список запущенных задач (программ) выполняющиеся в настоящий момент не в фоновом режиме, а также отображает их состояние. Также в данном окне можно снять задачу переключиться между задачами и запустить новую задачу при помощи соответствующих кнопок.

- о Вкладка «Процессы» отображает список запущенных процессов, имя пользователя

запустившего процесс, загрузку центрального процессора в процентном

соотношении, а также объем памяти используемого для выполнения процесса. Также присутствует возможность отображать процессы всех пользователей, либо принудительного завершения процесса. Процесс — выполнение пассивных инструкций компьютерной программы на процессоре ЭВМ.

- о Вкладка «Службы» показывает, какие службы запущены на компьютере. Службы — приложения, автоматически запускаемые системой при запуске ОС Windows и выполняющиеся вне зависимости от статуса пользователя.
- о Вкладка «Быстродействие» отображает в графическом режиме загрузку процессора, а также хронологию использования физической памяти компьютера. Очень эффективным инструментом наблюдения является «Монитор ресурсов». С его

помощью можно наглядно наблюдать за каждой из сторон «жизни» компьютера. Подробное изучение инструмента произвести самостоятельно, интуитивно.

- о Вкладка «Сеть» отображает подключенные сетевые адаптеры, а также сетевую активность.
- о Вкладка «Пользователи» отображает список подключенных пользователей.

5. После изучения диспетчера задач:

- о Потренируйтесь в завершении и повторном запуске процессов.
- о Разберитесь мониторинг загрузки и использование памяти.
- о Попробуйте запустить новые процессы при помощи диспетчера, для этого можно использовать команды: `cmd`, `msconfig`.

Задание 2. Командная строка Windows.

1. Для запуска командной строки в режиме Windows следует нажать:



(Пуск) > «Все программы» > «Стандартные» > «Командная строка»

2. Поработайте выполнением основных команд работы с процессами: запуская, отслеживая и завершая процессы.

Основные команды

4. Schtasks - выводит выполнение команд по расписанию
5. Start - запускает определенную программу или команду в отдельном окне.
Taskkill - завершает процесс
6. Tasklist - выводит информацию о работающих процессах

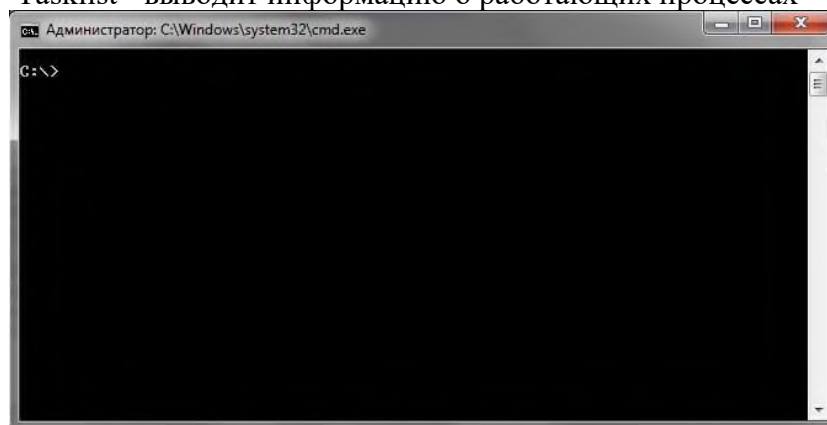


Рис. 22. Командная строка Windows 3. В появившемся окне (рис.22) наберите:

7. `cd/` - переход в корневой каталог;
8. `cd windows` – переход в каталог Windows. `dir` - просмотр содержимого каталога.

В данном каталоге мы можем работать с такими программами как «WordPad» и «Блокнот».

9. Запустим программу «Блокнот»: `C:\Windows > start notepad.exe`
 - 10.
 11. Отследим выполнение процесса: `C:\Windows > tasklist`
 12. Затем завершите выполнение процесса: `C:\Windows > taskkill /IM notepad.exe`
 13. Самостоятельно, интуитивно, найдите команду запуска программы WordPad. Необходимый файл запуска найдите в папке Windows.
 14. Выполнение задания включить в отчет по выполнению лабораторной работы.
- Задание 3.** Самостоятельное задание.

1. Отследите выполнение процесса `explorer.exe` при помощи диспетчера задач и командной строки.
2. Продемонстрируйте преподавателю завершение и повторный запуск процесса `explorer.exe` из:
 - Диспетчера задач;
 - Командной строки.

3. Выполнение задания включить в отчет по выполнению лабораторной работы.

Контрольные вопросы:

1. Дайте понятие процессу в операционной системе.
2. Дайте понятие службе в операционной системе.

Причислите основные команды работы с процессами при помощи командной строки.

Практическое занятие «Управление дисками и файловыми системами»

Цель занятия: изучить общие понятия о файловых системах и изучить методы управления файлами.

Содержание работы:

Совокупность каталогов и системных структур данных, отслеживающих размещение файлов на диске и свободное дисковое пространство, называется файловой системой. Основной структурной единицей любой файловой системы является файл и каталог.

Файл – минимальная структурированная именованная последовательность данных. Каталог (папка) является своеобразной объединяющей структурой для расположенных на диске файлов. Каталог может содержать в себе файлы и другие (вложенные) каталоги. Каталоги и файлы образуют на диске древовидную иерархическую структуру – дерево каталогов. Единственный каталог, не входящий ни в одну из директорий, называется корневым каталогом.

Магнитные диски являются устройствами произвольного доступа. В них каждая запись данных имеет свой уникальный адрес, обеспечивающий непосредственный доступ к ней, минуя все остальные записи. Для хранения данных служит диск (пакет из нескольких дисков), покрытый ферромагнитным слоем. Запись на магнитный диск и считывание данных с него осуществляется головками чтения/записи.

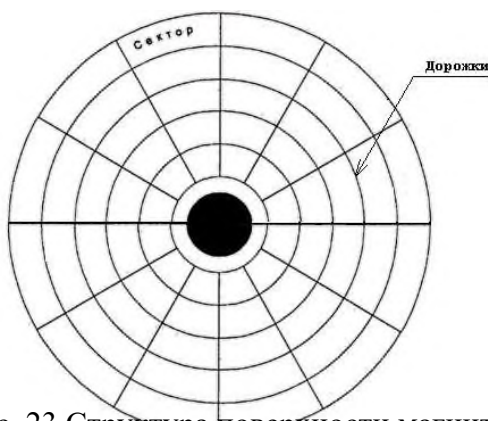


Рис. 23 Структура поверхности магнитного диска

Поверхность диска разбита на дорожки, представляющие собой окружности. Дорожки разделены на секторы. Размер сектора обычно составляет 512 байт.

В большинстве файловых систем пространство на диске выделяется кластерами, которые состоят из нескольких секторов. Кластер – минимальный размер места на диске, которое может быть выделено для хранения одного файла. Перед тем, как диск может быть использован для записи данных, он должен быть размечен — на его дорожки должны быть записаны заголовки секторов с правильными номерами дорожки и сектора, а также, если это необходимо, маркеры. Как правило, при этом же происходит тестирование поверхности диска для поиска дефектов магнитного слоя. Не следует путать эту операцию — физическое форматирование диска — с логическим форматированием, заключающемся в создании файловых систем. Современные жесткие диски обычно требуют физической разметки при их изготовлении.

Один физический жесткий диск может быть разделен на несколько разделов — логических дисков (томов). Каждый логический диск представляет собой как бы отдельное устройство. Следовательно, на нем может быть своя файловая система и свой корневой каталог.

В операционных системах MS-DOS и Windows каждое дисковое устройство обозначается латинской буквой. Для имени логического диска используются буквы от A до Z. Буквы A и B обозначают дисководы гибких магнитных дисков (FDD). Начиная с буквы C, именуется разделы жесткого диска (HDD), дисководы оптических дисков и виртуальные диски. Для обращения к файлу используется следующая спецификация:

устройство:\путь\имя файла.расширение Здесь путь – список каталогов, входящих друг в друга, в последнем из которых и

содержится указанный файл. Если путь не указан, следует что, файл находится в корневом каталоге данного диска. В MS-DOS имя файла состоит из 8 символов, точки и 3 символов расширения имени файла. Точка отделяет собственно имя от расширения. Имя файла может состоять из латинских букв, цифр 0 – 9, некоторых других символов, и не может содержать пробел. В Windows поддерживаются длинные имена файлов (от 1 до 255 символов), имя может содержать пробелы. При использовании файловых систем HPFS и NTFS имя файла может содержать несколько точек.

В именах файлов нельзя использовать символы “ * ” и “ ? ”, так как они используются в масках имен при поиске файлов.

Расширение имени необходимо для определения типа файла и связывания файла с определенной программой, с помощью которой он может быть открыт. Хотя имя файла может и не иметь расширения.

Различают следующие типы файлов:

- Текстовые файлы. Текстовые файлы могут содержать простой или размеченный текст, в кодировке ASCII, ANSI или UNICODE. Текст без разметки содержит только отображаемые символы и простейшие управляющие символы (возврат каретки и табуляции). Размеченный текст содержит бинарную и символьную разметку (межстрочный интервал, новая страница и т.п.), может содержать таблицы и рисунки;
- Графические файлы – файлы, содержащие точечные или векторные изображения;
- Файлы мультимедиа – различают файлы, содержащие оцифрованный звук (файлы аудио) и файлы видео (содержат изображение и звук);
- Исполняемые файлы – программы готовые к исполнению (файлы с расширением exe и com).
- Архивные файлы – файлы архивов rar, tar, zip, cab и т.п.
- Файлы библиотек – файлы с расширением DLL, OCX и LIB;
- Файлы данных – бинарные или текстовые файлы с различным расширением, используемые программами во время работы.

Информация о логической организации физического жесткого диска (числе логических дисков, их размере) расположена в главной загрузочной записи (MBR). MBR расположена в самом первом секторе жесткого диска и не входит в структуру файловой системы.

В операционных системах семейства UNIX разделение на логические диски отсутствует, а используется понятие корневого каталога файловой системы. Спецификация обращения к файлу выглядит следующим образом:

/путь/имя файла, тип

Современные операционные системы имеют возможность работать с несколькими файловыми системами одновременно. Прежде чем операционная система сможет использовать файловую систему, она должна выполнить над этой системой операцию, называемую монтированием.

В общем случае операция монтирования включает следующие шаги:

- Проверку типа монтируемой файловой системы; проверку целостности файловой системы;
- Считывание системных структур данных и инициализацию соответствующего модуля файлового менеджера (драйвера файловой системы). В некоторых случаях — модификацию файловой системы с тем, чтобы указать, что она уже смонтирована;
- Включение новой файловой системы в общее пространство имен. Многие пользователи MS DOS никогда не сталкивались с понятием монтирования. Дело в том, что эта система выполняет упрощенную процедуру монтирования при каждом обращении к файлу.

В данной работе продолжаем изучение работы с командной строкой Windows.

Задание 1. Проверка работы команд.

Потренироваться в выполнении нижеследующих команд. Работу проводить на установленной ранее ОС Windows 7.

1. Команда смены текущего диска A: - переход на диск C: - переход на диск C
2. Просмотр каталога
dir (путь)(имя_файла) (/p) (/w)

Если не введены путь и имя файла, то на экран выведется информация о содержимом каталога (имена файлов, их размер и дата последнего изменения).

Параметр /p задает вывод информации в поэкранном режиме, с задержкой до тех пор, пока пользователь не щелкнет по какой-либо клавише. Это удобно для больших

каталогов. Параметр /w задает вывод информации только об именах файлов в каталоге по пять имен в строке.

3. Переход в другой каталог `cd <имя каталога>`

4. Создание каталога `md <имя каталога>`

5. Удаление каталога `rd <имя каталога>`

6. Создание текстовых файлов `сору соп <имя_файла>`

После ввода этой команды нужно будет поочередно вводить строки файла. В конце каждой строки надо щелкать клавишей Enter. А после ввода последней - одновременно нажать Ctrl и Z, а затем Enter. Или клавишу F6, затем Enter.

7. Удаление файлов

`del (путь)имя_файла`

Путь прописывается только тогда, когда удаляемый файл находится в другом каталоге.

8. Переименование файлов `ren (путь)имя_файла1 имя_файла2`

Имя_файла1 - имя файла, который вы хотите переименовать.

Имя_файла2 - новое имя файла, которое будет ему присвоено после выполнения команды.

Путь прописывается только тогда, когда удаляемый файл находится в другом каталоге. 9.

Копирование файлов

`сору имя_файла (путь)имя_файла1`

Путь прописывается, если файл копируется в другой каталог.

Задание 2. Индивидуальная работа.

1. Получить у преподавателя индивидуальное задание.
2. Выполнить, результат внести в отчет о выполнении лабораторной работы.
3. Представить отчет преподавателю.

Контрольные вопросы:

1. Что такое «файл»?
2. Перечислите основные типы файлов.
3. Перечислите основные расширения файлов.

Практическое занятие «Настройка контроля доступа к файлам»

Цель занятия: настраивать контроль доступа к файлам.

Содержание работы:

Задание 1. Открыть Microsoft Virtual PC

Задание 2. Загрузить виртуальную машину Windows XP и создать новую учетную запись uir.

Задание 3. Загрузить виртуальную машину Windows XP с учетной записью uir.

Задание 4. Определение разрешений NTFS по умолчанию для только что созданной папки.

Запустить **Проводник**, создать папки **C:\Folder1** и **C:\Folder1\Folder2**. Просмотреть разрешения, установленные для созданных папок, щелкнув по вкладке **Безопасность** диалогового окна свойств папки. Обратить внимание на наследование разрешений папкой **Folder2** от родительской папки **Folder1**.

Если на экране не видна вкладка **Безопасность**, вам следует уточнить два вопроса:

1) Раздел вашего диска отформатирован как NTFS или как FAT? Только на разделах NTFS используются разрешения NTFS, и, таким образом, только на разделах NTFS видна вкладка **Безопасность**.

2) Используете вы простой общий доступ к файлам или нет? Щелкните кнопку **Отмена**, чтобы закрыть диалоговое окно свойств папки. В пункте меню **Сервис** выберите пункт **Свойства папки**. В диалоговом окне **Свойства папки** перейдите на вкладку **Вид**. В списке **Дополнительные параметры** снимите

флажок **Использовать простой общий доступ к файлам** (рекомендуется) и щелкните **ОК**.

Определить для какой группы установлены особые разрешения. Щелкнуть кнопку **Дополнительно**, выделить эту группу и просмотреть установленные разрешения.

Закрыть диалоговое окно свойств папки. Закрыть окно **Проводник** и завершить сеанс.

Задание 5. Создать новую учетную запись **uir-1**.

Задание 6. Войти в систему с учетной записью **uir-1**. Запустить **Проводник**, войти в папку **C:\Folder1**. Создать два текстовых документа, присвоив им имена **file 1** и **file 2**

Попытаться выполнить следующие операции с файлом **file1**: открыть файл; изменить файл; удалить файл. Какие действия вы смогли успешно совершить и почему?

Завершить сеанс работы и войти в систему, используя учетную запись **uir-2**. Запустить **Проводник**, войти в папку **C:\Folder1**. Попытаться выполнить следующие операции с файлом **file2**: открыть файл; изменить файл; удалить файл. Какие действия вы смогли успешно совершить и почему? В настоящее время ваша регистрационная запись — **uir-2**. Можете ли вы изменить разрешения, установленные для пользователя, пока вы подключены как **uir-2**? Почему? Завершить сеанс.

Задание 7. Установить разрешения NTFS для папки **C:\Folder1**. При этом необходимо соблюдать следующие правила:

1) все пользователи должны иметь возможность читать документы и файлы в папке **Folder1**;

2) все пользователи должны иметь возможность создавать документы в папке **Folder1**;

3) все пользователи должны иметь возможность изменять содержание, свойства и разрешения для создаваемых ими документов в папке **Folder1**;

4) пользователь **uir-2** несет ответственность за содержимое папки **Folder1** и должен иметь возможность изменять и удалять все файлы в папке **Folder1**.

Основываясь на полученной информации, определить, как следует изменить разрешения для соответствия этим четырем критериям?

Войти в систему, используя учетную запись **uir**. Открыть **Проводник**. Открыть папку **Folder1**. Щелкнуть правой кнопкой мыши значок папки **Folder1**, затем выбрать пункт меню **Свойства**. Перейти на вкладку **Безопасность** диалогового окна свойств папки. На вкладке **Безопасность** щелкнуть кнопку **Добавить**. Откроется диалоговое окно **Выбор: Пользователи или Группы**.

В текстовом поле **Введите имена выбираемых объектов** ввести **uir-2**, затем щелкнуть кнопку **Проверить имена**. В текстовом поле **Введите имена выбираемых объектов** должна появиться надпись **<имя компьютера>\uir-2**. Это свидетельствует, что Windows XP Professional обнаружила пользователя **uir-2** на компьютере **<имя компьютера>** и что это действительная учетная запись пользователя. Щелкнуть **ОК**, чтобы закрыть диалоговое окно **Выбор: Пользователи или Группы**. Теперь пользователь **uir-2** включен в список **Группы или пользователи** диалогового окна свойств папки **Folder1**. Какие разрешения установлены для пользователя **uir-2**?

Щелкнуть кнопку **Дополнительно**. Откроется диалоговое окно **Дополнительные параметры безопасности** для **Folder1**, и вы увидите, что пользователь **uir-2** включен в список **Элементы разрешений**. Убедиться, что строка **uir-2** выделена, и щелкнуть кнопку **Изменить**. Откроется диалоговое окно **Элемент разрешения для Folder1**, и вы увидите в текстовом поле **Имя** учетную запись пользователя **uir-2**.

В колонке **Разрешить** щелкнуть **Полный доступ**. Теперь в колонке **Разрешить** установлены все флажки. Щелкнуть **ОК**, чтобы закрыть диалоговое окно **Элемент разрешения для Folder1** и щелкнуть **ОК**, чтобы закрыть диалоговое окно **Дополнительные параметры безопасности для Folder 1**. Щелкнуть **ОК**, чтобы закрыть диалоговое окно свойств папки **Folder 1**.

Закрывать **Проводник** и завершить сеанс Windows XP Professional.

Войти в систему, используя учетную запись **uir-2**. Запустить **Проводник**, войти в папку **C:\Folder1**. Попытаться выполнить следующие операции с файлом **file2**: изменить файл; удалить файл. Какие действия вы смогли успешно совершить и почему? Завершить сеанс Windows XP Professional.

Задание 8. Проверить, как разрешения NTFS наследуются в иерархии папок.

Войти в систему, используя учетную запись **uir-1**. Запустить **Проводник**, войти в папку **C:\Folder1\Folder2**. Создать текстовый файл с именем **Ше3** в папке. Завершить сеанс Windows XP Professional.

Войти в систему, используя учетную запись **uir-2**. Запустить **Проводник**, войти в папку **C:\Folder1\Folder2**. Попытаться выполнить следующие операции с файлом **Ше3**: открыть файл; изменить файл; удалить файл. Какие действия вы смогли совершить и почему? Завершить сеанс Windows XP Professional.

Задание 9. Изучить результаты смены владельца файла.

Войти в систему, используя учетную запись **uir**. В папке **C:\Folder1** создать текстовый файл **file4**.

Щелкнуть правой кнопкой мыши значок документа **file4**, затем выбрать пункт меню **Свойства**. Откроется диалоговое окно **Свойства: file4** с активной вкладкой **Общие**. Перейти на вкладку **Безопасность** для просмотра разрешений, установленных для файла **file4**. Щелкнуть кнопку **Дополнительно**. Откроется диалоговое окно **Дополнительные параметры безопасности для file4** с активной вкладкой **Разрешения**. Перейти на вкладку **Владелец**. Кто является текущим владельцем файла **file4**?

Установка разрешения, позволяющего пользователю сменить владельца.

В диалоговом окне **Дополнительные параметры безопасности для file4** перейти на вкладку **Разрешения**. Щелкнуть кнопку **Добавить**. Откроется диалоговое окно **Выбор: Пользователи или Группы**. Убедиться, что в текстовом поле **Размещение**, которое расположено вверху диалогового окна, выбрано имя вашего компьютера. В текстовом поле **Введите имена выбираемых объектов** ввести **uir-3**, затем щелкните кнопку **Проверить имена**. Щелкнуть **ОК**.

Станет активным диалоговое окно **Элемент разрешения для file4**. Обратить внимание на то, что все элементы разрешений для пользователя **uir-3** не отмечены. В колонке **Разрешения** установить флажок **Разрешить** для разрешения **Сменить владельца**. Щелкнуть **ОК**. Щелкнуть **ОК** для того, чтобы вернуться к диалоговому окну свойств файла **file4**. Щелкнуть **ОК** для сохранения изменений и закрыть диалоговое окно свойств файла **file4**. Закрывать **Проводник** и выйти из системы. Смена владельца файла.

Войти в систему, используя учетную запись **uir-3**. Запустить **Проводник**, войти в папку **C:\Folder1**. Щелкнуть правой кнопкой мыши значок файла **file4** и выбрать пункт меню **Свойства**. Перейти на вкладку **Безопасность** для просмотра разрешений для файла. Щелкнуть **Дополнительно** и перейти на вкладку **Владелец**. В колонке **Изменить владельца** на выбрать **uir-3**, затем щелкнуть кнопку **Применить**. Кто теперь является владельцем файла **Ше4**?

Щелкнуть **ОК**, чтобы закрыть диалоговое окно **Дополнительные параметры безопасности для file4**.

Проверка разрешений для файла в качестве владельца.

Щелкнуть кнопку **Дополнительно** и снять флажок **Наследовать от родительского объекта применимые к дочерним объектам разрешения**, добавляя их к явно заданным в этом окне. Установить разрешение **Полный доступ** к текстовому документу **file4** и нажать кнопку **Применить**. Щелкнуть **ОК**, чтобы закрыть диалоговое окно **Дополнительные параметры безопасности для file4**. Щелкнуть **ОК**, чтобы закрыть диалоговое окно свойств файла **file4**.

Задание 10. Изучить изменение разрешений и прав владельца при копировании и перемещении папок.

Создание папки при подключении с учетной записью пользователя.

Пока вы зарегистрированы в системе под учетной записью **uir-3** создать папку с именем **Temp1** в корневой папке диска **C:**. Какие разрешения установлены для этой папки? Кто является владельцем папки?

Создание папок при подключении с учетной записью члена группы **Администраторы**.

Подключитесь с учетной записью **uir** и создайте папки **Temp2** и **Temp 3** в корневой папке диска **C:**.

Каковы разрешения для папок, которые вы только что создали? Кто является владельцем папок **Temp 2** и **Temp 3**?

Установить разрешения для папок **Temp 2** и **Temp 3**.

Снять флажок **Наследовать от родительского объекта применимые к дочерним объектам разрешения**, добавляя их к явно заданным в этом окне. В открывшемся диалоговом окне щелкните **Удалить»** для удаления всех разрешений, кроме указанных ниже.

Папка **Temp 2: Администраторы — Полный доступ; Пользователи — Чтение и выполнение**.

Папка **Temp 3: Администраторы — Полный доступ; Операторы архива — Чтение и выполнение; Пользователи — Полный доступ**.

Копирование папки в другую папку на одном и том же томе NTFS.

Пока вы находитесь в системе под учетной записью **uir**, скопировать папку **C:\Temp2** в папку **C:\Temp1**. Для этого выделить значок папки **C:\Temp2** и, удерживая нажатой клавишу **CTRL**, перетащить мышью **C:\Temp2** в **C:\Temp1**.

Выделив **C:\Temp1\Temp2**, просмотреть разрешения и права владельца, затем сравнить разрешения и права владельца с папкой **C:\Temp2**.

Перемещение папки на одном и том же томе.

Войти в систему с учетной записью **uir-3**. В **Проводнике** выделить значок папки **C:\Temp3**, затем переместить ее в папку **C:\Temp1**. Что произошло с разрешениями и владельцем для папки **C:\Temp1\Temp3**?

Задание 11. Самостоятельно определить? как предотвратить удаление пользователями, имеющими разрешение **Полный доступ** к папке, файла в этой папке, для которого установлен запрет на разрешение **Полный доступ**?

Контрольные вопросы

1. Что такое эффективные разрешения пользователя для ресурса?
2. Какие объекты по умолчанию наследуют разрешения, установленные для родительской папки?
3. Чем отличается разрешение «Удаление» от разрешения «Удаление подпапок и файлов»?
4. Какое разрешение NTFS для файлов следует установить для файла, если вы позволяете пользователям удалять файл, но не позволяете становиться владельцами файла?
5. Если вы хотите, чтобы пользователь или группа не имела доступ к определенной папке или файлу, следует ли запретить разрешения для этой папки или файла?

Практическое занятие «Изучение основных элементов и понятий ОС Windows»

Цель занятия: изучить основные элементы операционной системы Windows

Содержание работы:

Windows имеет модульную структуру, в которой код операционной системы и драйверы

выполняются в привилегированном режиме процессора (режиме ядра), обеспечивающем полный доступ ко всей аппаратной части компьютера, а пользовательские приложения выполняются в непривилегированном режиме процессора – пользовательском режиме без прямого доступа к оборудованию компьютера. В режиме ядра работают следующие компоненты.

1. **Уровень абстрагирования от оборудования (*Hardware Abstraction Layer, HAL*).** Его задачей является отделение операционной системы от особенностей конкретных реализаций в аппаратном обеспечении компьютера, т. е. от различий в материнских платах, в модификациях процессоров, в наборах микросхем и др. Благодаря этому уровню управление подсистемами прерываний, прямого доступа к памяти, системными шинами и таймерами для ядра операционной системы является одинаковым. Уровень *HAL* реализован в системном файле *Hal.dll*.

2. **Ядро операционной системы.** Ядро содержит наиболее часто вызываемые низкоуровневые функции операционной системы: планирование и распределение ресурсов между процессами, их переключение и синхронизацию. В обязанности ядра входит также управление прерываниями и обработка ошибочных ситуаций при функционировании операционной системы. Код ядра *Windows* не разделяется на потоки, а находится только в оперативной памяти и не может быть выгружен на диск. Код ядра *Windows XP* находится в системном файле *Ntoskrnl.exe*.

3. **Драйверы устройств.** Драйверы представляют собой подпрограммы, транслирующие вызовы, поступившие от пользовательских программ в запросы обработки данных для конкретных устройств. Значительное число драйверов входит в состав *Windows* (они располагаются в подкаталоге *Isystem32\drivers* системного каталога и имеют тип файла *.sys, например, драйвер дисковой подсистемы находится в файле *disk.sys*), а для нестандартных периферийных устройств драйверы находятся в комплектах поставки.

4. **Исполняющая подсистема (*NT Executive*).** Модуль *NT Executive* состоит из микроядра и подсистем диспетчеризации управления программами с доступом к виртуальной памяти, окнам и графической подсистеме. Виртуальная память предоставляет пользовательским программам виртуальные адреса адресного пространства процессов и соответствующие физические страницы оперативной памяти компьютера. Графическая подсистема предназначена для создания оконного интерфейса, рисования элементов управления, расположенных в окнах. К исполняющей подсистеме относятся системные файлы *Ntkrnlpa.exe*, *Kernel32.dll*, *Advapi32.dll*, *User32.dll*, *Gdi32.dll*.

Операционная система *Windows* в значительной мере использует возможности процессоров, совместимых с семейством *Intel x86*. В их аппаратной архитектуре предусматривается четыре уровня привилегий выполнения кода программ от 0-го наивысшего привилегированного до 4-го пользовательского режима с ограниченным набором команд процессора. Программы режима ядра операционной системы *Windows* функционируют в нулевом, защищенном и привилегированном режиме, а остальные пользовательские программы работают в менее привилегированных режимах, находясь под контролем программ режима ядра.

Недоступные в пользовательском режиме операции и приложения обращаются к системным вызовам ядра операционной системы *Win32 API*. В состав *API* входит более 250 функций, обращение к которым осуществляется при помощи системных вызовов, основанных на подпрограммах ядра операционной системы. Все вызовы *Win32 API* обслуживаются как системными службами *NT*, так и модулем *NT Executive* – исполняющей системы *Windows*. Модуль *NT Executive* представляет собой несколько программных потоков, которые выполняются в режиме ядра. Код практически всех подсистем этого модуля находится в файле *ntoskrnl.exe* (кроме подсистемы *Win32*, код которой расположен в файле *win32k.sys*) и уровне абстрагирования от оборудования *HAL*, который содержится в файле *hal.dll*. В модуле *NT Executive* сосредоточены все самые важные части операционной системы.

Микроядро отвечает за выделение памяти для приложений и распределение

процессорного времени, т. е. за реализацию многозадачности. Для этого в состав микроядра входит *планировщик потоков (threads scheduler)*, который назначает каждому из потоков один из 32 уровней приоритета. Уровень 0 зарезервирован для системы. Уровни от 1-го до 15-го назначаются исполняемым программам, а уровни от 16-го до 31-го могут назначаться только администраторами. Планировщик делит все процессорное время на кванты фиксированного размера. При этом каждый программный поток выполняется только в течение отведенного ему времени, и если по окончании кванта он не освобождает процессор, планировщик в принудительном порядке приостанавливает этот поток и меняет программное окружение процесса, настраивая его на выполнение другого потока, обладающего тем же приоритетом. Микроядро также осуществляет всю работу, связанную с обработкой программных и аппаратных прерываний.

5. Диспетчеризация управления программами. Модуль состоит из следующего набора системных программ:

- *Диспетчер ввода-вывода* – интегрирует добавляемые в систему драйверы устройств в операционную систему *Windows*;

- *Диспетчер объектов* – служит для управления всеми разделяемыми ресурсами компьютера. В момент обращения приложения к какому-либо ресурсу диспетчер объектов сопоставляет с этим ресурсом объект (например, окно) и отдает приложению дескриптор (№ окна) этого объекта. Используя дескриптор, приложение взаимодействует с объектом, совершая в его отношении различные операции. Монитор системы безопасности следит при этом за тем, чтобы с объектом выполнялись только разрешенные действия;

- *Диспетчер процессов* – предоставляет интерфейс, при помощи которого другие компоненты *Windows NT Executive*, а также приложения пользовательского режима могут манипулировать процессами и потоками. Во время работы диспетчер процессов сопоставляет с каждым процессом и потоком идентификатор процесса (*PID – Process Identifier*) и идентификатор потока (*TID – Thread Identifier*) соответственно, а также таблицу адресов и таблицу дескрипторов;

- *Диспетчер виртуальной памяти* – служит для управления организации подсистемы памяти, позволяет создавать таблицы адресов для процессов и следит за корректностью использования адресного пространства приложениями. Кроме того, обеспечивает возможность загрузки в оперативную память исполняемых файлов и файлов динамических библиотек. Диспетчер виртуальной памяти представляет физическую память для пользовательских приложений – каждому процессу выделяются 4 Гб виртуального адресного пространства, из которых младшие 2 Гб используются процессом, а старшие 2 Гб (общие для всех процессов) отводятся на нужды системы. Каждый процесс работает в своем изолированном адресном пространстве и «не знает» о других процессах. Процессы обмениваются данными через разделяемую память, которая может быть спроецирована на виртуальное адресное пространство нескольких процессов. Главная задача диспетчера виртуальной памяти – организация логической памяти, размер которой больше размера физической, установленной на компьютере. Это достигается благодаря тому, что страницы памяти, к которым долго не было обращений, и которые не имеют атрибута неперемещаемых, сохраняются диспетчером в файле *pagefile.sys* на жестком диске и удаляются из оперативной памяти, освобождая ее для других приложений. В момент, когда происходит обращение к данным, находящимся в перемещенной на винчестер странице, диспетчер виртуальной памяти копирует страницу обратно в оперативную память, затем обеспечивает доступ к ней. Этот механизм обеспечивает выделение дополнительной памяти программам, которые нуждаются в ней, и при этом следит за тем, чтобы все работающие в системе программы обладали достаточным объемом физической памяти для того, чтобы продолжать функционирование;

- *Диспетчер кэша* – используется для кэшированного чтения и записи и позволяет существенно ускорить работу жестких дисков и других устройств. При этом наиболее востребованные файлы дублируются диспетчером кэша в оперативной памяти компьютера, и обращение к ним обслуживается с использованием этой копии, а не оригинала,

расположенного на сравнительно медленном долговременном носителе. Кэш в *Windows* является единым для всех логических дисков, вне зависимости от используемой файловой системы. Кроме того, он является динамическим, а это значит, что диспетчер управляет его размерами в зависимости от доступного объема свободной физической памяти в каждый конкретный момент;

□ *Диспетчеры окон и графики* – выполняют все функции, связанные с пересылкой системных сообщений и отображением информации на экране.

Процесс функционирования *Windows* условно подразделяется на три фазы: процесс начальной загрузки, штатный режим работы и завершение работы. Для загрузки *Windows* используется следующий минимальный набор файлов:

- файлы, располагающиеся в корневом каталоге загрузочного диска: *Ntldr*, *Boot.ini*, *Bootsect.dos* (файл необходим только при использовании мультизагрузки), *Ntdetect.com*;

- файлы, располагающиеся в системном подкаталоге */system32*: *Ntoskrnl.exe*, *Hal.dll*, разделы реестра *SYSTEM*;

- файлы, располагающиеся в системном подкаталоге */system32/drivers*: (необходимые драйверы устройств).

Процесс загрузки компьютера начинается с процедуры начального тестирования оборудования (*POST – Power-OnSelfTest*). Код, выполняющий *POST*, зашит в базовой системе ввода-вывода (*BIOS*) каждого компьютера, при включении питания ему передается управление. Если в процессе тестирования обнаруживаются какие-либо ошибки, то *BIOS* генерирует коды ошибок (*POSTcodes*), которые отличаются для *BIOS* разных производителей, и звуковые коды. Если процедура *POST* завершается успешно, то *BIOS* передает управление главной загрузочной записи (*MBR – MasterBootRecord*) и первая «аппаратная» стадия загрузки компьютера, когда процесс зависит только от аппаратуры компьютера, завершается.

Далее загрузочная запись, оперируя данными о разбиении жесткого диска на логические тома, передает управление исполняемому коду, загрузчику *Ntldr*, расположенному в загрузочном секторе. Загрузчик переходит в защищенный режим и производит необходимые для успешного функционирования манипуляции с памятью, кроме этого, *Ntldr* имеет модули, позволяющие работать с файловой системой и некоторыми другими базовыми ресурсами системы. Все другие действия выполняются с помощью вызова прерываний *BIOS*.

Если в файле *boot.ini* зарегистрировано более одной операционной системы, то после первичной инициализации загрузчик предоставляет пользователю возможность выбора путем вывода *Ntldr* приглашения о выборе операционной системы. Если выбрана операционная система *WindowsXP*, загрузчик запускает файл *Ntdetect.com*. Этот компонент считывает из *CMOS*-памяти системную дату и время, после чего производит поиск и распознавание аппаратных средств, подключенных в данный момент к компьютеру. Завершив работу, *Ntdetect* возвращает управление и собранную им информацию обратно в *Ntldr*. Далее загружается и инициализируется ядро операционной системы *Ntoskrnl.exe* и уровень абстрагирования от оборудования *Hal.dll*. При инициализации ядро производит ряд действий в определенной последовательности:

- инициализация диспетчера памяти;
- инициализация диспетчера объектов;
- установка системы безопасности;
- настройка драйвера файловой системы;
- загрузка и инициализация диспетчера ввода-вывода;
- загрузка системных сервисов, которые реализуют взаимодействие с пользователем.

В состав системных сервисов входят следующие модули:

- *Smss.exe* (диспетчер сеансов) – модуль управляет другими сервисами и службами *Windows*; запускает: *Win32 (Csrss)* и некоторые системные утилиты, выполняемые на этапе загрузки; реализует графический пользовательский интерфейс и запуск

процессов *Csrss.exe* и *WinLogon.exe*;

- *Csrss.exe* – модуль предназначен для организации взаимодействия между компьютером и пользователем;

- *Lsass.exe* – служба, запускаемая *WinLogon.exe* и отвечающая за безопасность системы (предоставляет возможность пользователю зарегистрироваться в системе).

После загрузки операционной системы пользователь должен пройти процедуру *аутентификации* – ввести собственное регистрационное имя (*логин*) и пароль. Процедура подключения к системе позволяет определить, обладает ли пользователь правом входа и работы с системой. Эту процедуру выполняет служба *WinLogon*. При этом в системе происходят следующие события:

- *процесс WinLogon* отображает на экране фон рабочего стола и приглашение к вводу пользователем логина и пароля. Введенные данные передаются подсистеме безопасности;

- подсистема безопасности обращается к базе данных *SAM (Security Accounts Manager)* и проверяет, обладает ли пользователь полномочиями работы с системой.

Если пользователь является авторизованным пользователем системы, то подсистема безопасности формирует для него *идентификатор доступа*, который вместе с управлением передает обратно процессу *WinLogon*. Процесс *WinLogon* посредством обращения к подсистеме *Win32* создает новый процесс для пользователя и прикрепляет ему идентификатор доступа. Каждый процесс, в дальнейшем создаваемый пользователем, отмечается принадлежащим ему идентификатором доступа, поэтому доступ пользователя к ресурсам системы контролируется и отслеживается. Благодаря обязательной процедуре подключения к системе упрощается реализация механизмов: *аудит системы* и *квоты на использование ресурсов*. Пользовательский идентификатор доступа содержит идентификатор пользователя, а также идентификаторы всех групп, к которым принадлежит данный пользователь.

Если операционная система не загружается корректно, то при нажатии в процессе загрузки *WindowsXP* клавиши *F8* происходит переход в расширенное меню запуска, содержащее пункты:

- *Безопасный режим* – загрузка *Windows* с минимальным требуемым количеством системных файлов и драйверов устройств;

- *Безопасный режим с загрузкой сетевых драйверов* – загрузка *Windows* с минимальным требуемым количеством системных файлов и драйверов устройств с поддержкой подключения к сети;

- *Безопасный режим с поддержкой командной строки* – загрузка *Windows* с минимальным требуемым количеством системных файлов и драйверов устройств с загрузкой режима командной строки;

- *Включить протоколирование загрузки* – режим позволяет записать этапы загрузки *Windows* в файл *Ntbtlog.txt*;

- *Включить режим VGA* – режим, загружает драйвер стандартного монитора *VGA* с разрешением 640 на 480 точек на дюйм и 16 цветами;

- *Загрузка последней удачной конфигурации* – режим, восстанавливает последнюю неиспорченную копию реестра *Windows XP*.

Контрольные вопросы:

1. Каковы минимальные требования к аппаратной платформе, которые рекомендует Microsoft для установки *Windows*?
2. В чем преимущества файловой системы *NTFS*?
3. На какие фазы условно можно разделить работу ОС *Windows*?
4. Какие цели достигаются на уровне абстрагирования?
5. Что такое ядро системы?
6. Что такое драйвера устройств?
7. Что такое исполняющая система?
8. Что такое диспетчер ввода-вывода?
9. Что такое диспетчер объектов?

10. Что такое диспетчер процессов?
11. Что такое диспетчер кэша?
12. Опишите ряд действий выполняемых ядром при инициализации?

Практическое занятие «Работа с окнами: управление, переход»

Цель занятия: ознакомиться со структурой стандартного окна ОС Windows, приемами работы с одним и несколькими окнами.

Содержание работы:

Изучение приемов работы со стандартным окном Windows провести на примере окна папки диска D:. Для этого необходимо:

- 1) открыть папку диска D.;
- 2) представить схематический рисунок окна и описать назначение основных его элементов;
- 3) переместить окно на рабочем столе и сменить его размеры;
- 4) свернуть и обновить окно, а потом закрыть его; объяснить, в чем заключается разница между сворачиванием и закрытием окна;
- 5) привести примеры, когда целесообразно раскрывать окно на полный экран, а когда – только на его часть;
- 6) добавить, а потом снять панель инструментов;
- 7) определить команды меню, которые дублируются кнопками панели инструментов;
- 8) перечислить виды пунктов меню и привести примеры каждого вида;
- 9) объяснить, зачем нужны разные способы представления содержания папки;
- 10) вывести содержание диска D: таким образом, чтобы сначала выводились объекты, созданные недавно, а потом – более старые объекты;
- 11) перейти с окна диска D: в окно диска A:

Для изучения приемов работы с несколькими окнами необходимо:

- 1) установить в окне папки диска D: свойства, чтобы каждая папка открывалась в отдельном окне, и открыть сразу окна нескольких папок;
- 2) попробовать разные способы переключения из одного окна на другое и сравнить их;
- 3) выполнить упорядочение окон на рабочем столе разными способами и сравнить их.

Выполнение задания

1. Открываем папку диска D: двойным щелчком левой кнопкой мышки (или же выделить курсором мыши, а затем нажать клавишу Enter) или правой кнопкой мышки выводим на экран контекстное меню и с помощью меню «Открыть» открываем папку.

2. Схематический рисунок окна

Основные элементы:

Заголовок – отображается название папки, файла или программы, которые открыты в окне. Строка заголовка служит для перемещения окна по экрану.

Рамка – определяет границы окна, стороны и углы которого можно использовать для изменения его размеров.

Оконное меню – список тех операций, которые можно проделать с окном (переместить, восстановить, свернуть, развернуть, закрыть и т.д.).

Три кнопки: кнопка с горизонтальной чертой (свернуть), кнопка с квадратиком (полноэкранный режим) или двумя квадратиками (компактный вид), кнопка с крестиком (закрыть).

Стандартное меню – список тех операций, которые можно проделать с папками, находящимися в окне (копирование, перемещение, удаление, создание, печать, переименовать и т.д.).

Панель инструментов – содержит различные кнопки, которые используются в процессе

работы для навигации, обновления содержимого окна, возвращения на одно действие назад и т.д. – в зависимости оттого, что открыто в окне.

Адресная строка – путь к папке или документу, открытому в окне.

Рабочая область – центр окна. Эта область является самым крупным и важным элементом окна, в котором появляются сообщения, отображается содержимое файлов и папок и т.д.

Строка состояния (некоторые программы не отображают строку состояния, в этом случае команда «Строка состояния» в меню «Вид» отсутствует) – в ней обычно отображается информация о размере объекта, количестве знаков в тексте, о том, какая операция в данный момент совершается программой и т.д.

Информационная область – информация о том, какие действия можно совершать с объектами в окне, размер, дата создания и изменения и т.д.

Полоса прокрутки – бегунки, с помощью которых можно пролистать содержимое окна вверх, вниз и в стороны.

3. Перемещать окна достаточно просто. Для начала «зацепляем» заголовок окна мышкой (ставим над ним курсор мыши, нажимаем и удерживаем левую кнопку мыши). Затем, перемещая курсор мыши, перетаскиваем окно в нужную область экрана и отпускаем заголовок окна (для этого отпускаем левую кнопку мыши).

Изменение размеров окна:

Во-первых, можно переводить окно из полноэкранного размера в обычный (необходимо воспользоваться кнопкой с квадратиком/квадратиками в правом верхнем углу экрана).

Некоторые окна (например, окна папок) можно развернуть на весь экран. Для этого щелкаем мышью по кнопке со значком квадратика, при этом удерживая кнопку Ctrl на клавиатуре.

Во-вторых, можно изменять размер окна более точно и в трех направлениях – по горизонтали, по вертикали и одновременно по горизонтали и вертикали. Для этого воспользуемся рамкой окна. Зацепляем ее курсором мыши (при этом курсор превратиться в двустороннюю стрелку) и перетаскиваем в нужную сторону. Зацепив нижнюю рамку, можно изменить высоту окна, боковую – его ширину. Если же изменять размер окна по высоте и ширине одновременно, зацепляем угол рамки и перетаскиваем его в нужную сторону.

4. Иногда возникает необходимость убрать окно с экрана, при этом, не останавливая работы в нем. В Windows предусмотрена возможность сворачивать и разворачивать окна на панель задач. Чтобы свернуть окно, просто щелкаем по кнопке с изображением горизонтальной черты в правом верхнем углу окна. Чтобы снова восстановить окно на экране (то есть развернуть его), щелкаем по соответствующей окну кнопке на панели задач. Также можно произвести эти операции с помощью «**Оконного меню**», а также с помощью комбинациями «**горячих клавиш**» клавиатуры: WIN+D. Для отмены операции сворачивания всех окон используем другую комбинацию – Shift+WIN+M.

Для окончательного закрытия окна и остановки работы программы, запущенной в нем, просто щелкаем по кнопке-крестику, расположенной в правом верхнем углу окна или же воспользуемся комбинацией «**горячих клавиш**» Alt+F4, или же с помощью «**Оконного меню**»/ «**Закрыть**».

5. Когда мы работаем с одним окном целесообразно раскрывать окно на полный экран, а когда работаем с несколькими окнами – только на его часть для удобства работы с окнами одновременно (например, для перемещения папок или файлов из одного окна в другое и т.д.).

6. Для того чтобы добавить или удалить кнопки на панели инструментов в окне папки, открываем папку «**Мои документы**». Дважды щелкаем папку, которую требуется настроить.

В меню «**Вид**» выбираем команду «**Панели инструментов**», а затем — команду «**Настройка**». Чтобы добавить кнопку, выбираем ее в списке «**Имеющиеся кнопки**» и

нажимаем кнопку **«Добавить»**.

Для того чтобы удалить кнопку, выбираем ее в списке **«Панель инструментов»** и нажимаем кнопку **«Удалить»**.

Для того чтобы изменить расположение кнопок на панели инструментов, выбираем нужную кнопку в списке **«Панель инструментов»** и перемещаем ее, используя кнопку **«Вверх»** или **«Вниз»**.

Можно менять местоположение надписей на кнопках и размеры значков панели инструментов, изменяя значения параметров **«Текст кнопки»** или **«Размер значка»**. Для того чтобы восстановить стандартные кнопки на панели инструментов, нажимаем кнопку **«Сброс»**. При этом **«Текст кнопки»** не восстанавливается.

Для того чтобы настроить панель инструментов, можно также щелкнуть ее правой кнопкой мыши и выбрать команду **«Настройка»**.

7. Панель инструментов.

В строке под главным меню находится панель инструментов. В ней содержатся кнопки, например,

Кнопка «Назад»	-позволяет быстро вернуться в предыдущую просмотренную папку
Кнопка «Вперед»	-поможет после этого вернуться назад
Кнопка «Переход»	-отправит нас в папку более высокого уровня
Кнопка «Поиск»	-откроет нам в левой части проводника «Панель Поиска» , заменившую на время панель папок, для поиска информации в компьютере
Кнопка «Папки»	-позволяет вернуться в основной режим «Проводника» - папки в левом окне, файлы – в правом.
Кнопка «Вид»	-помогает настроить режим отображения файлов: в виде маленьких иконок, больших иконок, детального режима с указанием всех свойств файла напротив его имени или просто списком

также имеются и другие кнопки, такие как, например,

Кнопка «Журнал»	- гипертекстовый справочник, щелкнув на любую строчку которого, можно переместиться в соответствующую папку
Кнопки «Переместить» , «Копировать» , «Удалить»	- указывая кнопкой мыши на нужный файл или папку, нажимает на одну из трех кнопок, и либо отправляем их в новую папку, либо – в «Корзину»
Кнопка «Отменить»	- служит для отмены последних операций
Кнопка „Переместить в”	- перемещение выделенного объекта или группы объектов;
Кнопка „Копировать в”	- копирование выделенного объекта в буфер обмена;
Кнопка „Вставить”	- вставка содержимого буфера обмена;
Кнопка „Свойства папки”	- вывод свойств выделенной папки;
Кнопка „Создать папку”	- создание папки в текущем каталоге;
Кнопка „Избранное”	- включает ссылки на страницы Интернет.

8. Строка меню находится в верхней части окна - это **«Файл»**; **«Правка»**; **«Вид»**; **«Избранное»**; **«Сервис»**; **«Справка»**. Контекстное меню появляется, если нажать на какой-либо объект правой кнопкой мыши. Для каждого объекта список действий свой.

«Файл»	-содержит команды для открытия, закрытия, сохранения и вывода файлов на печать
«Правка»	-содержит команды для удаления, копирования и вставки, повторения и отмены последнего выполненного действия, а также другие команды редактирования

- "Вид"** -содержит команды для отображения значков (плитка, таблица, список, значки), упорядочивания значков (по имени, размеру, типу, изменению), свойства панели инструментов, строки состояния, панели обозревателя, а также команды «Обновить» и «Переход»
- "Избранное"**-содержит команды для добавления в «Избранное» и упорядочения в «Избранном», а также «Ссылки» (для быстрого доступа к наиболее часто используемым папкам и файлам)
- «Сервис»** -содержит команды для подключения к общей сетевой папке или для отключения, для синхронизации папок, а также свойства папок
- "Справка"** -открывает раздел справочной системы Windows, а также показывает объем оперативной памяти, доступный для использованной папки или файла

9. Способы представления содержания папки можно узнать с помощью пункта меню **«Вид»** (левой кнопкой мыши по пункту «Вид» в строке меню или правой кнопкой мыши на рабочей области окна):

- «Эскизы страниц»** -в этом режиме значок папки содержит эскизы содержащихся в ней графических объектов и HTML (Интернет) – страниц, а самим графическим файлам и Интернет - страницам присваивается значок с эскизом данного файла;
- «Диафильм»** -в папках, содержащих рисунки. Эти рисунки отображаются в виде строки эскизов
- «Плитка»** -значки отображаются в два ряда, сами значки крупные;
- «Значки»** -значки отображаются в несколько столбцов, размер значков меньше;
- «Таблица»** -значки отображаются в таблице, где первый столбец – сам значок, далее идут столбцы с информацией о типе объекта, времени создания и т.д. Причём количество колонок можно задать самостоятельно, выполнив Вид/Выбор столбцов в таблице. В появившемся окне выбираются требуемые колонки, их ширина и порядок.
- «Список»** -содержимое папки отображается в виде списка имен файлов и папок, перед каждым из которых стоит маленький значок

10. Вид значков в папке можно задать и с использованием кнопки панели инструментов **«Вид»**, которая имеет раскрывающийся список, содержащий перечень режимов отображения.

Порядок отображения значков задают команды вложенного подменю **«Упорядочить значки»**. Первая группа команд сортирует содержимое окна по:

- «Имя»** -в этом способе сортировка идет вначале по имени файлов, а затем по его расширениям;
- «Размер»** -по размеру, занимаемым файлом в памяти компьютера (истинный размер);
- «Тип»** -при этом способе файлы сортируются по своему типу, а для файлов одного типа - по имени файла;
- «Изменён»**-по дате создания, модификации или последнего открытия файлов (в различной их комбинации), в порядке убывания или возрастания значений этих дат.

Вторая группа команд позволяет выбрать сортировку по умолчанию (Автоматически), выровнять значки по сетке (одноимённая команда) и сгруппировать значки по определённому признаку. То есть, если выбран режим сортировки по имени, то значки будут сгруппированы по алфавиту, если размер – по размеру и т.д.

11. Для того чтобы перейти с окна диска D: в окно диска A: можно в «Адресной строке» набрать имя диска A: и нажать Enter.

12. Для того чтобы включить режим открытия каждой папки в отдельном окне, на панели управления открываем компонент **«Свойства папки»** (нажимаем кнопку **«Пуск»** и выбираем команду **«Панель управления»**) и дважды щелкаем значок **«Свойства папки»**.

На вкладке «**Общие**» устанавливаем переключатель «**Обзор папок**» в положение «*Открывать каждую папку в отдельном окне*».

В этом режиме каждая папка будет открываться в новом окне.

По умолчанию открываемая папка отображается в одном и том же окне, поверх предыдущего содержимого. Чтобы вернуться в этот режим, выбираем вариант «*Открывать папки в одном и том же окне*».

13. Так как Windows многозадачная система, может быть загружено одновременно несколько приложений. Следовательно, на экране окажется несколько одновременно открытых окон. Чтобы расположить открытые окна так, как нам нужно, имеется 2 способа:

• можно перемещать окна, ухватившись мышкой за строку заголовка;

• можно воспользоваться контекстным меню панели задач. В этом случае необходимо щелкнуть правой клавишей мыши на панели задач и из появившегося меню выбрать одну из команд, которые определяют способ расположения окон на экране: каскадом, мозаикой, вертикально.

Если открыто несколько окон, то переключиться из одного окна в другое можно:

• щелкнуть мышкой видимую часть нужного окна;

• щелкнуть кнопку нужного окна на Панели задач;

• с помощью клавиш ALT+Tab.

14. Упорядочение окон на рабочем столе:

Открываем папку «**Мои документы**» (нажимаем кнопку «Пуск» и выбираем команду «**Мои документы**»). Дважды щелкаем папку, которую требуется просмотреть. Выбираем в меню «**Вид**» команду «**Упорядочить значки**» и затем команду *по группам*.

Примечания

Чтобы изменить способ группирования элементов в папке, выделяем в меню «**Вид**» команду «**Упорядочить значки**» и затем выбираем нужный вариант сортировки.

Чтобы изменить список вариантов сортировки, доступных в меню

«**Упорядочить значки**», выбираем в меню «**Вид**» команду «**Выбор столбцов в таблице**».

В диалоговом окне «**Выбор столбцов в таблице**» устанавливаем флажки рядом с теми элементами, которые нужно включить в меню «**Упорядочить значки**». Снимаем флажки для элементов, которые необходимо исключить из списка.

Чтобы отменить отображение файлов и папок группами, выбираем в меню «**Вид**» команду «**Упорядочить значки**» и затем команду *по группам*; галочка, стоявшая рядом с последней командой, исчезнет.

Практическое «Работа с объектами ОС Windows»

Цель занятия: отработать навыки работы с объектами ОС Windows.

Содержание работы:

№ 1. Загрузка Windows, элементы Рабочего стола и работа с ними.

1. Включите компьютер, дождитесь окончания загрузки операционной системы.
2. Рассмотрите значки, расположенные на **Рабочем столе**.
3. Разместите значки на **Рабочем столе**, расположив их по своему усмотрению, предварительно зацепив мышкой за значок и перетащив в нужном направлении.
4. Выполните обратное действие, выстроив значки автоматически по левому краю **Рабочего стола**. Выберите из контекстного меню **Рабочего стола** команду **Упорядочить значки—Автоматически**.

5. Рассмотрите **Панель задач**. Закройте и откройте **Главное меню Windows**, используя кнопку **Пуск**. Перетащите **Панель задач**, разместив ее по вертикали в правой части **Рабочего стола**. Верните ее на место.

Задание № 2. Создать папки и файлы на Рабочем столе.

Методика выполнения работы

1. На свободной поверхности **Рабочего стола** вызовите контекстное меню и выберите команду **Создать—Папку**.
2. Наберите на клавиатуре имя папки, например **Курс 1 Группа 1-...**, и нажмите клавишу **Enter**.
3. Откройте свою папку. Вы ее только что создали.
4. В текущей папке создайте еще две папки, дайте им названия **Рабочий стол** и **Разное**.
5. В папке **Рабочий стол** создайте документ **Microsoft Word** под именем **Ответ**. Для этого на свободной поверхности рабочего стола вызовите контекстное меню и выберите команду **Создать — Документ Microsoft Word**, дальше наберите на клавиатуре имя документа (файла), например **Ответ**, и обязательно нажмите клавишу **Enter**.
6. Откройте документ с именем **Ответ**. Обратите внимание: документ пуст; напечатайте названия всех объектов, находящихся на **Рабочем столе** вашего компьютера.
7. Сохраните изменения в документе, для этого выберите команду меню **Файл—Сохранить**.
8. В папке **Разное** создайте текстовый документ под именем **Картинка**. Для этого на свободной поверхности рабочего стола вызовите контекстное меню и выберите команду **Создать — Документ Microsoft Word**.
9. Наберите на клавиатуре имя документа (файла), например **Картинка**, и обязательно нажмите клавишу **Enter**. Откройте документ с именем **Картинка**. Обратите внимание: документ пуст; нарисуйте квадрат, в нем треугольник, очертите квадрат кругом.
10. Наберите предложенный текст:

ОБЯЗАННОСТИ КАДЕТА

Учащийся-кадет подчиняется командиру отделения и отвечает за точное и своевременное выполнение возложенных на него обязанностей и поставленных, ему задач.

Учащийся-кадет обязан:

- строго соблюдать установленный в колледже распорядок дня, вовремя прибывать на занятия в установленной форме одежды, подготовленным к занятиям;
- при необходимости отлучиться спросить разрешения у мастера-преподавателя, поставить в известность командира отделения;
- добросовестно учиться, настойчиво овладевать специальностями, добиваться высоких результатов по всем предметам обучения;
- дорожить честью колледжа, званием кадета, не допускать самому и удерживать товарищей от недостойных поступков;
- быть честным, трудолюбивым, дисциплинированным, проявлять разумную инициативу и находчивость;
- оказывать уважение старшим, уважать честь и достоинство товарищей по учебе, быть примером высокой культуры, скромности и выдержанности, с достоинством вести себя в общественных местах;
- бережно относиться к учебно-материальной базе колледжа, принимать участие в ее совершенствовании;
- постоянно закаливать себя, совершенствовать свою физическую подготовку, соблюдать правила личной и общественной гигиены, быть опрятным, аккуратно и чисто одетым;
- принимать активное участие в общественной, спортивной жизни группы и колледжа;
- на занятиях и при выполнении различного рода работ строго соблюдать требования безопасности.

11. Сохраните изменения в документе. Для этого выберите команду меню **Файл—Сохранить**.

Задание № 3. Удалить и восстановить объекты.

Методика выполнения работы

1. Удалите с поверхности рабочего стола папку **Курс 1 Группа 1-...**, Для этого вызовите контекстное меню и выберите команду **Удалить**. В открывшемся **Диалоговом окне** подтвердите **намерения удаления**.
2. Восстановите удаленный объект. Для этого откройте папку **Корзина**, выделите нужный объект и **восстановите** его с помощью контекстного меню.
3. Удалите с поверхности рабочего стола папку **Курс 1 Группа 1-...** Для этого выделите объект и нажмите комбинацию клавиш **Delete + Shift**, подтвердите удаление объекта. Попробуйте восстановить. Возможно ли восстановление? Если нет — почему

Задание № 4. Используя быстрый поиск объектов, найти объект calc.exe (стандартное приложение Калькулятор).

1. Из **Главного меню** запустите программу пуска (**Пуск —Найти—Файлы и папки**).
2. В диалоговое окно **Что искать** введите критерий поиска: **calc.exe**.
3. Укажите место поиска. Для этого справа от диалогового окна **Где искать** активизируйте вкладку **Стрелка** и выберите **Мой компьютер** (поиск будет вестись на внешних носителях информации).
4. Активизируйте вкладку **Найти**.
5. Запустите найденное стандартное приложение **Калькулятор**.
6. Закройте все окна на **Рабочем столе**.

?Контрольные вопросы:

1. Операционные системы семейства Windows – это?
2. Стартовый экран Windows представляет собой ...?
3. Значки - это?
4. Графический интерфейс пользователя - это?
5. Как выделить объект?
6. Как произвести вызов контекстного меню?
7. Для чего служат полосы прокруток?
8. Как создать папку?
9. Загрузка Windows, элементы Рабочего стола...
10. Как создать текстовый документ Microsoft Word?
11. Как удалить объект?

Практическое занятие «Работа со стандартными приложениями Windows: Калькулятор»

Цель занятия: научиться работать с калькулятором.

Содержание работы:

Для загрузки Калькулятора нажмите кнопку **Пуск** и выберите в **Главном меню** команду **Программы → Стандартные → Калькулятор**.

II. Измените вид Калькулятора на **Инженерный** выбрав в меню **Вид → Инженерный**.

В окне программы в инженерном режиме имеются элементы управления, позволяющие вычислять тригонометрические, степенные функции, логарифмы, а также переводить числа из одной системы счисления в другую.

Перевод чисел в другую систему счисления

Калькулятор поддерживает десятичную, двоичную, восьмеричную и шестнадцатеричную системы счисления. Этим системам соответствуют переключатели, имеющие название:

✓ **Дес** – сокращение от **Decimal** – **десятичная**;

- ✓ **Bin** – сокращение от Binary – **двоичная**;
- ✓ **Oct** – сокращение от Octal – **восьмеричная**;
- ✓ **Hex** – сокращение от Hexadecimal – **шестнадцатеричная**.

Для преобразования числа из одной системы счисления в другую, например десятичное число 28 перевести в двоичную систему, необходимо ввести данное число и щелкнуть по переключателю **Bin** или зайти в Меню **Вид** → **Двоичный(Bin)**. Вы получите на индикаторе двоичное число 11100. если затем вы щелкните по переключателю **Hex**, то получите это же число в шестнадцатеричной системе: 1С.

Задание 1: Проверить все вычисления, сделанные на уроке.

Задание 2: Заполните таблицу, в каждой строке которой одно и тоже целое число должно быть записано в различных системах счисления.

Двоичная	Восьмеричная	Десятичная	Шестнадцатеричная
101010			
	127		
		269	
			9B

Практическое занятие «Работа со справочной системой Windows»

Цель занятия: Ознакомиться с использованием справочной системы Windows.

Содержание работы:

1.1. Запустить на выполнение файл **Помощь_по_FC.ppti** просмотреть презентацию. В дальнейшем можно выполнять задание по работе, переключаясь из одной программы в другую.

1.2. Запустить справочную систему из стартового меню **Windows**.

1.3. Выполнить поиск информации по содержимому и по индексу (ключевому слову).

1.4. Завершить работу со справочной системой.

2.1. Запустить программу **Free Commander** (далее **Free Com**).

2.2. Ознакомиться с назначением каждой команды в основном меню **Free. Com.**, используя справку по **Free Com**.

2.3. Изменить размер окна **Free Com**.

2.4. Выполнить создание, копирование, переименование, удаление каталога.

2.5. Выполнить создание, копирование, перемещение, переименование, удаление, сортировку файлов, а также поиск файла(лов) по шаблону в имени или расширении.

2.6. Выполнить разбиение файла определенного объема для записи, а затем сделать сборку файла.

2.7. Выполнить выделение нескольких файлов (подряд или вразброс) для выполнения операций копирования, перемещения, переименования, удаления.

2.8. Создать ярлык папки и поместить его на рабочий стол.

2.9. Завершить работу с программой **Free Com**.

Что нужно знать и уметь делать после выполнения практического занятия.

1. Как найти информацию по содержимому и по индексу (ключевому слову).
2. Как выполнить создание, копирование, переименование, удаление каталога.
3. Как выполнить копирование, переименование, удаление файла.
4. Как выполнить разбиение файла для записи определенного объема, а затем сделать сборку файла.
5. Как выполнить сортировку (фильтр) файлов по атрибутам, по расширению, по определенному сочетанию части имени.

Практическое занятие «Настройка рабочей среды Windows: утилиты панели управления»

Цель занятия: научиться настраивать рабочую среду Windows .

Содержание работы:

Панели инструментов следует располагать так, чтобы к ним было удобнее обращаться. Панель можно присоединить к окну программы и разместить на краю или расположить в отдельном «плавающем» окне. Положение панели инструментов изменяют перетаскиванием. Присоединенную панель перетаскивают за корешок, а свободную — за строку заголовка. Строка меню в окне программы имеет те же права, что и панели инструментов: ее также можно перемещать.

Некоторые панели инструментов представлены как отделяемые палитры. Если меню или палитра содержат корешок, их можно отделить и представить свободной панелью инструментов. Наведите на корешок указатель, нажмите левую кнопку мыши и освободите палитру перетаскиванием в удобное место.

Для повышения эффективности работы состав панелей инструментов изменяют: удаляют кнопки, которые практически не используются, и добавляют те, что нужны часто. Для настройки состава панелей инструментов используют диалоговое окно Настройка.

1. Откройте панель инструментов, состав которой требуется изменить, с помощью меню Вид > Панели инструментов.

2. Дайте команду Сервис > Настройка — откроется диалоговое окно Настройка, а панели инструментов перейдут в режим корректировки.

3. Откройте вкладку Команды. В раскрывающемся списке Сохранить в выберите шаблон, на котором основан документ. Никогда не вносите изменений в стандартный шаблон Normal.dot.

4. Чтобы удалить ненужную кнопку, перетащите ее за пределы панели инструментов.

5. Найдите на вкладке Команды элемент управления, который требуется добавить на панель инструментов. Сначала выберите соответствующий пункт в списке Категории, а затем просмотрите список Команды.

6. Перетащите значок с панели Команды на панель инструментов и поместите его в нужное место панели.

7. Закройте окно Настройка кнопкой Заккрыть.

Практическое занятие «Установка ОС Windows»

Цель занятия: Выполнить установку ОС Windows.

Содержание работы:

1. Установка Windows с CD-ROM

Сейчас вы установите Windows с CD-ROM на компьютер, на котором нет разделов на жестком диске и установленных операционных систем. Если ваш компьютер не загружается с CD-ROM или если на вашем компьютере уже установлена операционная система, выполните практическое занятие 2, чтобы научиться устанавливать Windows с CD-ROM без загрузки с установочного CD-ROM.

> Запуск программы установки

1. Вставьте CD-ROM Windows в привод CD-ROM и перезагрузите компьютер.

Примечание Чтобы на некоторых компьютерах загрузиться с CD-ROM, необходимо нажать любую клавишу. Если вас попросят нажать любую клавишу для загрузки с CD-ROM, нажмите пробел.

Во время загрузки файлов программа установки выводит экран установки Windows, а затем отображает экран установки Windows.

Примечание Если вы устанавливаете демонстрационную версию Windows, нажмите Enter для продолжения (или F3, чтобы выйти из программы установки). Программа установки выведет экран приветствия.

Внимание! Вы можете использовать программу установки Windows для восстановления поврежденной системы Windows.

2. Прочитайте текст на экране приветствия и нажмите Enter для продолжения. Программа установки выводит экран с лицензионным соглашением Windows.

3. Прочитайте лицензионное соглашение и нажмите F8, чтобы согласиться с условиями соглашения.

Программа установки выводит другой экран, где сообщается о необходимости создать раздел для установки Windows.

Примечание Если вы хотите использовать только часть доступного дискового пространства, укажите требуемый объем и нажмите Enter. ____

Внимание! Необходимо создать раздел объемом не менее 2000 Мбайт.

4. Выберите часть свободного пространства, объемом не менее 2000 Мбайт, и нажмите C Программа установки попросит ввести размер раздела.

5. Если вы хотите использовать все доступное пространство для создания раздела, нажмите Enter.

Примечание Если у вас есть уже созданные разделы, то сейчас их можно удалить. Если раздел C уже создан, то его удалить нельзя, потому что программа установки скопировала на него некоторые файлы.

Программа установки выводит список существующих разделов, чтобы вы смогли выбрать раздел для установки системы. |

6. Нажмите Enter, чтобы установить Windows в созданный вами раздел. Поскольку вы устанавливаете систему в недавно созданный неформатированный раздел, программа установки Windows предлагает отформатировать этот раздел.

ОСТОРОЖНО! Если вы планируете мультизагрузочную конфигурацию с операционной системой, не поддерживающей NTFS, ваш диск C нельзя форматировать в NTFS. Вы можете установить Windows в другой раздел и отформатировать его в NTFS.

7. После соответствующего сообщения отформатируйте раздел в NTFS.

Примечание Если вы отформатируете раздел в FAT, в Windows предусмотрена команда Convert, с помощью которой вы можете преобразовать раздел в NTFS по окончании установки, при этом необходимость форматирования отпадает и вы не потеряете данные на этом разделе.

Программа установки форматирует жесткий диск, сканирует его и затем копирует файлы в папки установки Windows.

8. Когда программа установки попросит вас перезапустить компьютер, удалите все диски из приводов и нажмите Enter.

Внимание! Убедитесь в том, что вы удалили установочный CD-ROM Windows из привода CD-ROM. Если вы этого не сделаете, а ваш компьютер поддерживает загрузку с CD-ROM, то он попытается загрузиться с CD-ROM. В этом случае удалите CD-ROM, затем снова перезапустите компьютер.

Компьютер перезагрузится. Появится окно сообщений с предложением вставить установочный CD-ROM Windows в привод CD-ROM.

Запуск мастера установки

1. Вставьте CD-ROM Windows в привод CD-ROM и щелкните ОК.

Мастер установки выведет диалоговое окно требуются установочные файлы (Files Needed) с предложением проверить путь к файлам установки Windows.

2. Убедитесь, что путь к файлам установки Windows указан верно, затем щелкните ОК. Windows устанавливает файлы. На это может уйти несколько минут.

Мастер установки предлагает выбрать необходимый регион и языки.

3. Выберите соответствующую системную локализацию, пользовательскую локализацию и раскладку клавиатуры или убедитесь, что они верны для вашего языка и региона, затем щелкните **Далее** (Next). Мастер установки переходит к странице информации о пользователе, где нужно ввести ваше имя и название организации.

Программа установки автоматически использует название вашей организации для создания имени компьютера. Многие приложения в дальнейшем будут использовать эти данные для регистрации и формирования сведений об авторе документа.

4. В поле **Имя** (Name) введите ваше имя. В поле **Организация** (Organization) — название

вашей организации, затем щелкните кнопку **Далее** (Next).

Мастер установки переходит к странице **Введите код продукта** (Your Product Key).

5. Введите 25-символьный код продукта, который указан сзади, на внутренней стороне коробки с диском CD-ROM Endows, затем щелкните **Далее** (Next).

Мастер установки переходит к странице **Имя компьютера и пароль администратора**

(Computer Name And Administrator Password).

6. Введите **Pro1** в поле **Имя компьютера** (Computer Name).

Примечание Windows отображает имя компьютера прописными буквами независимо от выбранного регистра.

Если ваш компьютер расположен в сети, свяжитесь с сетевым администратором перед назначением имени вашему компьютеру. В практических занятиях этого учебного пособия используется имя **Pro1**. Если вы не назвали ваш компьютер **Pro1**, то должны использовать свое имя компьютера в каждом практическом занятии. 7. В поле **Пароль администратора** (Administrator Password) и в поле **Подтвердите пароль** (Confirm Password) введите **password**, затем щелкните **Далее** (Next).

Мастер установки отображает страницу настройки параметров модемного соединения (Modem Dialing Information).

Примечание Если мастер установки не переходит к странице настройки параметров модемного соединения, вероятно, на вашем компьютере не установлен модем. Перейдите к пункту 12.

8. Убедитесь, что страна или регион выбраны верно.

9. Введите правильный код города.

10. Если вы набираете внешний местный или международный номер, укажите дополнительный номер.

11. Убедитесь, что выбран верный тип набора номера, затем щелкните **Далее** (Next).

Мастер установки переходит к странице настройки даты и времени.

12. При необходимости скорректируйте дату и время.

13. Из раскрывающегося списка **Часовой пояс** (Time Zone) выберите часовой пояс для вашей местности.

14. Если необходим автоматический переход на летнее время и обратно, убедитесь, что установлен флажок **Автоматический переход на летнее время и обратно** (Automatically Adjust Clock For Day light Saving Changes), затем щелкните **Далее** (Next).

Мастер установки копирует некоторые сетевые файлы и затем выведет страницу настройки сетевых параметров.

Установка сетевых подключений Windows

1. Убедитесь, что выбраны стандартные параметры (Typical Settings), затем щелкните **Next** (Далее). Мастер установки отображает страницу домена или рабочей группы.

2. Убедитесь, что помечен флажок **Нет, этот компьютер не находится в сети, или подключен к сети, но не домену** (No, This Computer Is Not On A Network, Or Is On A Network Without A Domain), именем рабочей группы является WORKGROUP, затем щелкните **Далее** (Next). Мастер установки копирует файлы. Этот процесс занимает несколько минут.

Завершение установки

Мастер установки заканчивает настройку системы, копирует файлы и завершает часть установки сетевых компонентов. Затем устанавливает элементы меню **Пуск** (Start), регистрирует компоненты, сохраняет параметры настройки и удаляет временные файлы. Этот процесс занимает несколько минут.

Компьютер перезагружается, и мастер установки выводит окно **Добро пожаловать в Microsoft Windows**.

Внимание! Если компьютер попытается загрузиться с CD-ROM, извлеките CD-ROM и перезапустите компьютер.

1. Чтобы продолжить, щелкните кнопку **Далее** (Next).

Мастер установки отображает страницу **Установить прямое подключение к Internet или через локальную сеть** (Will This Computer Connect To The Internet Directly, Or Through A Network).

2. Если Вы хотите соединиться с Интернетом сейчас, выберите соответствующий вид подключения, затем щелкните **Далее** (Next). Мастер установки выводит страницу активации Windows.

Внимание! В будущем вам придется активировать Windows. Однако нет необходимости делать это сейчас, во время изучения этого учебного пособия.

3. Щелкните **Да, активировать Windows через Internet сейчас** (Yes, Activate Windows Over The Internet Now), затем щелкните **Далее** (Next).

Мастер установки отображает страницу **Регистрация в Microsoft**. (The Ready To Register With Microsoft).

4. Щелкните **Да, я хотел бы зарегистрироваться в Microsoft сейчас** (Yes, I Would Like To Register With Microsoft Now), затем щелкните **Далее** (Next).

Мастер установки отображает страницу **Сбор необходимой для регистрации информации** (Collecting Registration Information).

5. Заполните соответствующие текстовые поля.

Мастер установки отображает страницу **Готов отослать информацию** (Ready To Send Information).

6. Щелкните **Далее** (Next).

Мастер установки отображает страницу **Вы хотите подключиться к Internet сейчас?** (Do You Want To Set Up Internet Access Now).

Примечание Для этого учебного пособия доступ в Интернет не требуется. Если вы хотите подключиться к Интернету сейчас, щелкните **Да, мне требуется помощь для подключения к Internet** (Yes Help Me Connect To The Internet), щелкните **Далее** (Next) и следуйте указаниям на своем экране.

7. Щелкните **Нет, не сейчас** (No, Not At This Time), затем щелкните **Далее** (Next).

Мастер установки выводит страницу **Кто будет пользоваться этим компьютером** (Who Will Use This Computer). В ней уже должно быть указано ваше имя.

8. Введите **Fred** в качестве имени второго пользователя и щелкните **Далее** (Next). Мастер установки выводит страницу **Спасибо** (Thank you).

9. Прочитайте текст на экране и щелкните кнопку **Конец** (Finish).

10. Для начала сеанса выберите **Fred** (или имя своей учетной записи, созданной во время установки). Вы завершили установку Windows и вошли в систему как администратор.

Если ваш компьютер не может загрузиться с CD-ROM или если на вашем компьютере уже есть операционная система, то можно установить Windows с CD-ROM, не загружаясь с установочного CD-ROM Windows. Если вы успешно выполнили практическое занятие 1, то это занятие можно пропустить.

Примечание Если ваш компьютер имеет El-Torito-совместимый привод CD-ROM, вы можете установить Windows, не используя установочные диски. Запустите

программу установки, перезапустив компьютер со вставленным в привод CD-ROM установочным диском.

Запуск программы установки

На этом этапе происходит сбор информации о компьютере.

1. Если на вашем компьютере уже установлена операционная система, включите компьютер, войдите в систему и вставьте CD-ROM Windows в привод CD-ROM.

2. Когда появится экран **Добро пожаловать в Microsoft Windows**, щелкните кнопку **Установка Windows** (Install Windows).

3. Если появится окно сообщений программы установки с предупреждением, что данная версия операционной системы не может быть обновлена и возможность обновления недоступна, щелкните **ОК**. Программа установки выводит экран приветствия.

4. В окне типа установки выберите **Новая установка** (New Installation (Advanced)), затем щелкните **Далее** (Next). Программа установки отображает экран лицензионного соглашения.

5. Прочитайте лицензионное соглашение, согласитесь с его условиями и щелкните **Далее** (Next). Программа установки отображает окно для ввода кода продукта.

6. Введите 25-символьный код продукта и щелкните **Далее** (Next).

Программа установки переходит к окну настройки следующих трех возможностей:

- **Дополнительные параметры** (Advanced Options), которые позволяют назначить путь к установочным файлам, скопировать их в другое место, определить, копировать ли все инсталляционные файлы на жесткий диск, задать имя диска и раздел в течение установки;

- **Специальные возможности** (Accessibility Options), где предлагается возможность работы с экранной лупой Microsoft в течение установки, чтобы увеличить отдельные части экрана в отдельном окне для пользователей с плохим зрением, а также применение говорящего помощника Microsoft, предназначенного слепым пользователям (помощник читает содержимое экрана);

- выбор языка по умолчанию и региона.

7. После настройки нужных параметров щелкните **Далее** (Next).

Программа установки переходит к окну **Обновить установочные файлы** (Get Updated Setup Files).

Примечание Если ваш компьютер имеет доступ в Интернет, то можно поставить флажок **Да, обновить установочные файлы (Рекомендуется)** (Yes, Download The Updated Setup Files (Recommended)) и щелкнуть **Далее** (Next).

8. Выберите **Нет, пропустить этот шаг и продолжить установку Windows** (No, Skip This Step And Continue Installing Windows) и щелкните **Далее** (Next).

Если установочный раздел не отформатирован в NTFS, то мастер установки переходит к окну **Перейти на файловую систему NTFS** (Upgrade To The Windows NTFS FileSystem).

Осторожно! Если вы планируете мультизагрузочную конфигурацию с операционной системой, которая не поддерживает NTFS, то диск C: нельзя преобразовать в NTFS. Вы можете установить Windows на другой диск и отформатировать его в NTFS. Если вы устанавливаете Windows не на диск C:, при работе с учебным пособием нужно помнить, что вы используете другой диск.

9. Если мастер установки переходит к окну **Перейти на файловую систему NTFS** (Upgrade To The Windows NTFS FileSystem), выберите **Да, перейти на файловую систему NTFS** (Yes, Upgrade My Drive) и щелкните **Далее** (Next).

Если вы устанавливаете пробную версию Windows, мастер установки напомним вам, что это пробная версия.

10. Если программа установки отображает экран уведомления об установке, нажмите Enter для продолжения.

Программа установки выводит экран приветствия.

Примечание. Вы также можете использовать этот метод, чтобы обратиться к консоли восстановления для восстановления текущей установки Windows, нажав R. Чтобы выйти из программы установки, нажмите F3.

Запуск мастера установки

1. В экране приветствия нажмите Enter, чтобы установить Windows. Мастер установки предлагает выбрать область свободного дискового пространства или существующий раздел для установки системы.

Примечание. Сейчас также можно удалить разделы. Если на вашем компьютере есть диск C:, то вы не сможете удалить его, потому что программа установки уже загрузила некоторые файлы в этот раздел. Объем раздела, который вы собираетесь использовать, должен составлять не менее 2000 Мбайт. Если для установки системы не используется диск C:, то нужно его заменить в этом учебном пособии на тот диск, куда вы устанавливаете Windows.

2. Выберите диск C:. / Мастер установки выведет следующее сообщение: «Вы хотите установить Windows в раздел, где уже установлена другая операционная система. Установка Windows на этот диск может привести к некорректной работе этой операционной системы*».

3. Нажмите C, чтобы продолжить установку в этот раздел.

Осторожно! В зависимости от операционной системы, уже установленной на диске C:, программа установки может вывести следующее сообщение: «Папка \WINDOWS существует, и в ней, возможно, уже установлена Windows*. Если вы продолжите установку, то папка Windows будет перезаписана. Если вы хотите сохранить обе операционных системы, нажмите ESC и назначьте для установки другую папку».

4. Если вы получите предупреждение о существующей папке \WINDOWS, то, чтобы использовать эту папку и установить систему поверх предыдущей версии, нажмите L. Если раздел не отформатирован в NTFS и вы решите это сделать, то программа установки сначала форматирует его в NTFS, а затем копирует файлы. В противном случае она сканирует раздел и потом копирует файлы.

Мастер установки перезагружает компьютер и продолжает копировать файлы в режиме графического интерфейса.

Мастер установки выводит страницу выбора языка и региональных параметров.

5. Выберите соответствующую системную локализацию, пользовательскую локализацию и раскладку клавиатуры или убедитесь, что они верны для вашего языка и региона, затем щелкните **Далее** (Next).

Мастер установки переходит к странице информации о пользователе, где нужно ввести ваше имя и название организации. Программа установки автоматически использует название вашей организации для создания имени компьютера. Многие приложения в дальнейшем будут использовать эту информацию для регистрации программы и сведений об авторе документа.

6. В текстовом поле **Имя** (Name) введите ваше имя. В текстовом поле **Организация** (Organization) укажите название вашей организации и щелкните **Далее** (Next). Мастер установки переходит к странице **Имя компьютера и пароль администратора**.

7. Введите **Pgo1** в поле **Имя компьютера** (Computer Name).

Windows XP Professional отображает имя компьютера прописными буквами независимо от регистра.

Осторожно! Если ваш компьютер находится в сети, свяжитесь с сетевым администратором перед назначением имени компьютеру. В упражнениях практических

занятий это-го учебного пособия используется имя **Pro1**. Если вы назвали компьютер как-то иначе, то должны использовать это имя в каждом практическом занятии.

8. В поле **Пароль администратора** (Administrator Password) и в поле **Подтвердите пароль** (Confirm Password) введите **password**, затем щелкните **Далее** (Next).

Внимание! В практических занятиях этого учебного пособия вы будете использовать слово password в качестве пароля для учетной записи администратора. При реальной работе следует назначать для учетной записи администратора сложный пароль (который трудно угадать). Microsoft рекомендует использовать символы верхнего и нижнего регистров, числа и символы (например, Lp6*g9f2),

Мастер установки отображает страницу настройки параметров модемного соединения.

Примечание. Если мастер установки не переходит к странице настройки параметров модемного соединения, вероятно, на вашем компьютере не установлен модем. Перейдите к пункту 13.

9. Убедитесь, что страна или регион выбраны верно.

10. Введите правильный код города.

11. Если вы набираете внешний местный или международный номер, укажите дополнительный номер.

12. Убедитесь, что выбран правильный тип набора номера, и щелкните **Далее** (Next). Мастер установки переходит к странице настройки даты и времени.

13. При необходимости скорректируйте дату и время. Из раскрывающегося списка **Часовой пояс** (Time Zone) выберите часовой пояс для вашей местности.

14. Если вам нужен автоматический переход на летнее время и обратно, убедитесь, что установлен флажок **Автоматический переход на летнее время и обратно** (Automatically Adjust Clock For Daylight Saving Changes), затем щелкните **Далее** (Next).

Примечание Если у вас реализована мультизагрузочная конфигурация с другой операционной системой, которая также может осуществлять переход на летнее время и обратно, во избежание повторного перехода поставьте флажок в той операционной системе, которую используете чаще.

Мастер установки установит некоторые сетевые файлы и затем выведет страницу настройки сетевых параметров.

***• Установка сетевых подключений Windows**

1. Убедитесь, что выбраны стандартные установки (Typical Settings), затем щелкните **Да - все** (Next).

Мастер установки отображает страницу домена или рабочей группы.

2. Убедитесь, что установлен флажок **Нет, этот компьютер не находится в сети или подключен к сети, но не домену** (No, This Computer Is Not On A Network, Or Is On A Network Without A Domain), именем рабочей группы является WORKGROUP, затем щелкните **Далее** (Next). Мастер установки копирует файлы. Этот процесс занимает несколько минут.

Завершение установки

Мастер установки заканчивает настройку системы, копирует файлы и завершает часть установки сетевых компонентов. Затем он устанавливает элементы меню **Пуск** (Start), регистрирует компоненты, сохраняет параметры настройки и удаляет временные файлы. Этот процесс занимает несколько минут.

Компьютер перезагружается, и мастер установки выводит окно **Добро пожаловать в Microsoft Windows**.

Внимание! Убедитесь, что вы удалили CD-ROM Windows XP Professional из привода CD-ROM. Если вы не сделали этого и ваш компьютер поддерживает загрузку

с CD.-ROM, он может попытаться загрузиться с CD-ROM. В этом случае извлеките CD-ROM и перезапустите компьютер.

1. Чтобы продолжить, щелкните кнопку **Далее (Next)**.

Мастер установки отображает страницу **Установить прямое подключение к Internet или через локальную сеть** (Will This Computer Connect To The Internet Directly, Or Through A Network).

2. Если вы хотите соединиться с Интернетом сейчас, выберите соответствующий вид подключения, затем щелкните **Далее (Next)**.

• **Примечание.** Если вы не хотите сейчас подключаться к Интернету, щелкните кнопку **Пропустить (Skip)**.

Мастер установки выводит страницу активации Windows.

3. Щелкните **Да, активировать Windows через Internet сейчас** (Yes, Activate Windows Over The Internet Now), затем щелкните **Далее (Next)**.

Мастер установки отображает страницу **Регистрация в Microsoft** (The Ready To Register With Microsoft).

4. Щелкните **Да, я хотел бы зарегистрироваться в Microsoft сейчас** (Yes, I Would Like To Register With Microsoft Now), затем щелкните **Далее (Next)**.

Мастер установки отображает страницу **Сбор необходимой для регистрации информации** (Collecting Registration Information).

5. Заполните соответствующие текстовые поля.

Мастер установки отображает страницу **Готов отослать информацию** (Ready To Send Information).

6. Щелкните **Далее (Next)**.

Мастер установки отображает страницу **Вы хотите подключиться к Internet сейчас?** (Do You Want To Set Up Internet Access Now).

Примечание. Для этого учебного пособия доступ в Интернет не требуется. Если вы хотите подключиться к Интернету сейчас, щелкните **Да, мне требуется помощь> для подключения к Internet** (Yes Help Me Connect To The Internet), затем — **Далее (Next)** и следуйте указаниям на экране.

7. Щелкните **Нет, не сейчас** (No, Not At This Time), затем щелкните **Далее (Next)**.

Мастер установки выводит страницу **Кто будет пользоваться этим компьютером** (Who Will Use This Computer). В ней уже должно быть указано ваше имя.

8. Введите **Умей** в качестве имени второго пользователя и щелкните **Далее (Next)**. Мастер установки выводит страницу **Спасибо** (Thank you).

9. Прочитайте страницу и щелкните кнопку **Конец** (Finish).

10. Для начала сеанса выберите **Fred** (или имя своей учетной записи, созданное в течение установки).

Вы завершили установку Windows и вошли в систему как администратор.

Практическое занятие «Установка виртуальной компьютерной сети на основе операционных систем Windows»

Цель работы: создать модель компьютерной сети предприятия на основе виртуальной.

Содержание работы:

Применение виртуального компьютера позволяет создать гибкую в настройках и безопасную для реального компьютера среду, в которой студент обладает правами администратора, что позволяет изучать все аспекты применения операционных систем без вмешательства в настройки реального (физического) компьютера. Это создает уникальные возможности для изучения любых ОС в составе сети предприятия без необходимости их установки на реальном компьютере.

Под управлением основной системы могут быть одновременно запущены любые операционные системы и процесс изучения ОС, приобретения и тестирования навыков проходит на порядок быстрее. Изолированность виртуальной машины от основной операционной системы исключает возможность распространения вирусов или срабатывания вредоносных механизмов исследуемого программного обеспечения.

Виртуальный компьютер представлен файлами на диске реального компьютера и может быть легко перенесен с одного компьютера на другой.

Создав одну виртуальную машину с нужным набором программного обеспечения, в течение нескольких минут можно установить ее на все машины компьютерного класса. Ничего страшного не произойдет, если обучаемый в процессе освоения преподаваемых технологий умышленно или нечаянно разрушит подопытную среду. Для восстановления поврежденной виртуальной машины из резервной копии понадобится всего несколько минут. При выполнении лабораторных работ все тестовые сети и компьютеры, находящиеся внутри них, создадим, используя средства комплекса виртуальных машин.

Указания к выполнению лабораторной работы

Создание виртуальных компьютеров для лабораторного комплекса необходимо выполнить в следующей последовательности:

1. Реализовать действия, необходимые для установки MicrosoftVirtualPC2004/2007 на компьютере учебного класса с операционной системой Windows XP или Windows Vista. Установка выполняется путем запуска установочного файла setup.exe из дистрибутива MicrosoftVirtualPC2004/2007, являющегося бесплатным и свободно распространяемым продуктом.

При запуске виртуальной машины появляется консоль управления виртуальными компьютерами (рисунок 1.1), предоставляющая возможность установки любых операционных систем и работы с ними после установки, как по отдельности, так и в составе компьютерной сети.

2. Для установки операционной системы на виртуальной платформе необходимо выбрать пункт New и далее Create a virtual machine. При переносе уже имеющейся машины на другой компьютер выбираем пункт Add an existing Virtual machine, позволяющий добавить в данный контейнер ранее созданный виртуальный компьютер.

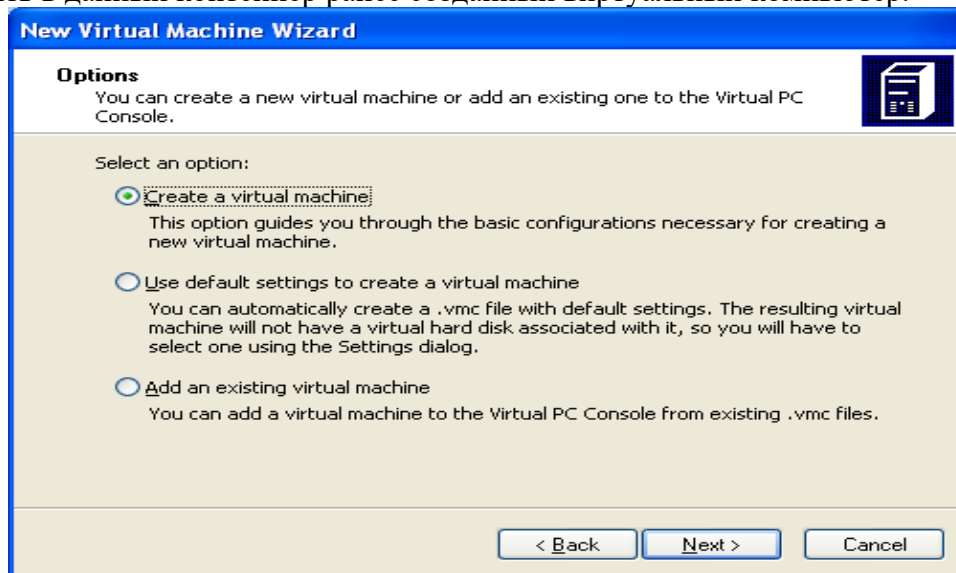


Рисунок 1.1 Выбор варианта установки виртуального компьютера

При установке выбираем объем оперативной памяти, достаточной для функционирования устанавливаемой ОС, и вариант создания жесткого диска машины, как это показано на рисунках 1.2 и 1.3.

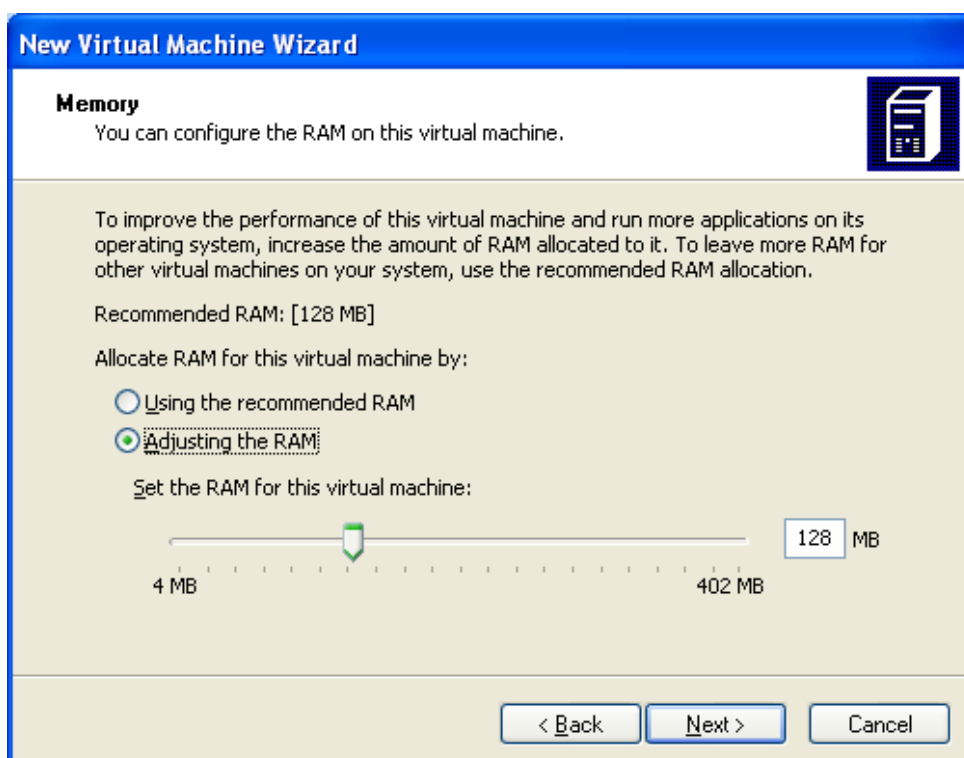


Рисунок 1.2 Выбор объема оперативной памяти виртуальной машины

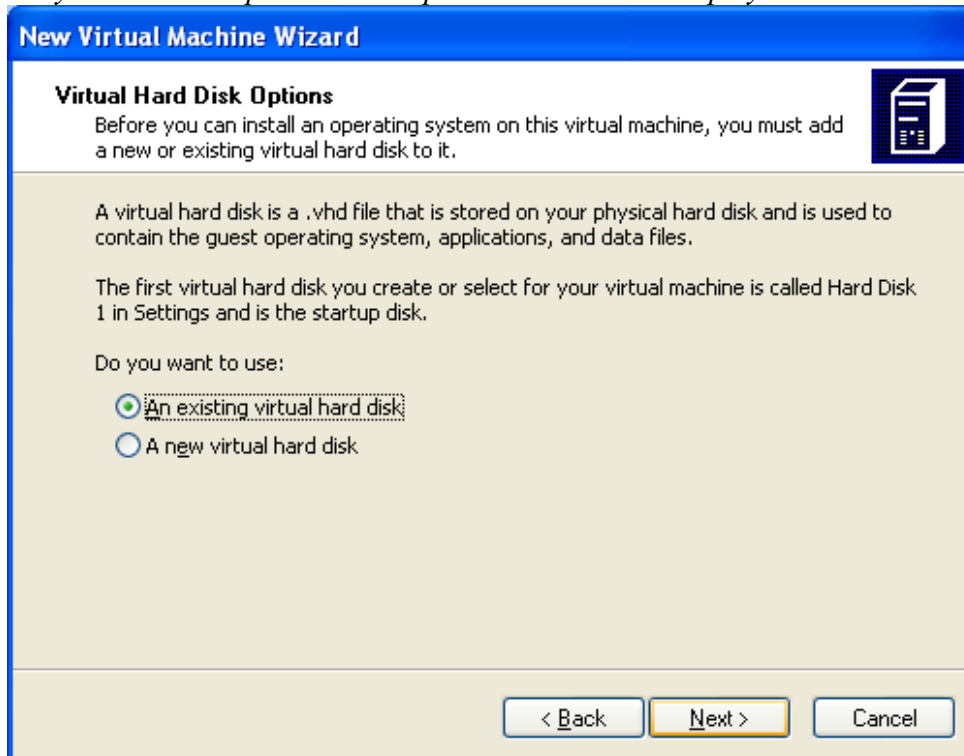


Рисунок 1.3 Выбор варианта создания жесткого диска виртуальной машины

Машину можно установить на новый виртуальный жесткий диск (A new virtual hard disk) или использовать диск, созданный ранее (An existing virtual hard disk).

После создания новой виртуальной машины ее имя появляется в консоли и ее можно запустить для установки операционной системы (рисунок 1.4).

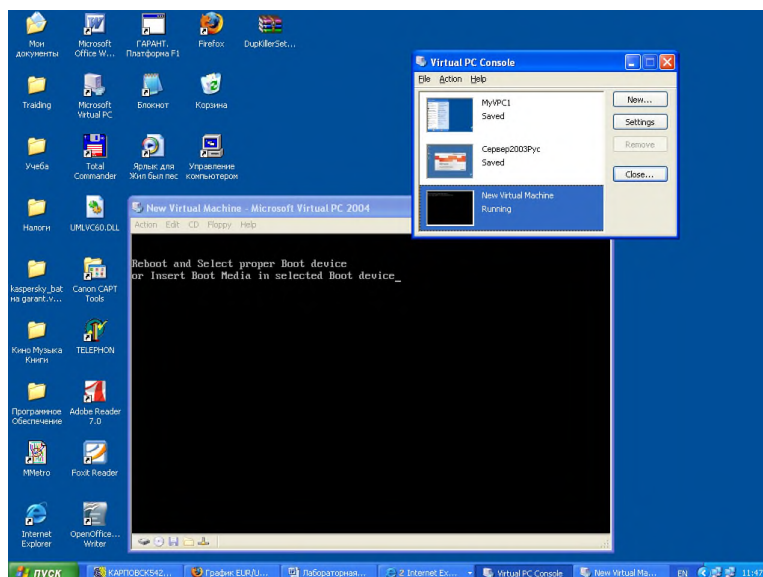


Рисунок 1.4 Запуск виртуальной машины для установки операционной системы

Для установки операционной системы необходимо иметь ее дистрибутив на CD/DVD диске или файл образа в формате ISO.

В первом случае в меню виртуального компьютера CD выберите Use physical drive и виртуальная машина будет использовать привод реального компьютера. При наличии ISO-образа выбираем пункт Capture ISO image и указываем соответствующий файл.

Далее установка операционной системы ничем не отличается от ее установки на реальном компьютере.

При установке ОС необходимо выбрать размер виртуального жесткого диска, достаточный для установки операционной системы и предполагаемого к установке ПО, а при форматировании диска обязательно выбрать опцию «Форматировать раздел в системе NTFS». Далее при установке следует выбирать стандартные настройки, предлагаемые по умолчанию. По окончании установки системы новый виртуальный компьютер появляется в консоли Управления виртуальной машиной и может быть запущен кнопкой Start.

Аналогично устанавливается операционная система WindowsServer2003. При установке выбираем все варианты по умолчанию, так как настройки сети и серверов предполагается выполнить позже в ходе выполнения лабораторных работ.

Среда, моделирующая компьютерную сеть предприятия, образуется при одновременном запуске виртуальных машин с серверной и клиентской операционными системами. Для организации сетевого взаимодействия виртуальных компьютеров необходимо настроить виртуальные сетевые соединения. Для этого используется раздел Settings установки параметров Virtual PC, в котором предусмотрено несколько режимов настройки сети. Для того, чтобы создать изолированную от реальной системы виртуальную сеть выберем режим Localonly, в котором виртуальные машины взаимодействуют только между собой.

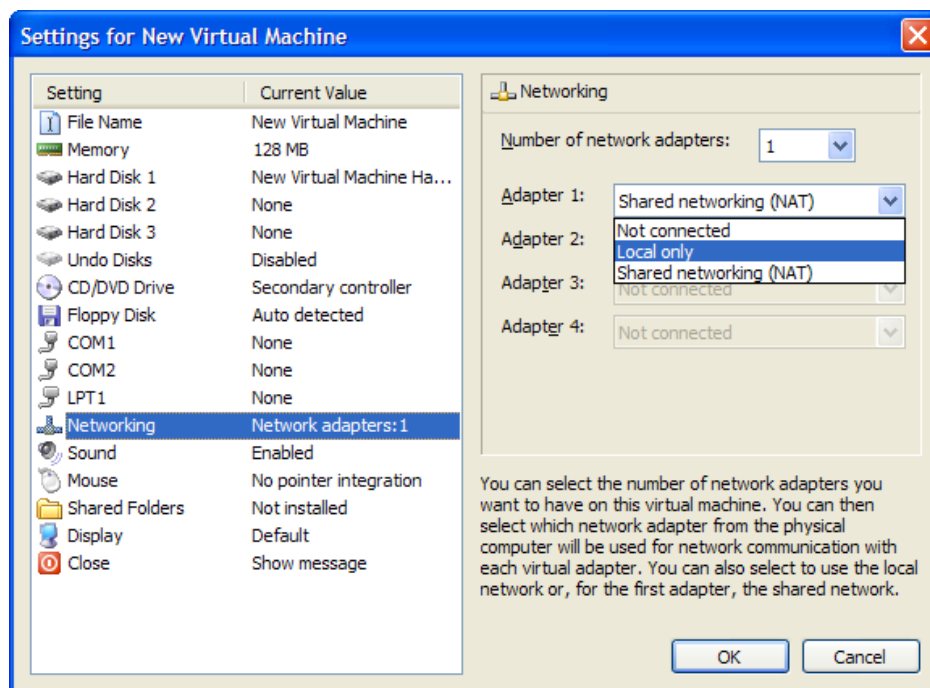


Рисунок 1.5 Выбор варианта сетевого взаимодействия виртуальных машин

Далее необходимо настроить протокол TCP/IP на сервере и клиентской машине. Настройка протокола TCP/IP на сервере в данном случае сводится к установлению постоянного (статического) IP-адреса виртуального компьютера с операционной системой Windows Server.

Для этого откроем окно свойств подключения по локальной сети и выберем компонент Протокол Интернета (TCP/IP) (рисунок 1.6).

Настроим свойства протокола, установив IP-адрес 192.168.1.1 и маску сети 255.255.255.0. Аналогично задается IP-адрес для клиентской машины. Установим его равным 192.168.1.10. В качестве шлюза по умолчанию укажите адрес сервера.

Проверим взаимодействие виртуальных компьютеров по сети. Для этого в режиме командной строки (выполняя Пуск/Выполнить/cmd.exe) введем команду проверки функционирования сети `ping 192.168.1.1` на машине – клиенте и `ping 192.168.1.10` на сервере.

Если сетевые адаптеры и протоколы взаимодействуют верно, то результаты выполнения команд будут показывать наличие обмена данными.

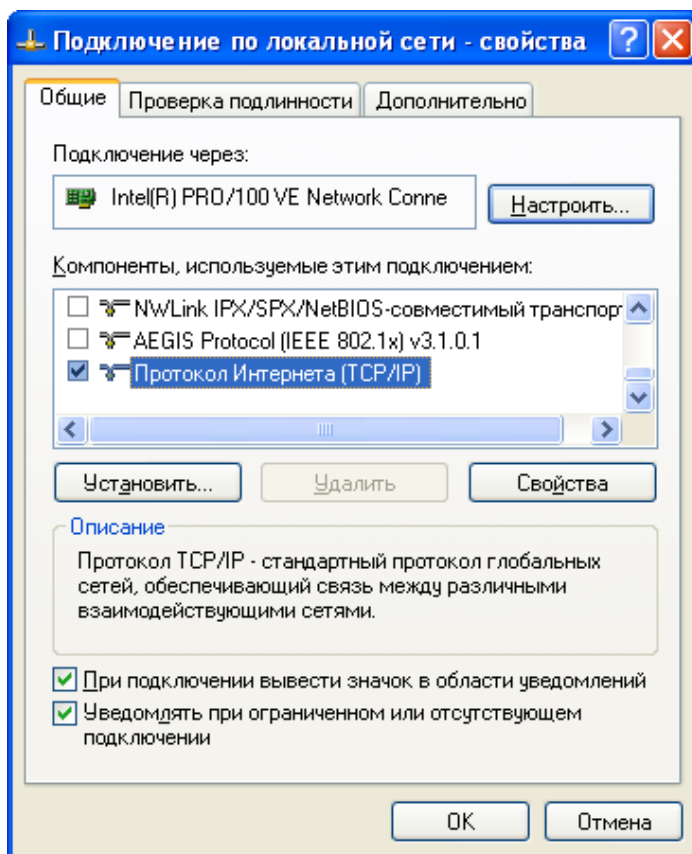


Рисунок 1.6 Выбор протокола TCP/IP для настройки IP-адресов

Таким образом, у нас получается рабочий макет стандартной схемы локальной сети масштаба предприятия. Для простоты понимания учебного примера проведена миниатюризация, заключающаяся в том, что в каждую сеть помещен только один компьютер. Этого достаточно для демонстрации обсуждаемых концепций и изучения ОС в составе лабораторного комплекса.

Задание на выполнение лабораторной работы

Одна из бригад студентов, выбранная преподавателем, выполняет следующее задание:

1. Установить виртуальную машину.
2. Установить на виртуальном компьютере операционную систему Windows XP или Vista.
3. Установить на следующем виртуальном компьютере операционную систему WindowsServer2003 или WindowsServer2008.
4. Настроить сетевое соединение компьютеров, как это предписано в методических указаниях.
5. Проверить взаимодействие компьютеров через сеть.
6. Создать копии файлов виртуальных машин на мобильном запоминающем устройстве – DVD– диске или Flash– носителе.
7. Остальные бригады студентов получают копии файлов виртуальных машин и создают на их основе виртуальные машины на своих рабочих местах.
8. Каждая бригада отчитывается перед преподавателем, демонстрируя работу виртуальных компьютеров, установленных ими на рабочем месте студента.

Контрольные вопросы

1. Назовите преимущества использования виртуальной машины при изучении операционных систем.
2. Назовите основные шаги установки виртуального компьютера.
3. Как установить виртуальную машину с параметрами по умолчанию?

4. Как установить виртуальную машину с использованием файлов имеющейся виртуальной машины?
5. Назовите способы установки операционных систем на виртуальную машину.
6. Каким образом выполняется выбор режимов работы сетевых адаптеров виртуальной машины?
7. Каким образом можно установить созданную виртуальную машину на другом компьютере?
8. Почему в данном случае в качестве файловой системы виртуальных машин необходимо выбирать систему NTFS?
- 9.

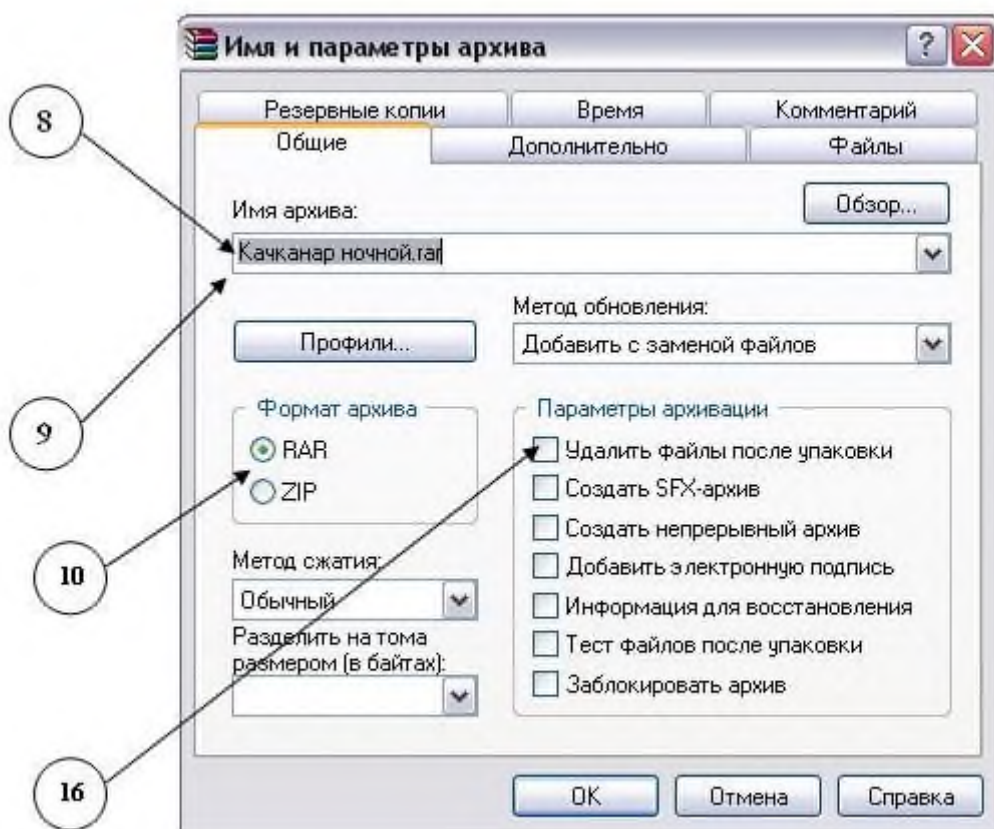
Практическое занятие «Архивирование файлов»

Цель занятия: научиться архивировать файлы с разными значениями сжатия.

Содержание работы:

1. Создайте в своей рабочей папке (папке с вашей группой) следующие папки: папку со своей фамилией, в ней папки **Архивы**.
2. Запустите программу **Win Rar**.
3. Откройте на компьютере учителя папку с исходным материалом для практической работы **Практикум**. В этой папке хранятся три типа файлов **.doc**, **.bmp**, **.exe**.

Ответьте на вопрос письменно в тетради, какого типа документы имеют вышеперечисленные расширения?



4. Скопируйте в папку **Архивы** файлы из папки **Практикум**.
5. Заархивируйте графический файл и сравните размеры обоих файлов. Для этого выполните следующие действия:
6. Щелчком правой кнопки мыши выделите файл типа **.bmp**

7. Щелкните на кнопке **Добавить в архив...**, появится диалоговое окно, уточняющее параметры архивации.
8. По умолчанию архивный файл имеет имя исходного файла.
9. Если необходимо задать иное имя архива, то введите его в поле ввода имени.
10. Выберите формат архивного файла, например RAR.
11. Остальные параметры оставьте без изменения.
12. Щелкните по кнопке **Ок**.
13. Сравните размеры исходного файла и архивного. Данные внесите в **таблицу 1**.
14. Заархивируйте файл типа .doc и сравните размеры обоих файлов. Данные внесите в **таблицу 1**.
15. Заархивируйте файл типа .exe и сравните размеры обоих файлов. Данные внесите в **таблицу 1**.
16. Удалите исходные файлы.
17. Заархивируйте файлы в формате архива **ZIP**. Заполните таблицу 1. полученными данными.

Внимание!!! Возможно упаковывать файлы с их последующим удалением, если был выбран такой метод.

Таблица 1

Формат архива	Имя файла и его расширение	Исходный размер	Размер после архивации

При создании нового архива нужно задать параметры архивирования. Прежде всего, необходимо задать имя архивного файла и место его сохранения на диске. Далее, нужно выбрать **формат архивации** RAR или ZIP (формат ZIP более широко распространен, а метод RAR обеспечивает больше возможностей и более сильное сжатие).

В обоих форматах поддерживаются шесть **методов архивации**: **Без сжатия**, **Скоростной**, **Быстрый**, **Обычный**, **Хороший** и **Максимальный**. Максимальный метод обеспечивает наиболее высокую степень сжатия, но с наименьшей скоростью. Напротив, Скоростной сжимает плохо, но очень быстро. Метод Без сжатия просто помещает файлы в архив без их упаковки. Если вы создаете архив для передачи по компьютерным сетям или для долговременного хранения, имеет смысл выбрать метод Максимальный для получения наилучшего сжатия. Если же вы создаете ежедневную резервную копию данных, то, как правило, лучше использовать Обычный метод. Следующий параметр архивации — размер словаря. Он может принимать значения 64, 128, 256, 512 и 1024 Кб. Чем больше размер словаря, тем лучше, но медленнее сжатие.

WinRAR позволяет создавать многотомные архивы, то есть архивы, состоящие из нескольких частей. Обычно тома используются для сохранения большого архива на нескольких дискетах или других сменных носителях. Первый том архива имеет обычное расширение rar, а расширения последующих томов нумеруются как r00, r01, r02 и так далее.

Архив может быть **непрерывным** (позволяет добиться максимальной степени сжатия) и **самораспаковывающимся** (SFX, от англ. Self-eXtracting). Для разархивации такого архива не нужна специальная программа, достаточно запустить файл архива на выполнение, так как он является исполняемым файлом и имеет расширение .exe.

Для архивации звуковых и графических файлов может использоваться

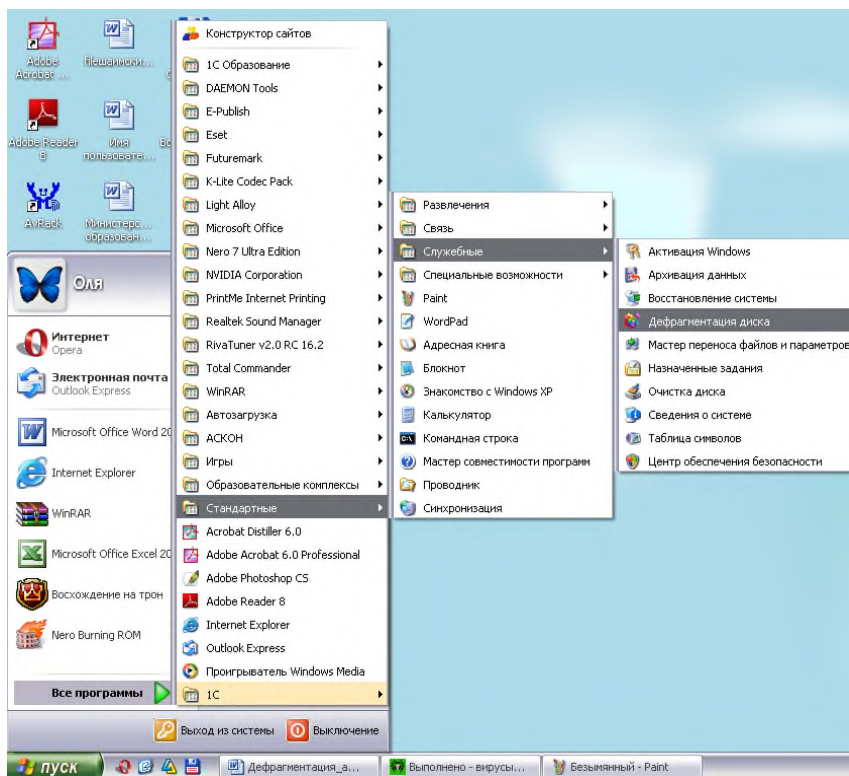
дополнительный специальный метод **мультимедиа сжатие**, при котором может быть достигнута на 30% более высокая степень сжатия, чем при обычном сжатии.

Практическое занятие «Дефрагментация, архивирование и форматирование дисков»

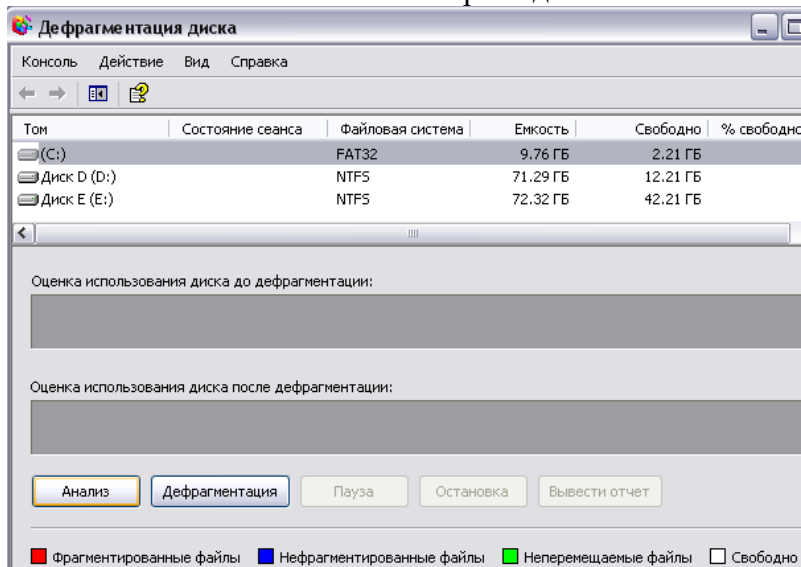
Цель занятия: Выполнить дефрагментацию, архивирование и форматирование диска.

Содержание работы:

1. Выберите программу для дефрагментации диска (*Пуск, Стандартные, Служебные, Дефрагментация диска*):



2. В появившемся окне выберите диск *C* и нажмите кнопку «Анализ».



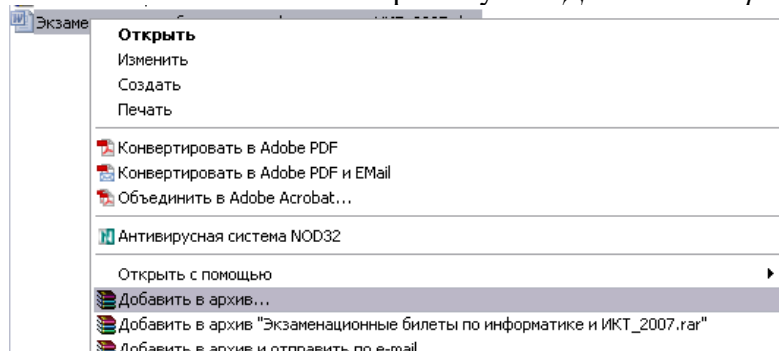
3. После анализа появится сообщение о том, требуется ли дефрагментация этого диска. Если это необходимо - выполните дефрагментацию диска *C*.

Архивирование файлов

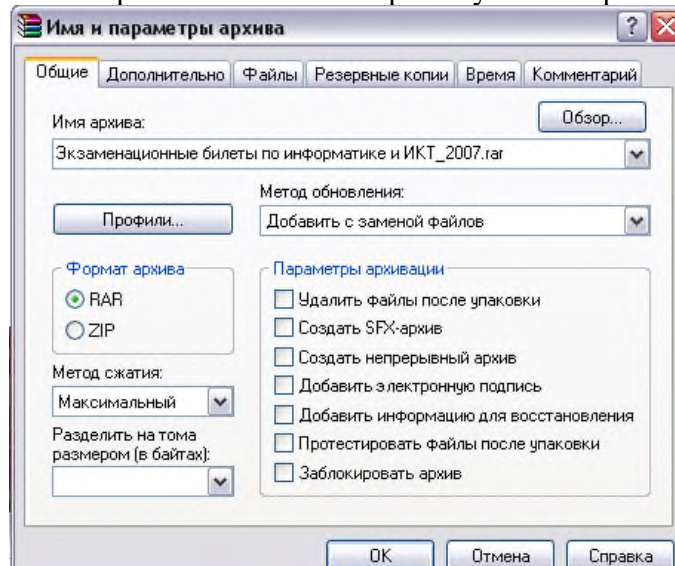
Найдите в общих документах папку «Архивирование». В ней 3 разных файла (музыкальный, графический, текстовый). Заархивируйте каждый из них. Сравните объемы исходных файлов и заархивированных. Сделайте вывод – какие файлы лучше архивируются и почему.

Этапы архивирования:

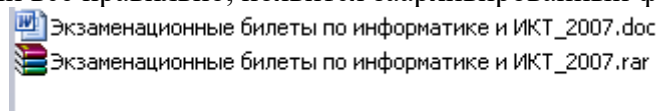
1. Щелкните правой кнопкой мыши по файлу, который хотите заархивировать. Появится контекстное меню. Выберите пункт «Добавить в архив».



2. В открывшемся окне выбрать нужные параметры:



4. Щелкнуть по кнопке «ОК». Просмотреть папку «Архивирование». Если вы сделали все правильно, появится заархивированный файл:



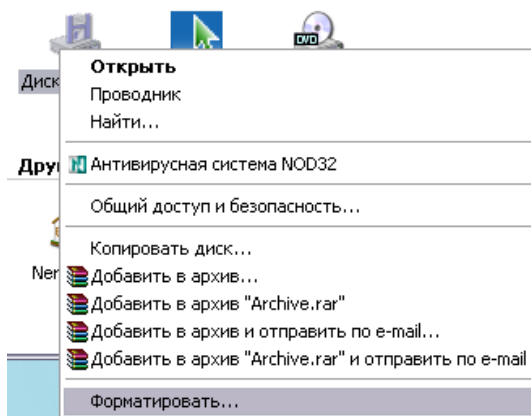
Примечание: Для того чтобы создать самораспаковывающийся архив поставьте флажок на пункте

☒ Создать SFX-архив

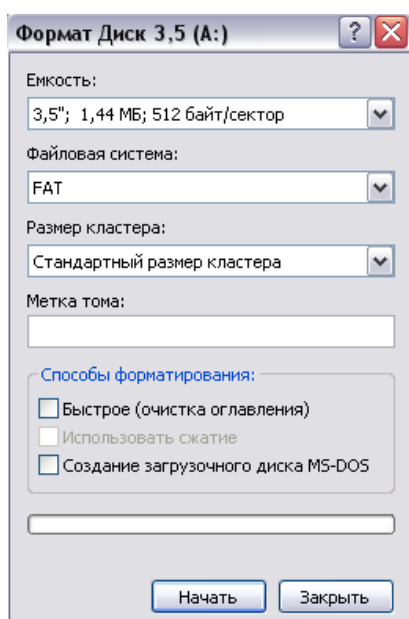
Попробуйте такой вид архивации на одном из файлов. Сравните способы открывания разных архивов.

Форматирование дискеты

1. Возьмите у преподавателя дискету, вставьте её. Откройте *Мой компьютер*. Щелкните по диску *A* правой кнопкой мыши. Выберите пункт «Форматировать»:

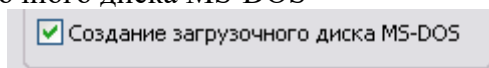


2. Выберите нужные параметры:



Нажмите кнопку «Начать».

Примечание: для создания загрузочной дискеты выберите пункт: Создание загрузочного диска MS-DOS



Если у вас осталось время выполните это действие.

Практическое занятие «Использование служебных программ. Резервное копирование»

Цель занятия: получение практических навыков в конфигурировании ОС Windows для ПК в соответствии с его функциональным назначением.

Содержание работы:

Задание 1. Удалите с вашего ПК файловый менеджер и установите его, пользуясь имеющимся дистрибутивом.

Задание 2. Оцените занимаемый объем следующих служб:

- 1.1. Службы доступа к файлам и принтерам в сети.
- 1.2. Обновление корневых сертификатов.
- 1.3. Очереди сообщений.
- 1.4. Службы индексирования.

- 1.5. Службы факсов.
- 1.6. Средства управления и наблюдения.

Задание 3. Выполните следующие действия:

1. Произведите проверку диска 2-я способами:
 - используя интерфейс Windows;
 - используя командную строку.
2. Запустите программу дефрагментации дисков. Проанализируйте 2 тома. После проведения анализа отображается диалоговое окно с сообщением о проценте фрагментированных файлов и папок в том и рекомендацией к действию, сохраните информацию анализа в файл.
3. Убедитесь, что на наименьшем по объему диске не менее 15% свободного пространства и выполните его дефрагментацию.
4. Выполните п.2,3 с помощью командной строки
Введите **defrag том [/a] [/v]**
5. Выполните очистку диска от всех временных файлов.
6. Выполните быстрое форматирование диска A:, с заданием метки.
7. Выполните форматирование флэш-карты в файловой системе NTFS.

Контрольные вопросы

1. Каково назначение служебных программ Windows?
2. Как осуществляется доступ к служебным программам Windows?
3. Для чего используется программа проверки диска?
4. Каково назначение программы дефрагментации диска?
5. Что такое дефрагментация диска?
6. Что подразумевается под «очисткой диска»?
7. Что такое форматирование дисков? В каких случаях прибегают к форматированию?

**Практическое занятие «Обслуживание системы, восстановление системы.
Работа с реестром»**

Цель занятия: Научиться самостоятельно настраивать параметры реестров в ОС Windows

Содержание работы:

- 1) перед работой с реестром сделать резервную копию:
Пуск- выполнить- regedit- реестр- экспорт файлов реестра- весь реестр- файлы реестра- имя файла (называть под своей фамилией)-И-206к-сохранить на рабочем столе
 - 2) ускорим открытие меню кнопки «Пуск»:
Hkey_Current_User\ Control Panel\ Desktop
ПКМ в правой части диалогового окна создать строковый параметр MenuShowDelay, значение присвоить равное 0
 - 3) проверить (закрыть все окна и перезагрузить ОС)
 - 4) с помощью реестра можно получить список шрифтов, используемых в ОС Windows:
Hkey_Local_Machine\ Software\ Microsoft\ Windows\ Current Version\ Fonts
Tahoma-tahoma.ttf
Impact-impact.ttf
Symbol-symbol.ttf
 - 5) с помощью реестра можно задать автозагрузку программы пользователя при начальной загрузке Windows:
выбрать программу и посмотреть полный путь загрузочного файла этой программы:
Hkey_Local_Machine\Software\Microsoft\Windows\Current Version\Run
Создать новый строковый параметр, имя- имя программы, значение- полный путь к выполняемому файлу
 - 6) перезагрузить, написать отчет согласно приложению №1
- Теоретические вопросы:

1. Что такое интерфейс и пользовательский интерфейс?
2. Какой основной принцип работы ОС Windows? Какие виды объектов ОС Windows Вы знаете?
3. Опишите основные объекты ОС Windows

Практическое занятие «Распределение прав пользователя»

Цель занятия: научиться создавать распределение прав пользователей.

Содержание работы:

Метод 1. Настройка новой установки

Новая установка не вызывает никаких проблем, поскольку в этом случае не требуется изменять существующую настройку. В этом примере используются две учетные записи Windows:

SteveAdmin. Первая учетная запись создается в процессе установки. Ее следует использовать исключительно для задач администрирования.

Steve. Вторая учетная запись — это учетная запись обычного пользователя с ограниченными правами. Она используется для повседневной работы.

Встроенная учетная запись **Администратор** не будет использоваться и останется отключенной.

Чтобы настроить ОС Windows 7, выполните следующие действия.

Установка ОС Windows 7, создание пользователя SteveAdmin

Это обычный процесс установки с DVD-диска. На выполнение начальных этапов потребуется некоторое время (и по меньшей мере одна перезагрузка), после чего появятся вопросы, касающиеся настройки пользователей.

При запросе введите имя первого пользователя **SteveAdmin**. Эта учетная запись автоматически создается с правами администратора.

Если учетной записи присваивается пароль, запомните его — он потребуется для выполнения всех задач администрирования на компьютере.

Завершение установки Windows 7

На этом этапе выполняется настройка автоматического обновления, установка необходимых драйверов, настройка сети и т. д.

Все эти действия выполняются административным пользователем **SteveAdmin**.

Создание новой учетной записи обычного пользователя Steve

После входа в систему с учетной записью **SteveAdmin** откройте панель управления.

Щелкните значок "Пуск".

Выберите пункт "Панель управления".

В разделе **Учетные записи пользователей и семейная безопасность** выберите **Добавление и удаление учетных записей пользователей**.

Под списком имеющихся учетных записей выберите **Создать новую учетную запись**.

В диалоговом окне введите имя нового пользователя **Steve** и выберите вариант **Обычный доступ**.

Нажмите кнопку **Создать учетную запись**.

Присвоение пароля новой учетной записи Steve (при необходимости)

После создания учетной записи появится список имеющихся пользователей с заголовком "Выберите учетную запись для изменения". Щелкните значок созданной учетной записи обычного пользователя **Steve**.

Выберите **Создание пароля** и введите пароль (дважды!) и подсказку для пароля (если требуется).

Метод 2. Преобразование уже установленной учетной записи администратора

Этот метод применяется, если операционная система Windows 7 уже установлена, а устанавливающий пользователь (в данном случае **Steve**) был автоматически создан с

правами администратора. С технической точки зрения можно было бы изменить имя учетной записи на **SteveAdmin** и создать новую учетную запись **Steve** для пользователя с ограниченными правами, однако это привело бы к повреждению профилей пользователей, рабочего стола и других личных настроек. Несмотря на наличие возможности копировать профили, проще создать новую учетную запись администратора и понизить уровень имеющейся учетной записи.

Ниже приводится описание действий.

Создание нового пользователя SteveAdmin

Войдите в систему в качестве пользователя **Steve**, который все еще является административным пользователем, и откройте панель управления для создания нового пользователя.

Щелкните значок "Пуск" и выберите пункт "Панель управления".

В разделе **Учетные записи пользователей и семейная безопасность** выберите **Добавление и удаление учетных записей пользователей**.

Под списком имеющихся учетных записей выберите **Создать новую учетную запись**.

В диалоговом окне введите имя нового пользователя **SteveAdmin** и выберите параметр **Администратор**.

Нажмите кнопку **Создать учетную запись**.

Итак, теперь у нас есть новая учетная запись **SteveAdmin**, но пока без пароля! А в системе зарегистрированы два пользователя с правами администратора.

Присвоение пароля новой учетной записи SteveAdmin (при необходимости)

После создания учетной записи появится список имеющихся пользователей с заголовком "Выберите учетную запись для изменения". Щелкните значок новой учетной записи **SteveAdmin**, под которым указано, что этот пользователь является администратором.

Выберите **Создание пароля** и введите пароль (дважды!) и подсказку для пароля (если необходимо).

На этом создание учетной записи **SteveAdmin** завершается. На компьютере остаются два пользователя с правами администратора.

Не закрывайте диалоговое окно! Мы перейдем к следующему этапу прямо сейчас.

Понижение прав пользователя Steve

При наличии настроенной учетной записи **SteveAdmin** самое время понизить права исходного устанавливающего пользователя **Steve** от администратора до обычного пользователя. Поскольку мы по-прежнему находимся в панели управления, продолжим работу с того места, где остановились.

Выберите **Управление другой учетной записью**.

Щелкните значок учетной записи **Steve**.

Выберите **Изменение типа своей учетной записи**.

Выберите **Обычный доступ**.

Нажмите кнопку **Изменение типа учетной записи**.

Закройте диалоговые окна панели управления.

При следующем входе пользователя **Steve** в систему он будет иметь права только обычного пользователя.

Выход и последующий вход пользователя Steve в систему

При выходе из системы удаляется маркер сеанса с правами администратора. При следующем входе в систему пользователь получает новый набор ограниченных прав.

После входа в систему в качестве пользователя с ограниченными правами при попытке выполнения административных задач появится запрос контроля учетных записей на ввод учетных данных пользователя **SteveAdmin**.

Отключение учетной записи администратора

К данному моменту в одной из двух процедур был настроен пользователь **Steve** с

ограниченными правами и учетная запись администратора **SteveAdmin**, однако некоторые пользователи могли ранее включить встроенную учетную запись **Администратор**.

Я считаю, что этого делать не следует, и рекомендую отключить эту учетную запись. Эти действия не требуется выполнять, если операционная система Windows 7 только что установлена или если учетная запись **Администратор** не отображается на странице входа в виде значка пользователя, который может войти в систему.

В случае сомнений действия, которые требуется выполнить для проверки и отключения, практически идентичны.

Открытие приложения "Управление пользователями"

Включение и отключение учетных записей выполняется не там, где создаются новые пользователи, поэтому требуется перейти в другое место.

Щелкните значок "Пуск".

Щелкните правой кнопкой мыши пункт **Компьютер** и выберите пункт **Управление**.

Выберите пункт **Пользователи**.

Дважды щелкните пункт **Администратор**.

Убедитесь, что установлен флажок **Отключить учетную запись** (если этот флажок уже установлен, все готово).

Закройте диалоговые окна.

Практическое занятие «Настройка внутренней политики безопасности Windows»

Цель занятия: Настройка внутренней политики безопасности Windows

Содержание работы:

1. Чтобы открыть локальную политику безопасности, на экране **Пуск** введите **secpol.msc**.
2. Дерево консоли для Параметры Windows\Параметры безопасности Policy\Windows локального компьютера
3. В разделе **Параметры безопасности** в дереве консоли, выполните одно из следующих действий:
 - Щелкните **политики учетных записей** изменение **Политика паролей** или **политики блокировки учетных записей**.
 - Щелкните **Локальные политики** изменение **политики аудита Назначение прав пользователя** или **Параметры безопасности**.
4. При нахождении параметр политики в области сведений дважды щелкните политику безопасности, которую требуется изменить.
5. Изменение параметров политики безопасности и нажмите кнопку **ОК**.

Настройка параметров для компьютера, присоединенного к домену

Следующая процедура описывает способы настройки параметра политики безопасности для объекта групповой политики, при работе на рабочей станции или сервере, присоединенных к домену.

Необходимо иметь соответствующие разрешения для установки и использования консоли управления (MMC) и обновить объект групповой политики (GPO) на контроллере домена для выполнения этих процедур.

1. Откройте консоль MMC и добавьте редактор объектов групповой политики, на экране **Пуск** введите **mmc.msc**.
2. На **файл** меню консоли MMC щелкните **Добавить или удалить оснастку** и нажмите кнопку **Добавить**.
3. В **Добавить изолированную оснастку** дважды щелкните **редактора объектов групповой политики**.
4. В **Выбор объекта групповой политики** щелкните **Обзор** перейдите к объекту групповой политики, вы хотите изменить и нажмите кнопку **Готово**.

5. Щелкните **Заккрыть**, а затем нажмите кнопку **ОК**. Эта процедура добавит оснастку в консоль MMC.
6. В дереве консоли найдите *Объект_групповой_политики [ИмяКомпьютера]* политики, щелкните **Конфигурация компьютера** щелкните **Параметры Windows** и нажмите кнопку **Параметры безопасности**.
7. Выполните одно из следующих действий:
 - Щелкните **политики учетных записей** изменение **Политика паролей** или **политики блокировки учетных записей**.
 - Щелкните **Локальные политики** изменение **политики аудита Назначение прав пользователя** или **Параметры безопасности**.
 - Щелкните **журнала событий** Чтобы изменить параметры журнала событий.
8. В области сведений дважды щелкните параметр политики безопасности, который требуется изменить.

Настройка параметров для контроллера домена

Следующая процедура описывает способы настройки параметра политики безопасности для только контроллер домена (контроллера домена).

1. Чтобы открыть политику безопасности контроллера домена, в дереве консоли, найдите *Объект_групповой_политики [ИмяКомпьютера]* политики, щелкните **Конфигурация компьютера** щелкните **Параметры Windows** и нажмите кнопку **Параметры безопасности**.
2. Выполните одно из следующих действий:
 - Дважды щелкните **политики учетных записей** изменение **Политика паролей**, **политики блокировки учетных записей** или **Политика Kerberos**.
 - Щелкните **Локальные политики** изменение **политики аудита Назначение прав пользователя** или **Параметры безопасности**.
 - Щелкните **журнала событий** Чтобы изменить параметры журнала событий.
3. В области сведений дважды щелкните политику безопасности, которую требуется изменить
4. Изменение параметров политики безопасности и нажмите кнопку **ОК**.

Практическое занятие «Знакомство с операционной системой UNIX.»

Цели занятия: познакомиться с операционной системой Unix.

Содержание работы:

Основы работы с командами в консоли ОС UNIX

Основные понятия:

1. сеанс работы
2. виртуальные консоли
3. оболочка
4. рабочая среда
5. удаленная консоль

Используемые команды:

alias	вывод списка или создание нового алиаса для команды
date	вывод или установка системной даты и времени
df	вывод информации об использовании дискового пространства
env	вывод информации о среде, запуск приложений с измененными переменными среды
exit	завершение сеанса работы
hostname	вывод имени машины
man	вывод справки по команде
uname	вывод системной информации
whereis	вывод информации о расположении файла
which	вывод полного пути до файла вызываемого командой

UNIX — многозадачная многопользовательская операционная система. Система

может работать в режиме с графической оболочкой или без нее. В рамках курса изучения ОС UNIX все работы будут проводиться в системе, работающей в многопользовательском режиме с поддержкой сети без графической оболочки.

Системная консоль — это монитор и клавиатура, связанные непосредственно с системой. Для входа в систему под несколькими учетными записями, используя один монитор и одну клавиатуру, система обеспечивает доступ к виртуальным консолям, которые позволяют войти в систему под несколькими именами в одно время. Сеанс работы начинается со входа в систему, для чего пользователь должен ввести свое имя и пароль. Для завершения сеанса работы с системой вы можете воспользоваться командой `exit` или сочетанием клавиш `CTRL-D`.

Оболочка (`shell`, командный интерпретатор) — это программа, которая воспринимает введенные пользователем данные и транслирует это в системные команды. Оболочка запускается сразу после входа пользователя в систему. Используя язык обращения к оболочке можно создавать скрипты.

Рабочая среда — это множество переменных, к которым имеют доступ все выполняемые команды.

Файл `/etc/profile` содержит переменные среды на уровне всей системы. Файл `.bash_profile` содержит переменные среды пользователя. При входе в систему первыми скриптами, выполняемыми оболочкой, являются эти скрипты.

Задание

1. Войдите в систему, используя имя пользователя и пароль, выданные вам преподавателем.
2. Определите имя машины. Найдите две разные команды, которыми можно это сделать.
3. Определите архитектуру процессора машины, используемой вами.
4. Выведите на экран время и дату в формате «31/12/2099 23:59».
5. Выведите на экран полный список алиасов пользователя. Создайте алиас для команды `newdate`, чтобы она выводила дату в формате «31/12/2099 23:59».
6. Выведите на экран переменные среды. Объясните, что означают переменные `SHELL`, `USER`, `PATH`, `HOME`, `LANG`, `MAIL`, `PS1`.
7. Измените переменную `PS1`, что изменилось?
8. Запустите приложение `bash`, изменив переменную домашнего каталога пользователя. Проверьте, что изменения вступили в силу. Как это можно сделать?
9. Определите местоположение в системе файла вызываемого командой `bash`, используя команду `which`.

Контрольные вопросы:

1. Опишите процесс загрузки операционной системы.
2. Что такое уровни инициализации, и зачем они нужны?
3. В чем заключается процедура авторизации пользователя? Цель авторизации? Какие действия совершает система после того, как пользователь ввел пароль, и до того, как пользователь получает возможность передавать системе какие-то команды?
4. Опишите две реальные ситуации, когда вам может понадобиться получить дату и время в консоли UNIX.
5. Что такое среда пользователя? Опишите, как добавить новую переменную в среду, как изменить значение существующей переменной для одного пользователя и всех пользователей в системе.
6. Опишите одну реальную ситуацию, когда вам может понадобиться изменить переменную среды при запуске приложения.
7. Для чего используются группы пользователей?
8. Зачем нужны идентификаторы пользователей?
9. Объясните содержание и назначение каждого поля регистрационной записи.
10. В чем отличие в диалоге прямыми сообщениями и почтовыми?

Цель занятия: ознакомиться с основными возможностями и функциями терминала о работе с текстом и текстовыми файлами, а также со встроенным редактором.

Содержание работы:

1) Копируем в домашний каталог файл /home/labs/lab1.txt.

```
marina@ubuntu:~$ cp /home/labs/lab1.txt /home/marina/
2) Выведем первые 10 строк файла lab1.txt.
marina@ubuntu:~$ head -n 10 lab1.txt
```

```
3) Выведем последние 10 строк файла lab1.txt.
marina@ubuntu:~$ tail -n 10 lab1.txt
```

4) Отсортируйте строки файла, исключив повторяющиеся, и выведите в файл отчета (lab2.txt) строки с 205-й по 215-ю из полученного результата. (Выполнить одним конвейером, используя команды sort, head, tail).

```
marina@ubuntu:~$ sort -u lab1.txt | head -n 215 | tail -n 10 | sort -o lab2.txt
```

5) Скопируйте в домашний каталог файл /home/labs/text1.txt.

```
marina@ubuntu:~$ cp /home/labs/text.txt /home/marina/
```

Текст файла text.txt:

Текст файла text1.txt:

6) Добавьте в файл отчета все строки файла text.txt которых нет в файле text1.txt. Используйте команду comm, не забудьте отсортировать файлы перед тем, как подавать их на вход команды comm.

Практическое занятие «Сбор сведений о системе»

Цель занятия: знакомство с командами по работе с процессами и командами, предоставляющими сведения о системе.

Содержание работы:

Изучаемые команды: grep, bg,

fgijobs, kill, mkfifo, mount, nice, nohup, ps, pstree, ps, renice, top, uname, who

Сценарий: Сбор сведений о системе

В данном сценарии изучаются команды, предоставляющие сведения о системе.

Начальные условия: Командная строка после входа в систему.

1. Определить имя текущей UNIX-системы с помощью команды **uname -a**

```
2. user@desktop ~ $ uname -a
```

```
Linux thinkpad 2.6.13-gentoo-r4 #1 Mon Oct 17 12:09:36 MSD 2005
i686 Intel(R)...
```

Здесь выводится полная версия системы, дата компиляции ядра и тип процессора.

3. Вывести содержимое каталога /proc с помощью команды **ls /proc**:

```
4. user@desktop ~ $ ls /proc
5. 1/ 7/ 8113/ 8209/ 9073/ dri/ modules
6. 103/ 7060/ 8124/ 8211/ 9324/ driver/ mounts@
7. 106/ 7128/ 8125/ 8213/ 9325/ execdomainsmtrr
8. 185/ 7693/ 8131/ 8215/ 937/ fbnet/
9. 186/ 7770/ 8135/ 8319/ 9376/ fb1/ partitions
```

...

Признаком операционной системы Linux является наличие каталога /proc, которая содержит псевдофайлы, предоставляющие системную информацию и позволяющие настраивать ядро.

10. Вывести текущие пользовательские сеансы с помощью команды **who**:

```
11. user@desktop ~ $ who
12. user tty1  2005-10-24 13:17
13. userpts/0 2005-10-24 13:19
```

```
userpts/2 2005-10-24 23:59
```

14. Вывести список всех примонтированных устройств с помощью команды **mount**:

```
15. user@desktop ~ $ mount
16. /dev/hda6 on / type reiserfs (rw,noatime)
17. proc on /proc type proc (rw)
18. sysfs on /sys type sysfs (rw)
19. udev on /dev type tmpfs (rw,nosuid)
20. devpts on /dev/pts type devpts (rw)
```

```
/dev/hda1 on /mnt/win type ntfs (ro,uid=1000,gid=100,nls=utf8)
```

Первый столбец — имя устройства, второй — точка монтирования, далее следует тип файловой системы и опции монтирования.

21. Вывести загруженность примонтированных дисков с помощью команды **df -h**:

```
22. user@desktop ~ $ df -h
23. Файловая система  РазмИспДостИсп% смонтирована на
24. /dev/hda6      24G 21G 3,4G 86% /
25. udev          251M 180K 251M 1% /dev
26. /dev/hda1     9,4G 7,7G 1,7G 83% /mnt/win
```

```
none      251M 0 251M 0% /dev/shm
```

27. Вывести информацию о всех выполняющихся процессах с помощью команды **psaux**:

```
28. user@desktop ~ $ psaux
29. USER  PID %CPU %MEM VSZ RSS TTY  STAT START TIME
    COMMAND
30. root   1  0.0  0.0 1432 480 ?   S   13:16 0:01 init [3]
31. root   2  0.0  0.0   0   0 ?   SN  13:16 0:00 [ksoftirqd/0]
32. root   3  0.0  0.0   0   0 ?   S<  13:16 0:00 [events/0]
33. ..
34. user 8456  0.3  4.4 34932 22944 ?   S   13:24 2:15 emacs
35. user 22537 0.0  0.3 3720 1560 pts/3 Ss  22:37 0:00 /bin/bash
```

```
user 8839 0.0  0.1 2644 932 pts/3 R+  23:01 0:00 psaux
```

36. Оставить в выводе предыдущей команды только системные процессы с помощью команды **psaux | grep -v user**:

```
37. user@desktop ~ $ ps aux | grep -v user
38. USER  PID %CPU %MEM VSZ RSS TTY  STAT START TIME
    COMMAND
39. root   1  0.0  0.0 1432 480 ?   S   13:16 0:01 init [3]
40. root   2  0.0  0.0   0   0 ?   SN  13:16 0:00 [ksoftirqd/0]
41. root   3  0.0  0.0   0   0 ?   S<  13:16 0:00 [events/0]
42. ...
43. root 6989  0.0  0.1 1672 704 ?   Ss Oct25 0:00 /usr/sbin/syslog
44. root 7128  0.0  0.1 2020 716 ?   Ss Oct25 0:00 /usr/sbin/fcron
```

Часть процессов, показанные в квадратных скобках, — это потоки ядра, исполняющие специальные функции. Можно выделить демонов — в столбце терминала они имеют символ «?». Например, системный журнал (syslog) или планировщик задач (cron).

45. Вывести иерархию процессов с помощью команды **ps tree**:

```
46. user@desktop ~ $ ps tree
47. init—acpid
48.   |—5*[agetty]
49.   |—2*[artsd]
50.   |—cpufreqd
51.   |—dcopserver
52.   |—events/0
53.   |—fcron
54.   |—gconfd-2
55.   |—hcid
56.   |—kaccess
57.   |—kded—kded—kded
58.   |—kdeinit—artsd—artsd—artsd
59.   |   |—kio_file
60.   |   |—kio_imap4—kio_imap4
```

...

В вершине дерева стоит процесс **init**.

61. Рассмотреть поведение процессов интерактивно с помощью команды **top**.

Сценарий: Управление процессами с помощью сигналов

В данном сценарии изучаются сигналы, управляющие клавиши для передачи процессам сигналов, команды для управления процессами.

Начальные условия: Командная строка после входа в систему.

1. Запустите команду **yes**, производящую бесконечный вывод символа **y** на экран. Прервите её нажатием **Ctrl-C**.

```
2. user@desktop ~ $ yes
3. y
4. y
5. y
6. y
7. ...
8. y
9.
```

user@desktop ~ \$

При этом запущенному процессу был отправлен сигнал **SIGTERM** — завершения программы.

10. Запустим сбор информации обо всех файлах системы с помощью команды **find / > files.txt**.

```
user@desktop ~ $ find / > files.txt
```

11. Найдём идентификатор запущенного только что процесса с помощью команды **ps aux | grep find**, запущенной в другом терминале.

```
12. user@desktop ~ $ ps aux | grep find
13. user 8178 9.7 0.1 2844 1020 pts/1 R+ 23:45 0:00 find /
```

```
user 8230 0.0 0.1 2708 736 pts/2 R+ 23:45 0:00 grepfind
```

Первая строка содержит необходимый PID.

14. Отправим сигнал завершения этому процессу с помощью команды **kill 8178**, указав в качестве параметра идентификатор процесса.

```
user@desktop ~ $ kill 8178
```

Можно убедиться, что на терминале с запущенной командой появились строка «Terminated» и приглашение.

15. Если попытаться завершить системный процесс, например командой **kill 1**, появится сообщение об ошибке доступа:

```
16. user@desktop ~ $ kill 1
```

```
bash: kill: (1) - Operationnotpermitted
```

Отправление сигналов системным процессам может производить только суперпользователь.

17. Альтернативным способом отправления сигналов процессам — по имени процесса, а не по PID — является команда **killall**. Выполнив команду **killallbash**, мы завершим все командные оболочки, а тем самым и сеансы пользователей.

```
user@desktop ~ $ killallbash
```

18. До этого мы отправляли только сигнал завершения процесса. Он может перехватываться и игнорироваться программами. Неперехватываемым является сигнал **SIGKILL**, который может быть отправлен, например, следующей командой: **killall -SIGKILL find**

```
user@desktop ~ $ killall -SIGKILL find
```

Сценарий: Выполнение задач в фоновом режиме

В данном сценарии изучается работа с заданиями командной оболочки, запуск заданий в фоновом режиме.

Начальные условия: Командная строка после входа в систему.

1. Запустим длительную команду, например **find / > files.txt** и приостановим её выполнение с помощью нажатия **Ctrl-Z**. При этом процессу посылается сигнал **SIGSTOP**.

```
2. user@desktop ~ $ find / > files.txt
```

```
3.
```

```
4. [1]+ Stoppedfind / > files.txt
```

```
user@desktop ~ $
```

Команда приостановлена и запомнена как задача 1 (номер в квадратных скобках).

5. Текущий список запущенных задач командной оболочки можно посмотреть командой **jobs**.

```
6. user@desktop ~ $ jobs
```

```
7. [1]+ Stoppedfind / > files.txt
```

8. Возобновить исполнение задания можно командой **fg 1**, аргументом которой является номер задания.

```
9. user@desktop ~ $ fg 1
```

```
10. find / > files.txt
```

11. Если еще раз приостановить процесс, можно запустить задачу в фоновом режиме: **bg 1**.

```
12. user@desktop ~ $ fg 1
```



```
13. [1]+ find / > files.txt &
```

```
user@desktop ~ $
```

Тогда можно будет продолжать работу в командной строке.

14. Команды можно сразу запускать в фоновом режиме. При этом необходимо добавить символ «&» (амперсанд) в конец строки команды: **find / -name "*.xml" >xml-list&**

```
15. user@desktop ~ $ find / -name "*.xml" > xml-list &
```

```
16. [1] 9413
```

```
user@desktop ~ $
```

При этом командная оболочка выводит номер задания и PID созданного процесса.

Сценарий: Запуск демонов

В данном сценарии рассматриваются демоны как процессы, не связанные ни с одним терминалом.

Начальные условия: Командная строка после входа в систему.

1. Если запустить команду в фоновом режиме и выйти из командной оболочки: например, **find / -name "*.html" -execgrep -Hn "linuxloader" \{\} \; &** и **exit**, то запущенная команда завершится по сигналу SIGHUP.

```
2. user@desktop ~ $ find / -name "*.html" -exec grep -Hn "linux loader"
```

```
3. \{\} \; &
```

```
4. [1] 9555
```

```
user@desktop ~ $ exit
```

5. Для того, чтобы программы не получали сигнал SIGHUP, используется специальная команда **nohup**: **nohupfind / -name "*.txt" -execgrep -Hn "linuxloader" \{\} \; &**

```
6. user@desktop ~ $ nohup find / -name "*.html" -exec grep -Hn "linux  
loader" \{\} >> result\; &
```

```
7. [1] 9783
```

```
user@desktop ~ $ exit
```

Эта программа завершится корректно после окончания поиска.

Сценарий: Изменение приоритетов выполняющихся программ

В данном сценарии изучается механизм приоритетов UNIX и команды для изменения приоритетов запускаемых процессов.

Начальные условия: Командная строка после входа в систему.

1. Каждый процесс в системе имеет свой уровень приоритета (в UNIX он называется «nice»), который можно увидеть с помощью команды **ps -l** — в столбце «NI».

```
2. user@desktop ~ $ ps -l
```

```
3. F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY TIME CMD
```

```
4. 0 S 1000 17223 17222 0 75 0 - 994 waitpts/1 00:00:00 bash
```

```
0 R 1000 17485 17223 0 75 0 - 612 - pts/1 00:00:00 ps
```

По умолчанию, приоритет процессов простого пользователя равен нулю.

5. Для запуска процесса с заданным приоритетом воспользуемся командой **nice**. Например, запустим создание архива с пониженным приоритетом: **nice -n 10 tar -cjf libraries.tar.bz2 /usr/lib/**

```
user@desktop ~ $ nice -n 10 tar -cjf libraries.tar.bz2 /usr/lib/
```

6. Чтобы изменить приоритет уже запущенной программы используется команда **renice**. Например, для понижения приоритета заранее запущенного процесса архивации **tar -cjf libraries.tar.bz2 /usr/lib&** воспользуемся командой **renice +10 -p 3442**:

```
7. user@desktop ~ $ tar -cjf libraries.tar.bz2 /usr/lib &  
8. [1] 3442
```

```
user@desktop ~ $ renice +10 -p 3442
```

Простые пользователи не могут повышать приоритет процессов, только понижать.

Задания для самоподготовки

1. Сравните время создания корневого каталога с текущим временем системы.
2. Получите список всех типов файловых систем, примонтированных в настоящий момент. Какой тип имеет корневая файловая система?
3. Найдите три процесса, использующих самый большой объем памяти. Какой процент памяти они потребляют от общего объема?
4. Получите иерархию родительских процессов для текущей командной оболочки.
5. Найдите, в каком файле в каталоге `/proc` хранится информация о системной памяти.
6. С помощью сигнала остановки процесса и команды **bg** запустите три параллельных процесса архивации каталога `/etc`, `/home` и `/tmp`.

Практическое занятие Работа с файлами в UNIX. Работа с каталогами в UNIX»

Цель занятия: знакомство с командной строкой UNIX, командами по работе с файловой системой и командами обработки текстовых файлов.

Содержание работы:

Изучаемые

команды: cat, cd, cp, cut, echo, find, grep, head, ln, ls, mkdir, mv, paste, pwd, sort, tail, tar, touch, uniq, wc

Сценарий: Знакомство с каталогами UNIX

В этом сценарии будет освещено знакомство с командами UNIX, изучение файловой системы и базовых каталогов, создание рабочей среды в домашнем каталоге пользователя для выполнения всех последующих команд.

Начальные условия: Командная строка после входа в систему.

1. Получить имя текущего каталога с помощью команды **pwd**:

```
2. user@desktop ~ $ pwd
```

```
/home/user/
```

3. Перейти в корневой каталог с помощью команды **cd /**:

```
4. user@desktop ~ $ cd /
```

```
5. user@desktop / $ pwd
```

```
/
```

Отметить, как изменилась строка приглашения.

6. Посмотреть содержимое корневого каталога с помощью команды **ls**:

```
7. user@desktop / $ ls
```

```
8. bin dev home mnt proc sbin tmp var
```

```
boot etc lib opt root sys usr
```

9. Сравнить с использованием «раскарашенного» вывода команды **ls -F --color**:

```
10. user@desktop / $ ls -F --color
```

```
11. bin/ dev/ home/ mnt/ proc/ sbin/ tmp/ var/
```

```
boot/ etc/ lib/ opt/ root/ sys/ usr/
```

Директории отмечаются синим цветом и знаком «/» после имени.

12. Посмотреть содержимое домашнего каталога с помощью команды **ls -F --color ~**:

```
13. user@desktop / $ ls -F --color ~
```

```
user@desktop / $
```

Возможно, домашний каталог пуст.

14. Вернуться в домашний каталог с помощью **cd** без параметров:

```
15. user@desktop / $ cd
```

```
user@desktop ~ $
```

16. Создать директорию **test** с помощью команды **mkdir test**:

```
user@desktop ~ $ mkdir test
```

17. Посмотреть обновлённое содержимое домашнего каталога **ls -F --color**:

```
18. user@desktop ~ $ ls -F --color
```

```
test/
```

19. Создать поддиректорию **subtest** в директории **test** командой **mkdir test/subtest**:

```
user@desktop ~ $ mkdir test/subtest
```

20. Посмотреть содержимое домашнего каталога и его подкаталогов с помощью ключа рекурсивного просмотра **-R** в команде **ls -F --color -R**:

```
21. user@desktop ~ $ ls -F --color -R
```

```
22. .:
```

```
23. test/
```

```
24. test:
```

```
25. subtest/
```

```
26.
```

```
test/subtest:
```

Сценарий: Изучение типов файлов в UNIX

В этом сценарии рассматриваются типы файлов в UNIX: простые файлы, директории, ссылки. Изучаются команды создания и копирования файлов. Показывается разница между жесткими и символическими ссылками, между копированием и переносом файла.
Начальные условия: Командная строка. Директория **test** после предыдущего сценария.

1. Создать пустой файл с помощью команды **touch test/first.txt**:

```
2. user@desktop ~ $ touch test/first.txt
```

```
3. user@desktop ~ $ ls -F --color test/
```

```
first.txt subtest/
```

4. Добавить строку текста в конец файла с помощью команды **echo "Hello, world" >> test/first.txt** и перенаправления вывода:

```
user@desktop ~ $ echo "Hello, world" >> test/first.txt
```

5. Посмотреть содержимое файла с помощью команды **cat test/first.txt**:

```
6. user@desktop ~ $ cat test/first.txt
7. Hello, world
```

```
user@desktop ~ $
```

8. Посмотреть расширенную информацию о каталоге, используя ключ -
l команды **ls**. **ls -F --color -l test**:

```
9. user@desktop ~ $ ls -F --color -l test
10. итого 4
11. -rw-r--r-- 1 user users 13 Окт 15 20:48 first.txt
```

```
drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
```

12. Добавить псевдоним командной оболочки, чтобы сократить размер команды, с
помощью команды **alias ls='ls -F --color -l'**:

```
13. user@desktop ~ $ alias ls='ls -F --color -l'
14. user@desktop ~ $ ls test
15. итого 4
16. -rw-r--r-- 1 user users 13 Окт 15 20:48 first.txt
```

```
drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
```

17. Посмотреть глубже на директорию test, используя ключи -a и -i. **ls -a -i test**

```
18. user@desktop ~ $ ls -a -i test
19. итого 7
20. drwxr-xr-x 3 user users 104 Окт 15 20:49 ./
21. drwx----- 67 user users 3352 Окт 15 20:42 ../
22. -rw-r--r-- 1 user users 13 Окт 15 20:48 first.txt
```

```
drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
```

Первый столбец – номера индексных узлов файловой системы. Третий столбец –
число жёстких ссылок файла.

23. Перейти в каталог test с помощью команды **cd test**

```
user@desktop ~ $ cd test
```

24. Скопировать файл с помощью команды **cp first.txt copy1.txt**:

```
25. user@desktop test $ cp first.txt copy1.txt
26. user@desktop test $ ls -a -i
27. итого 11
28. 5522107 drwxr-xr-x 3 user users 136 Окт 15 20:54 ./
29. 384269 drwx----- 67 user users 3352 Окт 15 20:42 ../
30. 5938127 -rw-r--r-- 1 user users 13 Окт 15 20:54 copy1.txt
31. 5937954 -rw-r--r-- 1 user users 13 Окт 15 20:48 first.txt
```

```
6292625 drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
```

Новый файл имеет свой собственный индексный узел.

32. Переименовать файл с помощью команды **mv first.txt orig.txt**. **mv first.txt orig.txt**:

```
33. user@desktop test $ mv first.txt orig.txt
34. user@desktop test $ ls -a -i
35. итого 11
36. 5522107 drwxr-xr-x 3 user users 128 Окт 15 20:59 ./
37. 384269 drwx----- 67 user users 3352 Окт 15 20:42 ../
38. 5938127 -rw-r--r-- 1 user users 13 Окт 15 20:54 copy1.txt
```

```
39. 5937954 -rw-r--r-- 1 user users 13 Окт 15 20:48 orig.txt
```

```
6292625 drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
```

Отметим, что изменилось только имя файла, все остальные атрибуты остались прежними.

40. Создать жёсткую ссылку командой **ln orig.txt copy2.txt**:

```
41. user@desktop test $ ln orig.txt copy2.txt
42. user@desktop test $ ls -a -i
43. итого 15
44. 5522107 drwxr-xr-x 3 user users 160 Окт 15 21:00 ./
45. 384269 drwx----- 67 user users 3352 Окт 15 20:42 ../
46. 5938127 -rw-r--r-- 1 user users 13 Окт 15 20:54 copy1.txt
47. 5937954 -rw-r--r-- 2 user users 13 Окт 15 20:48 copy2.txt
48. 5937954 -rw-r--r-- 2 user users 13 Окт 15 20:48 orig.txt
```

```
6292625 drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
```

Добавилась ещё одна ссылка на тот же файл, число ссылок увеличилось на 1.

49. Создать символическую ссылку командой **ln -s orig.txt orig.lnk**:

```
50. user@desktop test $ ln -s orig.txt orig.lnk
51. user@desktop test $ ls -a -i
52. итого 15
53. 5522107 drwxr-xr-x 3 user users 184 Окт 15 21:18 ./
54. 384269 drwx----- 67 user users 3352 Окт 15 21:02 ../
55. 5938127 -rw-r--r-- 1 user users 13 Окт 15 20:54 copy1.txt
56. 5937954 -rw-r--r-- 2 user users 13 Окт 15 20:48 copy2.txt
57. 5938189 lrwxrwxrwx 1 user users 8 Окт 15 21:18 orig.lnk -> orig.txt
58. 5937954 -rw-r--r-- 2 user users 13 Окт 15 20:48 orig.txt
```

```
6292625 drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
```

Новый файл имеет новый индексный узел и размер, равный имени файла orig.txt.

59. Сравним содержимое файлов при обращении к ним по имени:

```
60. user@desktop test $ cat orig.txt
61. Hello, world
62. user@desktop test $ cat copy2.txt
63. Hello, world
64. user@desktop test $ cat orig.lnk
```

```
Hello, world
```

К одним и тем же данным можно обратиться через жесткую или символическую ссылку.

Сценарий: Поиск системных журналов

В этом сценарии изучается команда поиска файлов и директорий.

Начальные условия: Командная строка, директория test после прошлого сценария.

1. Перейти в каталог test с помощью команды **cd test**:

```
user@desktop ~ $ cd test
```

2. Вывести список всех файлов и директорий в текущей директории, включая содержимое поддиректорий с помощью команды **find**:

```
3. user@desktop ~ $ find
4. .
5. ./orig.lnk
```

```
6. ./orig.txt
7. ./subtest
8. ./copy1.txt
```

```
./copy2.txt
```

9. Найти все файлы и директории в текущей директории и ее поддиректориях, которые начинаются на «о» с помощью команды **find -name "o*"**:

```
10. user@desktop ~ $ find -name "o*"
11. ./orig.lnk
```

```
./orig.txt
```

12. Найти все файлы и директории в директории /etc и ее поддиректориях, которые начинаются на «о» с помощью команды **find /etc -name "o*"**:

```
13. user@desktop ~ $ find /etc -name "o*"
14. /etc/apm/other.d
15. /etc/joe/syntax/ocaml.jsf
16. /etc/opt
17. /etc/ppp/options-pptp
18. /etc/ppp/options-pppoe
19. /etc/ppp/options
20. /etc/ssl/openssl.cnf
21. /etc/dpkg/origins
22. /etc/pam.d/other
```

```
/etc/texmf/tex/plain/config/omega.ini
```

23. Найти все обычные файлы в директории /etc и ее поддиректориях, которые начинаются на «о» с помощью команды **find /etc -name "o*" -a -type f**:

```
24. user@desktop ~ $ find /etc -name "o*" -a -type f
25. /etc/joe/syntax/ocaml.jsf
26. /etc/ppp/options-pptp
27. /etc/ppp/options-pppoe
28. /etc/ppp/options
29. /etc/ssl/openssl.cnf
30. /etc/pam.d/other
```

```
/etc/texmf/tex/plain/config/omega.ini
```

31. Найти все обычные файлы в директории /var/ и ее поддиректориях, заканчивающиеся на «log»:

```
32. user@desktop ~ $ find /var -name "*log" -a -type f
33. find: /var/lib/mysql: Permission denied
34. find: /var/lib/samba/private: Permission denied
35. ...
36. /var/log/emerge.log
37. /var/log/lastlog
38. /var/log/faillog
```

```
/var/log/Xorg.0.log
```

Список найденных файлов может быть слишком большим и для его просмотра можно воспользоваться «прокруткой» терминала с помощью клавиш **Shift-PgUp** и **Shift-PgDn**

39. Исключить вывод сообщений об ошибках (например, «Permission denied») с помощью перенаправления вывода ошибок в специальное устройство /dev/null. Для этого ввести команду **find /var -name "*"log" -a -type f 2>/dev/null**

```
40. user@desktop ~ $ find /var -name "*"log" -a -type f 2>/dev/null
41. /var/log/emerge.log
42. /var/log/lastlog
43. /var/log/faillog
```

```
/var/log/Xorg.0.log
```

44. Создать директорию logs с помощью команды **mkdir test/logs**:

```
user@desktop ~ $ mkdir test/logs
```

45. Скопировать найденные файлы в локальную директорию с помощью параметра -exec команды find. Для этого выполнить: **find /var -name "*"log" -a -type f -exec cp {} test/logs/ 2>/dev/null**:

```
46. user@desktop ~ $ find /var -name "*"log" -a -type f -exec cp {}
    test/logs/ 2>/dev/null
47. user@desktop ~ $ ls test/logs
48. итого 789
49. -rw-r----- 1 user users 601033 Окт 16 18:37 emerge.log
50. -rw-r--r-- 1 user users 292292 Окт 16 18:37 lastlog
```

```
-rw-r--r-- 1 user users 37383 Окт 16 18:37 Xorg.0.log
```

Были скопированы все файлы, права на чтение которых у нас есть.

Сценарий: Архивирование и деархивирование файлов и директорий

В этом сценарии изучается команда архивирования файлов и директорий. Командная строка, директория test после прошлого сценария.

Начальные условия:

1. Перейти в домашнюю директорию с помощью команды **cd**:

```
2. user@desktop test $ cd
```

```
user@desktop ~ $
```

3. Создать архив с именем test.tar.gz с применением сжатия, содержащий директорию test с помощью команды **tar -czf test.tar.gz test**:

```
4. user@desktop ~ $ tar -czf test.tar.gz test
5. user@desktop ~ $ ls
6. drwxr-xr-x 4 user users 208 Окт 16 18:36 test/
```

```
-rw-r--r-- 1 user users 79173 Окт 16 18:49 test.tar.gz
```

7. Создать новую директорию для содержимого архива командой **mkdir test2**:

```
user@desktop ~ $ mkdir test2
```

8. Перейти в новую директорию с помощью команды **cd test2**:

```
user@desktop ~ $ cd test2
```

9. Развернуть содержимое архива в текущую директорию с помощью команды **tar -xzf ../test.tar.gz**:

```
user@desktop test2 $ tar -xzf ../test.tar.gz
```

10. Убедиться, что содержимое распакованного архива совпадает с оригинальной директорией. Ввести команду **ls -l -F**:


```

11. user@desktop test2 $ ls -l -F
12. итого 0
13. drwxr-xr-x 4 user users 208 Окт 16 18:36 test/
14. user@desktop test2 $ ls -l -F test
15. итого 12
16. -rw-r--r-- 1 user users 13 Окт 15 20:54 copy1.txt
17. -rw-r--r-- 2 user users 13 Окт 15 20:48 copy2.txt
18. drwxr-xr-x 2 user users 136 Окт 16 18:37 logs/
19. lrwxrwxrwx 1 user users 8 Окт 16 18:54 orig.lnk -> orig.txt
20. -rw-r--r-- 2 user users 13 Окт 15 20:48 orig.txt

drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/

```

Сценарий: Создание новых текстовых файлов

В этом сценарии изучаются команды для создания и модифицирования текстовых файлов. Командная строка.

Начальные условия:

1. Ввести команду **echo "One line"**:

```
2. user@desktop test $ echo "One line"
```

```
One line
```

Данная команда принимает строку в виде аргумента и выводит её на стандартный вывод.

3. Ввести ту же команду, но перенаправить вывод в файл `second.txt` с помощью «>». Ввести команду **echo "One line" > second.txt**:

```
user@desktop test $ echo "One line" > second.txt
```

4. Добавить в конец файла `second.txt` строку «A line» с помощью другого перенаправления **echo "A line" >> second.txt**:

```
user@desktop test $ echo "A line" >> second.txt
```

5. Вывести содержимое файла с помощью команды **cat second.txt**:

```
6. user@desktop test $ cat second.txt
```

```
7. One line
```

```
A line
```

8. С помощью команды `cat` можно создавать многострочные файлы —, если переопределить вывод программы в файл и вводить текст до нажатия **Ctrl-D** (конец ввода). Ввести команду **cat >multiline.txt** и набрать текст

```
9. user@desktop test $ cat >multiline.txt
```

```
10. Simple text:
```

```
11. blah-blah-blah
```

```
12. 1 2 3 4 5 6 7 8 9 0
```

```
13.
```

```
14. bye!
```

```
15.
```

```
16. (Ctrl+D)
```

```
user@desktop test $
```

17. Убедиться, что содержимое файла совпадает с введённым текстом, включая все переводы строки. Для этого ввести команду **cat multiline.txt**

```
18. user@desktop test $ cat multiline.txt
```



```
19. Simple text:
20. blah-blah-blah
21. 1 2 3 4 5 6 7 8 9 0
22.
23. bye!
24.
```

25. Основным назначением команды `cat` является объединение файлов, имена которых передаются как аргументы командной строки. Объединить файлы с помощью команды **`cat orig.txt second.txt multiline.txt > big.txt`**:

```
user@desktop test $ cat orig.txt second.txt multiline.txt > big.txt
```

26. Убедиться, что новый файл содержит строки из перечисленных файлов с помощью команды **`cat big.txt`**:

```
27. user@desktop test $ cat big.txt
28. Hello, world
29. One line
30. A line
31. Simple text:
32. blah-blah-blah
33. 1 2 3 4 5 6 7 8 9 0
34.
35. bye!
36.
```

Сценарий: Разрезание и склеивание файлов

В данном сценарии изучается, как выделять информацию, разбитую на части с помощью специального символа-разделителя (например, пробела или символа `:`), из файлов и создавать новые структурированные файлы.

Начальные условия: Командная строка.

1. Посмотреть содержимое файла `/etc/passwd`, в котором содержится информация о пользователях системы, с помощью команды **`cat /etc/passwd`**:

```
2. user@desktop test $ cat /etc/passwd
3. root:x:0:0:root:/root:/bin/bash
4. bin:x:1:1:bin:/bin:/bin/false
5. daemon:x:2:2:daemon:/sbin:/bin/false
6. adm:x:3:4:adm:/var/adm:/bin/false
7. lp:x:4:7:lp:/var/spool/lpd:/bin/false
8. sync:x:5:0:sync:/sbin:/bin/sync
9. shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
```

...

Каждая строка хранит запись об отдельном пользователе. Запись состоит из полей, разделённых символом `«:»`.

10. Выделить первый столбец в каждой записи, установив разделитель в `«:»`, с помощью команды **`cat /etc/passwd | cut -f1 -d:`**:

```
11. user@desktop test $ cat /etc/passwd | cut -f1 -d:
12. root
13. bin
14. daemon
15. adm
16. lp
17. sync
```

18. shutdown

...

19. Отсортировать список пользователей по алфавиту с помощью команды **cat /etc/passwd | cut -f1 -d: | sort**:

```
20. user@desktop test $ cat /etc/passwd | cut -f1 -d: | sort
21. adm
22. alias
23. apache
24. at
25. bin
26. cron
27. cyrus
```

...

28. Выделить седьмой столбец в каждой записи, соответствующий командной оболочке пользователя, с помощью команды **cat /etc/passwd | cut -f7 -d:**

```
29. user@desktop test $ cat /etc/passwd | cut -f7 -d:
30. /bin/bash
31. /bin/false
32. /bin/false
33. /bin/false
34. /bin/false
35. /bin/sync
```

...

Список содержит много повторяющихся строк.

36. Удалить повторяющиеся строки командой **cat /etc/passwd | cut -f7 -d: | uniq**:

```
37. user@desktop test $ cat /etc/passwd | cut -f7 -d: | uniq
38. /bin/bash
39. /bin/false
40. /bin/sync
41. /sbin/shutdown
42. /sbin/halt
43. /bin/false
```

...

Некоторые повторяющиеся строки остались, так как сравниваются только последовательные строки.

44. Добиться полной уникальности, предварительно отсортировав строки с помощью команды **sort**. Ввести команду **cat /etc/passwd | cut -f7 -d: | sort | uniq**:

```
45. user@desktop test $ cat /etc/passwd | cut -f7 -d: | sort | uniq
46. /bin/bash
47. /bin/false
48. /bin/sync
49. /dev/null
50. /sbin/halt
```

/sbin/shutdown

51. Выделить первый и седьмой столбец файла /etc/passwd в отдельные файлы. Для этого ввести команды **cat /etc/passwd | cut -f1 -d: > users** и **cat /etc/passwd | cut -f7 -d: > shells**:

```
52. user@desktop test $ cat /etc/passwd | cut -f1 -d: > users
```

```
user@desktop test $ cat /etc/passwd | cut -f7 -d: > shells
```

53. Объединить результаты с помощью команды **paste users shells**:

```
54. user@desktop test $ paste users shells
```

```
55. root /bin/bash
```

```
56. bin /bin/false
```

```
57. daemon /bin/false
```

```
58. adm /bin/false
```

```
59. lp /bin/false
```

```
60. sync /bin/sync
```

```
...
```

Сценарий: Быстрый анализ текстов

В данном сценарии изучаются команды, которые позволяют провести первичный анализ файла: с чего он начинается, как заканчивается и каков его размер.

Начальные условия: Командная строка.

1. Получить число строк в файле /etc/passwd с помощью команды **wc -l /etc/passwd** :

```
2. user@desktop test $ wc -l /etc/passwd
```

```
44 /etc/passwd
```

3. Или другим способом: посчитав число слов в созданном ранее файле users с помощью команды **cat users | wc -w**:

```
4. user@desktop test $ cat users | wc -w
```

```
44
```

5. Посмотреть первые несколько строк длинного файла с помощью команды **head logs/Xorg.0.log**:

```
6. user@desktop test $ head logs/Xorg.0.log
```

```
7. X Window System Version 6.8.2
```

```
8. Release Date: 9 February 2005
```

```
9. X Protocol Version 11, Revision 0, Release 6.8.2
```

```
10. Build Operating System: Linux 2.6.13-gentoo i686 [ELF]
```

```
11. Current Operating System: Linux thinkpad 2.6.13-gentoo #1 Wed Sep 7 01:03:14 MSD 2005 i686
```

```
12. Build Date: 08 October 2005
```

```
13. Before reporting problems, check http://wiki.X.Org
```

```
14. to make sure that you have the latest version.
```

```
Module Loader present
```

15. Аналогично, посмотреть последние несколько строк с помощью команды **tail logs/Xorg.0.log**:

```
16. user@desktop test $ tail logs/Xorg.0.log
```

```
17. (II) RADEON(0): [RESUME] Attempting to re-init Radeon hardware.
```

```
18. (II) RADEON(0): [agp] Mode 0x1f000201 [AGP 0x8086/0x3340; Card 0x1002/0x4c66]
```

```
19. (II) Mouse1: ps2EnableDataReporting: succeeded
```

```

20. SetClientVersion: 0 8
21. SetGrabKeysState - disabled
22. SetGrabKeysState - enabled
23. (II) 3rd Button detected: disabling emulate3Button
24. SetClientVersion: 0 8
25. SetGrabKeysState - disabled

```

```
SetGrabKeysState - enabled
```

Это бывает полезно при просмотре системных журналов – в них самые последние события содержатся в конце файлов.

Сценарий: Поиск строк по регулярным выражениям

В данном сценарии рассматривается практическое использование регулярных выражений для поиска по текстовым файлам.

Начальные условия: Командная строка. Директория test/log с найденными системными журналами.

1. Среди найденных системных журналов посмотреть файл журнала загрузки графической системы XFree86.0.log с помощью команды **less** **test/log/XFree86.0.log**:

```
user@desktop ~ $ less test/log/XFree86.0.log
```

Выйти из просмотрщика можно нажатием клавиши **q**.

2. Найти все предупреждения в журнале: строка, начинающаяся с «(WW)». Для этого ввести команду **grep** **"^(WW)" test/log/XFree86.0.log**:

```

3. user@desktop ~ $ grep "^(WW)" test/log/XFree86.0.log
4. (WW) The directory "/usr/share/fonts/freefonts/" does not exist.
5. (WW) The directory "/usr/share/fonts/unifont/" does not exist.
6. (WW) The directory "/usr/share/fonts/artwiz/" does not exist.

```

...

7. Получить все строки, содержащие следующий шаблон: последовательность из 1 или более цифры или буквы, затем символ собаки, затем еще одна последовательность 1 или более цифры или буквы или точки, затем символ точки, затем от двух до четырех букв (имя домена первого уровня). Также служебный вывод ошибок удаляется (перенаправляется в нулевое устройство). Для этого ввести команду **grep** **"[a-zA-z0-9]\+@[a-zA-z0-9\.\+\.][a-z]\{2,4\}" /etc/* 2>/dev/null**:

```

8. user@desktop ~ $ grep "[a-zA-z0-9]\+@[a-zA-z0-9\.\+\.][a-z]\{2,4\}" /etc/* 2>/dev/null
9. /etc/devfsd.conf:# Richard Gooch <rgooch@atnf.csiro.au> 3-JUL-2000
10. /etc/inittab:# Author: Miquel van Smoorenburg, <miquels@cistron.nl>
11. /etc/inittab:# Modified by: Patrick J. Volkerding, <volkerdi@ftp.cdrom.com>

```

...

При работе с несколькими файлами **grep** по умолчанию использует вывод имени файла перед найденной строкой.

12. Избавиться от имени файла в начале строки с помощью ключа **-h**. Ввести команду **grep -h** **"[a-zA-z0-9]\+@[a-zA-z0-9\.\+\.][a-z]\{2,4\}" /etc/* 2>/dev/null**:

```

13. user@desktop ~ $ grep -h "[a-zA-z0-9]\+@[a-zA-z0-9\.\+\.][a-z]\{2,4\}" /etc/* 2>/dev/null
14. # Richard Gooch <rgooch@atnf.csiro.au> 3-JUL-2000

```

```
15.          # Author: Miquel van Smoorenburg,
    <miquels@cistron.nl>
16.          # Modified by: Patrick J. Volkerding,
    <volkerdi@ftp.cdrom.com>
```

...

17. Для вывода только найденной подстроки используется ключ **-o**. Ввести команду **grep -ho "[a-zA-Z0-9]\+@[a-zA-Z0-9\.]\\+\.[a-z]\{2,4\}" /etc/* 2>/dev/null:**

```
18. user@desktop ~ $ grep -ho "[a-zA-Z0-9]\+@[a-zA-Z0-9\.]\\+\.[a-
    z]\{2,4\}" /etc/* 2>/dev/null
19. rgooch@atnf.csiro.au
20. miquels@cistron.nl
21. volkerdi@ftp.cdrom.com
```

...

22. Получить все файлы директории **/etc/** с помощью команды **find /etc -type f 2>/dev/null**, игнорируя ошибки:

```
23. user@desktop ~ $ find /etc -type f 2>/dev/null
24. /etc/X11/xorg.conf.example
25. /etc/X11/Sessions/kde-3.4
26. /etc/X11/Sessions/Xsession
27. /etc/X11/chooser.sh
```

...

28. Для каждого из найденных файлов произвести поиск подстроки «nameserver». Для этого ввести команду **find /etc -type f -exec grep "nameserver" {} \; 2>/dev/null:**

```
29. user@desktop ~ $ find /etc -type f -exec grep "nameserver" {} \;
    2>/dev/null
30. # merge the new nameservers with the other options from the old ...
31. grep --invert-match '^nameserver[[:space:]]' $REALRESOLVCONF
32. nameserver 10.22.10.20
```

...

33. Для большего удобства добавить вывод имени файла (ключ **-H**) и номера строки (ключ **-n**), на которой было найдено совпадение, с помощью команды **find /etc -type f -exec grep -Hn "nameserver" {} \; 2>/dev/null:**

```
34. user@desktop ~ $ find /etc -type f -exec grep -Hn "nameserver" {} \;
    2>/dev/null
35. /etc/ppp/ip-up:24:  # merge the new nameservers with the other
    optio...
36. /etc/ppp/ip-up:26:  grep --invert-match '^nameserver[[:space:]]
37. /etc/ppp/resolv.conf:1:nameserver 10.22.10.20
```

...

Таким образом, команда **grep** может эффективно использоваться вместе с командой **find** для нахождения информации в файлах.

Задания для самоподготовки

1. Создайте каталог **test1** в домашней директории. Сравните время создания системных каталогов **/bin**, **/tmp** с каталогом **test1**.

2. Скопируйте файл /bin/lis в локальную директорию. Посмотрите атрибуты этого файла. Попробуйте запустить его.
3. Создайте в локальной директории символическую ссылку tmplnk на директорию /tmp. Скопируйте несколько файлов в директорию tmplnk.
4. Сравните файлы /dev/tty1 и /dev/hda1. Какой тип они имеют? Чем они отличаются?
5. Найдите все файлы в системе, которые были модифицированы не более суток назад.
6. С помощью одной команды найдите все файлы с расширением .html в директории /usr и скопируйте их в локальную папку htmls.
7. Создайте архив директории /etc и узнайте его размер. Попробуйте создать сжатый архив той же папки. Сравните степень сжатия gzip и bzip2 на этом примере.
8. Создайте текстовый файл a.txt из десяти строк. Узнайте его размер. Создайте новый файл, содержащий в себе файл a.txt четыре раза.
9. С помощью одной команды найдите все файлы с расширением .txt в директории /usr и объедините их в один большой файл big.txt.
10. Получите с помощью одной команды файл, содержащий все числовые идентификаторы пользователей, зарегистрированных в системе.
11. Получите с помощью одной команды строки с 5-ой по 8-ю в файле /proc/cpuinfo, содержащем информацию о процессоре.
12. Придумайте регулярное выражение, соответствующее URL с использованием протокола http. Найдите все строки, содержащие такие URL в файлах директории /etc.
13. С помощью одной команды найдите все файлы с расширениями .txt или .html в директории /usr, в которых содержится слово user.

Практическое занятие «Организация работы с разделяемой памятью»

Цель занятия: научиться организовывать работы с разделяемой памятью.

Содержание работы:

1. Запустить операционную систему.
2. Войти в виртуальную машину, контейнер или на удаленный сервер приложений (IP адресом XX.XX.XX.254, пользователь lxc<NN>, пароль спросить у преподавателя).
3. В домашнем каталоге пользователя создать каталог work.
4. Написать программу для обмена текстовыми сообщениями между процессами, с использованием механизма разделяемой памяти. Обеспечить синхронизацию обмена с помощью механизма семафоров.
5. Написать makefile, обеспечивающий трансляцию, установку, очистку и удаление программы (см. лаб. 4).
6. Оттранслировать программу и установить ее в каталог bin каталога work с помощью команды make.
7. Очистить каталог work от вспомогательных файлов с помощью команды make.
8. Запустить оттранслированную программу.
9. Представить результаты выполнения работы преподавателю.

Практическое занятие «Изучение понятия протоколов в UNIX»

Цель занятия: изучение инструментов конфигурирования сети в UNIX, включающих настройку параметров TCP/IP-сети.

Содержание работы:

Изучаемые команды: [arp](#), [ifconfig](#), [netstat](#), [ping](#), [route](#), [ssh](#), [telnet](#), [traceroute](#)

Сценарий: Настройка сетевого интерфейса

Сценарий посвящен сетевым интерфейсам – прослойке между канальным и сетевым уровнем в UNIX. В сценарии показано, как получать информацию о настроенных в системе сетевых интерфейсах и как производится их диагностика и конфигурирование.

Начальные условия: Командная строка суперпользователя после входа в систему.

1. Получить сведения обо всех настроенных сетевых интерфейсах с помощью команды **ifconfig -a**:
2. desktop ~ # ifconfig -a
3. eth0 Link encap:Ethernet HWaddr 00:0D:60:8D:42:AA
4. inet addr:192.168.1.5 Bcast:192.168.1.255 Mask:255.255.255.0
5. UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
6. RX packets:204779 errors:0 dropped:0 overruns:0 frame:0
7. TX packets:107606 errors:0 dropped:0 overruns:0 carrier:0
8. collisions:0 txqueuelen:1000
9. RX bytes:302429520 (288.4 Mb) TX bytes:9177476 (8.7 Mb)
10. Base address:0x8000 Memory:c0220000-c0240000
- 11.
12. lo Link encap:Local Loopback
13. inet addr:127.0.0.1 Mask:255.0.0.0
14. UP LOOPBACK RUNNING MTU:16436 Metric:1
15. RX packets:228 errors:0 dropped:0 overruns:0 frame:0
16. TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
17. collisions:0 txqueuelen:0
- RX bytes:17724 (17.3 Kb) TX bytes:17724 (17.3 Kb)
18. Проверить возможность соединения с локальной машиной с помощью команды **ping 127.0.0.1**.
19. desktop ~ # ping 127.0.0.1
20. PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
21. 64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.052 ms
22. 64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.051 ms
- 64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.055 ms
23. Перед конфигурированием интерфейса *eth0* необходимо убедиться, что он отключен. Отключение сетевого интерфейса *eth0* производится командой **ifconfig eth0 down**.
24. desktop ~ # ifconfig eth0 down
25. desktop ~ # ifconfig -a
26. lo Link encap:Local Loopback
27. inet addr:127.0.0.1 Mask:255.0.0.0
28. UP LOOPBACK RUNNING MTU:16436 Metric:1
29. RX packets:228 errors:0 dropped:0 overruns:0 frame:0
30. TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
31. collisions:0 txqueuelen:0
- RX bytes:17724 (17.3 Kb) TX bytes:17724 (17.3 Kb)
32. Для связи сетевого интерфейса *eth0* с IP-адресом 192.168.1.1 выполним команду **ifconfig eth0 192.168.1.1 up**.
33. desktop ~ # ifconfig eth0 192.168.1.1 up
34. desktop ~ # ifconfig -a
35. eth0 Link encap:Ethernet HWaddr 00:0C:F1:2E:0E:F9
36. inet addr:192.168.1.1 Bcast:192.168.1.255 Mask:255.255.255.0
37. UP BROADCAST MULTICAST MTU:1500 Metric:1
38. RX packets:0 errors:0 dropped:0 overruns:0 frame:0
39. TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

40. collisions:0 txqueuelen:1000
41. RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
42. Interrupt:11 Base address:0x2000 Memory:c0210000-c0210fff
- 43.
44. lo Link encap:Local Loopback
45. inet addr:127.0.0.1 Mask:255.0.0.0
46. UP LOOPBACK RUNNING MTU:16436 Metric:1
47. RX packets:228 errors:0 dropped:0 overruns:0 frame:0
48. TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
49. collisions:0 txqueuelen:0
RX bytes:17724 (17.3 Kb) TX bytes:17724 (17.3 Kb)
- При этом по умолчанию используется сеть класса C, т.е. маска сети «255.255.255.0».
50. Для задания специфической маски подсети используется параметр *netmask*.
Например, данная команда задаёт параметры сети класса A: **ifconfig eth0 10.10.1.1 netmask 255.0.0.0 up**.
51. desktop ~ # ifconfig eth0 10.10.1.1 netmask 255.0.0.0 up
52. desktop ~ # ifconfig -a
53. eth0 Link encap:Ethernet HWaddr 00:0C:F1:2E:0E:F9
54. inet addr:10.10.1.1 Bcast:10.255.255.255 Mask:255.0.0.0
55. UP BROADCAST MULTICAST MTU:1500 Metric:1
56. RX packets:0 errors:0 dropped:0 overruns:0 frame:0
57. TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
58. collisions:0 txqueuelen:1000
59. RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
60. Interrupt:11 Base address:0x2000 Memory:c0210000-c0210fff
- 61.
62. lo Link encap:Local Loopback
63. inet addr:127.0.0.1 Mask:255.0.0.0
64. UP LOOPBACK RUNNING MTU:16436 Metric:1
65. RX packets:228 errors:0 dropped:0 overruns:0 frame:0
66. TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
67. collisions:0 txqueuelen:0
RX bytes:17724 (17.3 Kb) TX bytes:17724 (17.3 Kb)
68. С помощью команды **arp** можно узнать текущую ARP-таблицу операционной системы (соответствие MAC-адресов канального уровня IP-адресам). Таблица автоматически поддерживается операционной системой в процессе сетевого обмена.
69. desktop ~ # arp
70. Address HWtype HWaddress Flags Mask Iface
gate.localnet ether 00:02:44:8F:16:B7 C eth0

Сценарий: Настройка таблицы маршрутизации

В сценарии производится изучение и настройка таблицы маршрутизации IP. С помощью специальной программы производится изучение маршрута следования пакетов.

Начальные условия: Командная строка суперпользователя, сетевой интерфейс настроен на статический IP-адрес.

1. Для просмотра таблицы маршрутизации воспользуемся командой **route -n**:
2. desktop ~ # route -n
3. Kernel IP routing table
4. Destination Gateway Genmask Flags Metric Ref Use Iface
5. 10.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 eth0
6. 127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo

- 0.0.0.0 10.10.1.254 0.0.0.0 UG 0 0 0 eth0
7. Без использования ключа *-n* для всех имён будут использоваться символичные значения: **route**
 8. desktop ~ # route
 9. Kernel IP routing table
 10. Destination Gateway Genmask Flags Metric Ref Use Iface
 11. localnet * 255.0.0.0 U 0 0 0 eth0
 12. loopback * 255.0.0.0 U 0 0 0 lo
 - default gate.localnet 0.0.0.0 UG 0 0 0 eth0
 13. Для добавления новой строки в таблицу нужно воспользоваться параметром *add*:
route add -host 10.10.2.1 dev eth0.
 14. desktop ~ # route add -host 10.10.2.1 dev eth0
 15. desktop ~ # route -n
 16. Kernel IP routing table
 17. Destination Gateway Genmask Flags Metric Ref Use Iface
 18. 10.10.2.1 0.0.0.0 255.255.255.255 UH 0 0 0 eth0
 19. 10.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 eth0
 20. 127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
 - 0.0.0.0 10.10.1.254 0.0.0.0 UG 0 0 0 eth0
 - Эта команда добавляет явный маршрут до отдельного хоста с указанным IP-адресом через интерфейс *eth0*.
 21. Аналогичным образом маршрут удаляется, используется параметр *del*: **route del -host 10.10.2.1.**
 22. desktop ~ # route add -host 10.10.2.1 dev eth0
 23. desktop ~ # route -n
 24. Kernel IP routing table
 25. Destination Gateway Genmask Flags Metric Ref Use Iface
 26. 10.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 eth0
 27. 127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
 - 0.0.0.0 10.10.1.254 0.0.0.0 UG 0 0 0 eth0
 28. В качестве назначения маршрута можно указывать также целую сеть (параметр *-net*). Рассмотрим команду, которая задаёт маршрут в сеть «192.168.1.0» через шлюз «10.10.1.253»: **route add -net 192.168.1.0 gw 10.10.1.253.**
 29. desktop ~ # route add -net 192.168.1.0 gw 10.10.1.253
 30. desktop ~ # route -n
 31. Kernel IP routing table
 32. Destination Gateway Genmask Flags Metric Ref Use Iface
 33. 192.168.1.0 10.10.1.253 255.255.255.255 UG 0 0 0 eth0
 34. 10.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 eth0
 35. 127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
 - 0.0.0.0 10.10.1.254 0.0.0.0 UG 0 0 0 eth0
 36. Рассмотрим простой маршрут движения пакетов до хоста в Internet с помощью программы **traceroute ya.ru**:
 37. desktop ~ # traceroute ya.ru
 38. traceroute to ya.ru (213.180.204.8), 64 hops max, 40 byte packets
 39. 1 10.10.1.254 (10.10.1.254) 3.418 ms 2.67 ms 0.719 ms
 40. 2 cs7206.rinet.ru (195.54.192.28) 1.34 ms 1.378 ms 0.647 ms
 41. 3 ix2-m9.yandex.net (193.232.244.93) 1.554 ms 1.457 ms 1.420 ms
 42. 4 c3-vlan4.yandex.net (213.180.210.146) 2.137 ms 2.154 ms 1.842 ms
 - 5 ya.ru (213.180.204.8) 2.646 ms 2.183 ms 2.220 ms

Сценарий: Изучение службы доменных имён

Сценарий посвящен изучению службы доменных имён – её использованию и конфигурированию.

Начальные условия: Командная строка суперпользователя, сетевой интерфейс настроен на статический IP-адрес.

1. Просмотрим содержимое файла /etc/hosts, содержащего имена локальных хостов:
cat /etc/hosts
2. desktop ~ # cat /etc/hosts
3. #
4. # hosts This file describes a number of hostname-to-address
5. # mappings for the TCP/IP subsystem. It is mostly
6. # used at boot time, when no name servers are running.
7. # On small systems, this file can be used instead of a
8. # "named" name server. Just add the names, addresses
9. # and any aliases to this file...
10. #
- 11.
12. # Localhost
13. 127.0.0.1 localhost
- 14.
15. # Home LAN
16. 10.10.1.254 gate.localnet gate
17. 10.10.1.20 boss.localnet boss
18. Проверим работоспособность DNS с помощью команды обращения к хосту в Internet по имени **ping ya.ru**:
19. desktop ~ # ping ya.ru
20. PING ya.ru (213.180.204.8) 56(84) bytes of data.
21. 64 bytes from ya.ru (213.180.204.8): icmp_seq=1 ttl=54 time=3.56 ms
22. 64 bytes from ya.ru (213.180.204.8): icmp_seq=2 ttl=54 time=2.22 ms
23. Для корректной работы службы доменных имен необходимо прописать используемые серверы DNS в файле /etc/resolv.conf. Просмотрим его содержимое **cat /etc/resolv.conf**:
24. desktop ~ # cat /etc/resolv.conf
25. domain localnet
26. nameserver 10.10.1.17
27. С помощью команды **host ya.ru** узнаем информацию DNS о хосте в Internet:
28. desktop ~ # host ya.ru
29. ya.ru has address 213.180.204.8
30. ya.ru mail is handled by 10 cmail.yandex.ru.
31. Вторым аргументом команды **host ya.ru ns1.yandex.ru** имя DNS-сервера, с которого необходимо получить информацию:
32. desktop ~ # host ya.ru ns1.yandex.ru
33. ya.ru has address 213.180.204.8
34. Using domain server:
35. Name: ns1.yandex.ru
36. Address: 213.180.193.1#53
37. Aliases:
- 38.
39. Using domain server:
40. Name: ns1.yandex.ru
41. Address: 213.180.193.1#53
42. Aliases:

39.

ya.ru mail is handled by 10 cmail.yandex.ru.

Сценарий: Простая диагностика работы сети

Сценарий рассматривает самые простые способы диагностики работы сети.

Начальные условия: Командная строка суперпользователя, сетевой интерфейс настроен на статический IP-адрес.

1. Для проверки работоспособности сетевых служб воспользуемся командой удалённого терминала: **telnet ya.ru 80**. В данном случае будет установлено соединение с хостом в Internet по порту 80 (HTTP):
2. desktop ~ # telnet ya.ru 80
3. Trying 213.180.204.8...
4. Connected to ya.ru.
5. Escape character is '^']'.
6. GET / HTTP/1.0
- 7.
8. HTTP/1.0 200 OK
9. Server: thttpd/2.25b 29dec2003
10. Content-Type: text/html; charset=windows-1251
11. Date: Wed, 23 Nov 2005 05:40:33 GMT
12. Last-Modified: Mon, 07 Nov 2005 15:13:14 GMT
13. Accept-Ranges: bytes
14. Connection: close
15. Content-Length: 2005
- 16.
17. <html>
18. <head>
- ...
19. Если во время соединения с удалённым узлом ввести команду **netstat -t**, то можно увидеть, что состояние этого соединения – «ESTABLISHED»:
20. desktop ~ # netstat -t
21. Active Internet connections (servers and established)
tcp 0 0 desktop:42639 ya.ru:http ESTABLISHED
22. Информацию обо всех соединениях в системе можно получить с помощью команды **netstat -a**. В этом случае будет выводиться информация обо всех TCP-, UDP- и локальных сокетах:
23. desktop ~ # netstat -a
24. Active Internet connections (servers and established)
25. Proto Recv-Q Send-Q Local Address Foreign Address State
26. tcp 0 0 *:32769 *: LISTEN
27. tcp 0 0 *:32770 *: LISTEN
28. tcp 0 0 *:sunrpc *: LISTEN
29. tcp 0 0 *:ssh *: LISTEN
30. tcp 0 0 desktop:42639 ya.ru:http ESTABLISHED
31. udp 0 0 *:32768 *:
32. udp 0 0 *:32769 *:
33. udp 0 0 *:sunrpc *:
34. Active UNIX domain sockets (servers and established)
35. Proto RefCnt Flags Type State I-Node Path
36. unix 2 [ACC] STREAM LISTENING 8344 /var/run/acpid.socket
37. unix 2 [ACC] STREAM LISTENING 8866 /var/run/sdp
- ...

Сценарий: Работа по удалённому терминалу

Сценарий рассматривает работу по удалённому сетевому терминалу с использованием программы ssh.

Начальные условия: Командная строка суперпользователя, сетевой интерфейс настроен на статический IP-адрес.

1. С помощью команды **ssh user@10.10.1.222**
2. desktop ~ # ssh user@10.10.1.222
3. Password:
4. Last login: Sat Nov 21 15:56:20 2005 from 10.10.1.5
5. user@remote ~ \$
user@remote ~ \$ exit
6. Выполним команду **who**, чтобы убедиться, что находимся на удалённой машине.
Для всех пользователей, работающих удалённо, указывается IP-адрес.
7. user@remote ~ \$ who
8. user vc/1 Nov 14 14:04
user pts/0 Nov 22 10:55 (10.10.1.5)
9. Для завершения сеанса удалённого терминала нужно выйти из командной оболочки с помощью команды **exit**.
10. user@remote ~ \$ exit
11. logout
12. Connection to 10.10.1.5 closed.
desktop ~ #

Практическое занятие «Поиск информации в справочной системе UNIX»

Цель занятия: научиться выполнять поиск информации в справочной системе UNIX.

Содержание работы:

Система UNIX с самого начала ее использования снабжалась богатым комплектом документации. Некоторое количество информации часто находится в каталогах /usr/doc или /usr/local/doc в виде текстовых файлов. Традиционной для UNIX'a с момента ее создания является команда man, созданная еще в эпоху телетайп-терминалов и отлично работающая до сих пор на всех видах оборудования, но имеющая определенные ограничения: невозможность использования графических иллюстраций и гипертекстовых ссылок. В рамках GNU-проекта была создана система info, также работающая на всех видах алфавитно-цифровых терминалов, но с поддержкой гипертекста. Для всех GNU-утилит прилагаются соответствующие справочные файлы как в формате man, так и в формате info. Практически каждым коммерческим производителем UNIX-систем была создана собственная система помощи, включавшая как поддержку гипертекстов, так и графику, и работающая под управлением системы X Window. С появлением HTML справочная информация стала предоставляться в этом формате прямо в составе системы или на WWW-серверах компаний производителей.

Команда man

Синтаксис:

SYSV	man [-t] [-s i] имя
GNU, BSD	man [-t] [-i] имя
SYSV, GNU, BSD	man [-k]

Описание:

Команда man выводит страницу руководства для указанного имени на стандартный вывод или при помощи pager'a, установленного для данной сессии для постраничной выдачи информации.

Каждая страница руководства имеет стандартную форму со следующими разделами:

- NAME — название и назначение
- SYNOPSIS — синтаксис
- DESCRIPTIONS — описание
- FILE — используемые файлы
- SEE ALSO — смежные разделы
- DIAGNOSTIC — диагностика ошибок
- BUGS — замеченные ошибки

В UNIX-системах наблюдается две основные схемы разбиения страниц руководства на разделы:

Раздел	BSD	SYSV
Команды пользователя	1	1
Системные вызовы	2	2
Библиотечные функции	3	3
Спец. файлы и внешние устройства	4	7
Форматы файлов	5	4
Игры и демонстрации	6	6 или 1 или отсутствуют
Разное (наборы символов, типы файловых систем и т.д.)	7	7
Команды для системного администрирования	8	1m
Команды поддержки	8	8
Драйверы устройств	4	7 или 9

Таблица 3.1. Схемы разбиения страниц руководства на разделы

Опции:

-k	Работает аналогично команде apropos
-t	Осуществляет вывод информации в формате PostScript
-s <i>i</i> , <i>i</i>	Номер <i>i</i> указывает секцию руководства, в которой надо производить поиск

Примеры: Получение справки о системной команде man.

```
$ manman
man(1)      man(1)

NAME
man - format and display the on-line manual pages

SYNOPSIS
man [-acdfFhkKtwW] [--path] [-m system] [-p string]
[-C config_file] [-M pathlist] [-P pager] [-B
browser] [-H htmlpager] [-S section_list] [section] name ...

DESCRIPTION
```

man formats and displays the on-line manual pages. If you specify section, man only looks in that section of the manual. name is normally the name of the manual page, which is typically the name of a command, function, or file. However, if name contains a slash (/) then man interprets it as a file specification, so that you can do man ./foo.5 or even man /cd/foo/bar.1.gz.

See below for a description of where man looks for the manual page files.

Справочная система **whatis**

Синтаксис:

whatis *имя...*

Описание:

Выводит краткое описание указанной страницы руководства.

Пример:

```
$ whatis mount
mount (2) - mount and unmount filesystems
mount (8) - mount a file system
```

Примечание:

Требует наличия специальной базы данных, предварительно созданной командой catman или makewhatis.

Команда **apropos**

Синтаксис:

apropos *слово*

Описание:

Ищет страницы описаний по ключевому слову.

Примеры:

```
$ aproposprintf
format (n) - Format a string in the style of sprintf
printf (1) - format and print data
printf (3) - formatted output conversion
printf (3p) - print formatted output
```

Примечание:

Требует наличия специальной базы данных, предварительно созданной командой catman или makewhatis

Справочная система **info**

Синтаксис:

info [-directory *каталог...*] [-file *файл*]

Описание:

Просмотр справочной информации в формате *texinfo*. Путь поиска файлов *info* помещается в переменную \$INFOPATH.

Опции:

-directory <i>каталог</i>	Список каталогов для поиска
-file <i>файл</i>	Читать из файла

Управляющие последовательности:

h	Просмотр руководства
?	Получение подсказки
n	Следующий раздел

p	Предыдущий раздел
u	Подняться на уровень
m	Выбор пункта меню
f	Перейти по ссылке
SPACE	Пролистывание вниз
DEL	Пролистывание вверх
q	Выход

Примеры:

```
$ info --directory /usr/share/info/
```

```
File: dir Node: Top This is the top of the INFO tree
```

This (the Directory node) gives a menu of major topics.

Typing "q" exits, "?" lists all Info commands, "d" returns here,

"h" gives a primer for first-timers,

"mEmacs<Return>" visits the Emacs topic, etc.

In Emacs, you can click mouse button 2 on a menu item or cross reference to select it.

* Menu:

Texinfo documentation system

* Pinfo: (pinfo). curses based lynx-style info browser.

* Texinfo: (texinfo). The GNU documentation format.

* info standalone: (info-std) Read Info documents without Emacs.

* infokey: (info-std) Invoking infokey. Compile Info customizations.

* install-info: (texinfo) Invoking install-info. Update info/dir entries.

* makeinfo: (texinfo) Invoking makeinfo. Translate Texinfo source.

* texi2dvi: (texinfo) Format with texi2dvi. Print Texinfo documents.

* texi2pdf: (texinfo) PDF Output. PDF output for Texinfo.

* texindex: (texinfo) Format with tex/texindex. Sort Texinfo index files.

Miscellaneous

* As: (as). The GNU assembler.

* Bfd: (bfd). The Binary File Descriptor library.

* Binutils: (binutils). The GNU binary utilities.

Практическое занятие «Управление безопасностью в UNIX.»

Цель занятия: научиться организовывать работу по безопасности в Unix.

Содержание работы:

Сценарий посвящен изучению процесса аутентификации в UNIX.

Начальные условия: Приглашение входа в систему.

1. Для входа в систему необходимо ввести логин и пароль пользователя. При этом символы вводимого пароля не отображаются.
2. Welcome to desktop / tty1
3. desktop login: user
4. Password:

5. Last login: Wed Dec 07 00:20:09 2006 from tty1
user@desktop ~ \$
При входе в систему в графическом режиме (через X11), пользователю вместо командной строки предоставляется графический десктоп.
6. Для завершения сеанса работы в командной строке необходимо ввести команду **exit**:
user@desktop ~ \$ exit

Сценарий: Изучение базовых прав доступа

Сценарий рассматривает базовые права доступа в UNIX, их просмотр и изменения.

Начальные условия: Командная строка после входа в систему.

1. Для просмотра прав доступа можно использовать команду ls с ключом вывода расширенной информации: **ls -l**.
2. user@desktop ~ \$ ls -l
3. drwxr-xr-x 1 user users 22 Дек 19 11:18 test/
-rw-r--r-- 1 user users 90 Сен 19 00:20 test.txt
Особый интерес представляют: первая колонка (права доступа), третья и четвёртая – владелец и группа владельцев соответственно.
4. Рассмотрим исполняемый файл **ls -l /bin/bash**.
5. user@desktop ~ \$ ls -l /bin/bash
-rwxr-xr-x 1 root root 746544 Дек 21 13:40 /bin/bash*
Исполняемые файлы в UNIX определяются наличием специального бита прав доступа.
6. Для директории права на исполнения трактуются по-другому. Рассмотрим права директории **/tmp** **ls -ld /tmp**:
7. user@desktop ~ \$ ls -ld /tmp
drwxrwxrwt 26 root root 5168 Дек 22 20:04 /tmp/
Директория имеет дополнительный sticky-bit, определяющий права на создание и удаление файлов в директории.
8. Изменение файла с недостатком прав приводит к ошибке доступа. Например, команда: **rm /bin/bash**.
9. user@desktop ~ \$ rm /bin/bash
10. rm: удалить защищенный от записи обычный файл `/bin/bash'? y
rm: невозможно удалить `/bin/bash': Permission denied
Для всех файлов, на которые данный пользователь не имеет прав записи команда rm может выводить предупреждение об удалении.
11. Изменение прав доступа производится с помощью команды chmod. Для задания файлу прав только для чтения воспользуемся командой: **chmod a=r test.txt**.
user@desktop ~ \$ chmod a=r test.txt
12. Для лишения всех прав группы владельцев и остальных пользователей воспользуемся командой: **chmod go-rwx test.txt**.
user@desktop ~ \$ chmod go-rwx test.txt

Сценарий: Переход в режим суперпользователя

Сценарий рассматривает процесс повышения привилегий пользователя и переход в режим суперпользователя.

Начальные условия: Командная строка после входа в систему простого пользователя.

1. Некоторые исполняемые программы обладают специальным suid-битом, например, программа passwd, рассмотрим права доступа к этому исполняемому файлу: **ls -l /bin/passwd**.
2. user@desktop ~ \$ ls -l /bin/passwd
-rws--x--x 1 root root 28660 Янв 8 13:05 /bin/passwd*

3. Запускаются suid-программы от имени владельца файла. В этом можно убедиться, если запустить команду **passwd**, а затем на другом терминале сделать **ps aux | grep passwd**:
4. user@desktop ~ \$ passwd
5. Changing password for user
6. (current) UNIX password:
7. ...
8. user@desktop ~ \$ ps aux | grep passwd
9. root 12937 0.0 0.1 3228 1012 pts/2 S+ 23:28 0:00 passwd
user 12989 0.0 0.1 2740 748 pts/3 R+ 23:28 0:00 grep passwd
10. Для длительной работы в режиме суперпользователя обычно используют команду su. Используем параметр -, чтобы проинициализировать окружение суперпользователя: **su -**. Для того, чтобы перейти в режим суперпользователя, необходимо знать его пароль.
11. user@desktop ~ \$ su -
12. Password:
desktop ~ #
При этом запускается новая командная оболочка, уже с новыми привилегиями.
13. Для завершения сеанса суперпользователя необходимо воспользоваться командой **exit**:
14. desktop ~ # exit
15. logout
user@desktop ~ \$

Сценарий: Изучение базы данных пользователей

В сценарии показано, где и каким образом хранится информация о пользователях системы.

Начальные условия: Командная строка после входа в систему.

1. Данные о зарегистрированных в системе пользователях хранятся в файле /etc/passwd. Рассмотрим его содержимое **cat /etc/passwd**:
2. user@desktop ~ \$ cat /etc/passwd
3. root:x:0:0:root:/root:/bin/bash
4. bin:x:1:1:bin:/bin:/bin/false
5. daemon:x:2:2:daemon:/sbin:/bin/false
6. adm:x:3:4:adm:/var/adm:/bin/false
- ...
Подробную информацию о формате этого файла можно получить на соответствующей странице руководств: **man 5 passwd**.
7. Аналогичным образом данные о группах хранятся в файле /etc/group. Рассмотрим его содержимое **cat /etc/group**:
8. user@desktop ~ \$ cat /etc/group
9. root:x:0:root
10. bin:x:1:root,bin,daemon
11. daemon:x:2:root,bin,daemon
12. sys:x:3:root,bin,adm
- ...

Сценарий: Добавление и удаление пользователей

Сценарий описывает администрирование пользователей системы: добавление новых пользователей, удаление существующих, изменение параметров пользователей. Такие изменения учётных записей пользователей доступны только суперпользователю.

Начальные условия: Командная строка после входа в систему суперпользователя.

1. Для добавления новых пользователей используется команда useradd. При этом в качестве параметров можно указать домашнюю директорию и командную

- оболочку пользователя: **useradd testuser -d /home/users/testuser -s /bin/sh**,
результат можно увидеть следующим образом: **cat /etc/passwd | grep testuser**.
2. `desktop ~ # cat /etc/passwd`
`testuser:x:1003:1003::/home/users/testuser:/bin/sh`
 3. С помощью команды **passwd** можно задать пароль для нового пользователя: **passwd testuser**.
 4. `desktop ~ # passwd testuser`
 5. New UNIX password:
 6. Retype new UNIX password:
`passwd: пароль успешно обновлён`
 7. Для изменения параметров учётной записи можно отредактировать файл `/etc/passwd`, однако более корректным способом является использование команды **usermod**. Например, для изменение командной оболочки пользователя на `/bin/false` приведёт к невозможности его входа в систему: **usermod -s /bin/false testuser**.
`desktop ~ # usermod -s /bin/false testuser`
 8. Удаление пользователя производится с помощью команды **userdel**: **userdel testuser**.
`desktop ~ # userdel testuser`

Практическое занятие «Управление процессами в UNIX.»

Цель занятия: Научиться управлять процессами.

Содержание работы:

Задание

1. Выведите на экран листинг характеристик (в длинном и коротком форматах) процессов, инициализированных с Вашего терминала. Проанализируйте и объясните содержание каждого поля сообщения.
2. Выведите на экран листинг характеристик всех процессов. Используйте при необходимости конвейер с `more` для постраничного просмотра листинга. Какой процесс является родительским для большинства процессов? Что означает символ `?` в поле управляющий терминал процесса?
3. Выведите на экран листинг процессов, запущенных конкретным пользователем. Какой ключ пришлось использовать? Что говорит значение `?` в поле управляющий терминал процесса?
4. Разработайте и запустите простейшую процедуру в фоновом режиме с бесконечным циклом выполнения, предусматривающую, например, перенаправление вывода каких-то сообщений в файл или в фиктивный файл, и использующую команду `sleep` для сокращения частоты циклов процедуры.
5. Выполните п. 1. Объясните изменения в листинге характеристик процессов.
6. Понижьте значение приоритета процедуры. На что и как повлияет эта операция при управлении вычислительным процессом системы? Как отразятся ее результаты в описателях процессов?
7. Проанализируйте листинг процессов. Какой процесс является родительским для процедуры.
8. Выйдите из системы и войдите заново. Проанализируйте листинг процессов. Объясните изменения в системе.
9. Запустите процедуру в фоновом режиме, но предусмотрите ее защиту от прерывания при выходе из системы.
10. Выполните п.6. Объясните изменения PPID процедуры.
11. Завершите выполнение процесса процедуры.
12. Запустите процедуру в интерактивном режиме с перенаправлением вывода в соответствующий файл.

13.Переведите задание с процедурой в фоновый режим и проанализируйте сообщение на экране. Что пришлось дополнительно сделать? Как выглядят приостановленные процессы в листинге команды ps?

14.Переведите задание с процедурой в интерактивный режим и проанализируйте сообщение на экране.

15.Завершите выполнение процедуры и проанализируйте сообщение на экране.

16.Поставьте эксперимент, позволяющий определить, что будет происходить с процедурой, запущенной в фоновом режиме, в случае попытки ввода с клавиатуры. Как все-таки и обеспечить ввод?

17.Поставьте эксперимент, иллюстрирующий относительные скорости выполнения нескольких фоновых процессов, запущенных с разными значениями поправки к приоритету. Завершите сразу все фоновые процессы одной командой pkill. Какие опции команды пришлось использовать для выделения фоновых процессов, запущенных с Вашего терминала?

Контрольные вопросы:

1.Какую комбинацию клавиш нужно использовать для принудительного завершения задания, запущенного в интерактивном режиме?

2.Какую комбинацию клавиш нужно использовать для при остановке задания, запущенного в интерактивном режиме?

3.Какая команда позволяет послать сигнал конкретному процессу?

4.Какая команда позволяет поменять поправку к приоритету уже запущенного процесса?

5.Какая команда позволяет запустить задание с пониженным приоритетом?

6.Какая команда позволяет запустить задание с защитой от прерывания при выходе из системы пользователя?

7.Какой процесс всегда присутствует в системе и является предком всех процессов?

8.Каким образом можно запустить задание в фоновом режиме?

9.Каким образом задание, запущенное в фоновом режиме, можно перевести в интерактивный режим?

10.Каким образом приостановленное задание можно перевести в интерактивный режим?

11.Что произойдет с заданием, выполняющимся в фоновом режиме, если оно попытается обратиться к терминалу?

Практическое занятие «Управление службами в UNIX»

Цель занятия: научиться управлять службами в Unix.

Содержание работы:

Загрузку операционной системы можно разделить на несколько этапов. Начальный этап загрузки не зависит от того, какая операционная система установлена на компьютере, он в первую очередь связан с особенностями архитектуры используемого компьютера. Затем следуют этапы загрузчиков, которые также могут не относиться к определённым операционным системам, после чего инициализируется ядро операционной системы и производятся специфические только для этой ОС операции.

Рассмотрим загрузку операционной системы UNIX как следующую последовательность этапов (Рисунок 6.1, «Этапы загрузки ОС UNIX»):

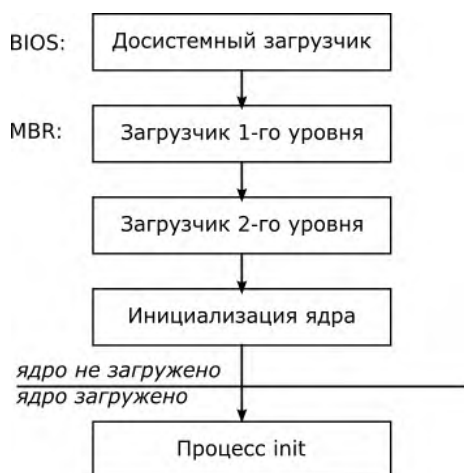


Рисунок 6.1. Этапы загрузки ОС UNIX

досистемный загрузчик

Как правило, сразу после включения питания программа ПЗУ BIOS проводит тестирование оборудования, затем запускается досистемный загрузчик.

Задача этого этапа – определить (возможно, с помощью пользователя), с какого устройства будет идти загрузка, загрузить оттуда специальную программу-загрузчик и запустить её. Например, выяснить, что устройство для загрузки – жесткий диск, считать самый первый сектор этого диска и передать управление программе, которая находится в считанной области.

загрузчик первого уровня

Загрузчик первого уровня занимает обычно не более одного сектора в самом начале диска – в его загрузочном секторе (MasterBootRecord).

Ядро операционной системы имеет довольно сложную структуру – а значит, и непростой способ загрузки; оно может быть довольно большим, и может располагаться неизвестно где на диске, подчиняясь законам файловой системы (например, состоять из нескольких частей, разбросанных по диску). Учесть все это первичный загрузчик не в состоянии, поэтому его задача: определить, где на диске находится *загрузчик второго уровня*, загрузить его в память и передать ему управление.

загрузчик второго уровня

Вторичный загрузчик – сложная программа с интерфейсом пользователя, который даёт возможность выбирать операционную систему или параметры загрузки ядра. Загрузчик должен иметь доступ к образу ядра, поэтому в него включается поддержка *файловых систем*.

В любом случае вторичный загрузчик читает образ ядра в определённый адрес памяти и передаёт туда управление.

инициализация ядра операционной системы

Как мы уже выяснили ранее, ядро – очень сложная программа, взаимодействующая с различным оборудованием, поэтому прежде чем начать работу с системой, её необходимо проинициализировать.

Этот этап специфичен для различных операционных систем. В UNIX-подобных системах при этом обычно выводится информация о загрузке ядра отладочного характера.

Первым делом ядро занимается определением параметров вычислительной подсистемы компьютера: выясняет тип и быстродействие центрального процессора, объем оперативной памяти, объем и структуру кэш-памяти; делает предположение об архитектуре компьютера в целом и многое другое.

На следующем шаге ядро определяет состав и архитектуру всего аппаратного наполнения компьютера: тип и параметры шин передачи данных и устройств управления ими (контроллеров), список внешних устройств, доступных по шинам, настройки этих устройств – диапазон портов ввода-вывода, адрес ПЗУ, занимаемое аппаратное

прерывание, номер канала прямого доступа к памяти и т.п..

Ядро на основании переданного ему параметра выбирает корневой раздел – файловую систему, содержащую будущий каталог /и его подкаталоги (для системной начальной загрузки важны каталоги/etc,/bin, и/sbin). Корневой раздел *монтируется* в качестве/. После этого ядро запускает свой первый процесс –*init*(по умолчанию,/sbin/init).

процесс init

С этого момента операционная система обеспечивает полноценную функциональность всем исполняющимся процессам. В UNIX первым запускаемым процессом является *init*, о котором сказано в следующем разделе.

Процесс *init*

Процесс *init* является обычным процессом операционной системы, однако он имеет некоторые особенности: его PID всегда равен 1, и процесс этот работает всё время, пока работает система.

В UNIX-системах *init* играет две важные роли:

- производит инициализацию системы – как правило, для работы запущенного ядра не достаточно, нужно примонтировать все файловые системы, загрузить дополнительные драйверы устройств, запустить демоны и т.п.;
- является родителем для всех процессов в системе – это является гарантией того, что в UNIX любой процесс имеет своего родителя.

Как правило, процесс *init* запускается из исполняемого файла /sbin/init и является специфичным для различных UNIX-систем. Рассмотрим примеры различных современных версий UNIX и их классификацию с точки зрения инициализации системы.

Различия загрузки UNIX-подобных систем

Исторически, различные версии UNIX наследовались от двух систем: оригинальной UNIX компании AT&T (вплоть до версии System V) и BSD UNIX, созданной в университете Беркли. В них применялись различные принципы загрузки системы, так что современные версии UNIX по этому критерию можно разделить на:

- наследники System V – так называемая UNIX SystemGroup (USG-системы): AIX, Solaris, UnixWare, Linux (дистрибутивы RedHat, Mandriva, ALT Linux);
- наследники BSD: семейство BSD, Linux (Slackware);
- смешанные схемы: Linux (Debian, Gentoo).

Системы, наследующие System V

Основным признаком этих систем является наличие *уровня выполнения*(runlevel) – одного из возможных режимов работы системы. Каждый уровень исполнения имеет свой номер – часть этих номеров стандартизована. В любой момент времени система может находиться в одном из них – изменение режима работы производится с помощью перезапуска *init* с указанным номером.

Уровень 0

остановка системы(halt) – работа системы должна быть прекращена;

Уровень 1

однопользовательский режим работы – система инициализирует минимум служб и даёт единственному пользователю (как правило, суперпользователю) без проведения аутентификации командную строку. Как правило, этот режим используется для восстановления системы;

Уровень 2

многопользовательский режим– пользователи могут работать на разных терминалах, вход в систему с процессом аутентификации;

Уровень 3

многопользовательский сетевой режим– в отличие от предыдущего уровня, осуществляется настройка сети и запускаются различные сетевые службы;

Уровень 4

не имеет стандартного толкования и практически не используется;

Уровень 5

запуск графической подсистемы— по сравнению с уровнем 3 производится также старт графической подсистемы X11 (см.Глава 7, *Графическая подсистема UNIX*), и вход в систему осуществляется уже в графическом режиме;

Уровень 6

перезагрузка системы— при включении этого режима останавливаются все запущенные программы и производится перезагрузка.

Таким образом, каждый уровень системы подразумевает запуск определённого набора программ, который может быть задан администратором системы. Стартовые скрипты, соответствующие уровням выполнения располагаются в директории /etc/rc.d.

На практике, в серверных системах обычно при старте системы используется 3-й уровень выполнения, в домашних – 5-й.

Системы, наследующие BSD

В этих системах традиционно используется *линейная схема* загрузки. Эта схема устроена проще, но она более сложна в администрировании.

Инициализация системы осуществляется единым скриптом /etc/rc. В этом скрипте прописаны последовательные команды инициализации системы, запуска демонов и т.п.. Следом за ним следует запуск скрипта/etc/rc.local, который служит для запуска всех *локальных* программ и настроек, установленных системным администратором сверх дистрибутива операционной системы.

При обновлении отдельных программ или изменении их настроек администратору приходится вручную править стартовые скрипты. Эти сложности привели к тому, что в современные BSD-системы внедряются более легкие в администрировании схемы загрузки.

Системы со смешанной схемой загрузки

Некоторые современные UNIX-подобные системы (в частности, многие дистрибутивы Linux) предоставляют собственные схемы загрузки системы, сочетающие в себе достоинства обозначенных выше схем.

Для примера можно рассмотреть схему, используемую в дистрибутивах LinuxDebian и Gentoo. Вводится понятие *программных уровней выполнения*(software runlevels) – которые могут создаваться и изменяться администратором системы.

Каждому уровню выполнения соответствует набор сервисов, которые будут запущены при переключении системы на этот уровень выполнения. По умолчанию, используется один уровень исполнения – *default*.

Службы связаны между собой через так называемые *зависимости*: к примеру, служба, монтирующая сетевые папки, требует наличия сконфигурированной сети, а значит зависит от службы конфигурации сети. Службы конфигурации сети, в свою очередь, зависит от службы, загружающей дополнительные модули ядра (например, драйвер сетевой карты).

Управление уровнями загрузки – какие программы необходимо запускать на каждом из них – производится аналогично System V-системам.

Конфигурация запуска init

Конфигурация процесса init находится в файле /etc/inittab.

Пример 6.1. Пример файла /etc/inittab

```
# Default runlevel.
```

```
id:3:initdefault:
```

```
# System initialization, mount local filesystems, etc.
```

```
si::sysinit:/sbin/rc sysinit
```

```
# Further system initialization, brings up the boot runlevel.
```

```
rc::bootwait:/sbin/rc boot
```

```
l0:0:wait:/sbin/rc shutdown
```

```
l1:S1:wait:/sbin/rc single
```

```

l2:2:wait:/sbin/rc nonetwork
l3:3:wait:/sbin/rc default
l4:4:wait:/sbin/rc default
l5:5:wait:/sbin/rc default
l6:6:wait:/sbin/rc reboot
# TERMINALS
c1:12345:respawn:/sbin/agetty 38400 tty1 linux
c2:2345:respawn:/sbin/agetty 38400 tty2 linux
c3:2345:respawn:/sbin/agetty 38400 tty3 linux
c4:2345:respawn:/sbin/agetty 38400 tty4 linux
c5:2345:respawn:/sbin/agetty 38400 tty5 linux
c6:2345:respawn:/sbin/agetty 38400 tty6 linux
# What to do at the "Three Finger Salute".
ca:12345:ctrlaltdel:/sbin/shutdown -r now
# Used by /etc/init.d/xdm to control DM startup.
x:a:once:/etc/X11/startDM.sh

```

Основными параметрами загрузки, задаваемыми в этом файле, являются:

- начальный уровень загрузки (строка с `initdefault`) – номер уровня выполнения, в который переводится система при старте;
- скрипты для запуска уровней исполнения – для каждого уровня (0 – 6) задаётся программа с аргументами запуска, которая будет запущена в случае перевода системы на один из уровней выполнения;
- настройки виртуальных терминалов – сколько необходимо инициализировать при старте системы, какую программу для этого использовать;
- настройка реакции на нажатие **Ctrl-Alt-Del** – какую программу необходимо запустить при этом;
- прочие настройки, специфичные для данной версии UNIX.

Системные службы

Системные службы – это программы, выполняющие в системе определённую функцию и, как правило, стартующие при запуске системы. Каждой системной службе соответствует *стартовый скрипт* – специальная программа, осуществляющая запуск демона или программы, которая и обеспечивает функциональность службы.

Таким образом, можно построить **дерево зависимостей**, и перезапуск одного скрипта будет приводить к перезапуску всех его потомков.

Запуск и остановка служб

В System V-системах стартовые скрипты находятся в директории `/etc/init.d` и принимают единственный стандартный аргумент – один из: «start», «stop», «restart». Таким образом, каждая служба может быть остановлена, запущена или перезапущена.

Например, для перезапуска службы системного журнала необходимо выполнить команду `/etc/init.d/syslogdrestart`.

Пример 6.2. Пример перезапуска службы

```

desktop ~ # /etc/init.d/syslogdrestart
* Stopping syslog-ng ... [ ok ]
* Starting syslog-ng ... [ ok ]
desktop ~ #

```

Как правило, для управления службами необходимо обладать правами суперпользователя.

Автоматическая загрузка служб

Службы используются в UNIX-системах, использующих System V-подобную схему загрузки системы. При этом каждому уровню выполнения соответствует набор служб, запускаемых при переключении на этот уровень.

В директории `/etc/rc.d` можно увидеть директории `rc0.d`, `rc1.d` и т.д. – по одной на

каждый уровень выполнения. В этих директориях содержатся ссылки на службы, которые будут запущены или остановлены при переходе на соответствующий уровень выполнения.

Особый интерес представляют имена ссылок на стартовые скрипты служб: например, `/etc/rc.d/rc0.d/K60crond` и `/etc/rc.d/rc3.d/S40crond`, указывающие на один скрипт `/etc/init.d/crond` службы системного журнала. Скрипт, начинающийся с «K» соответствует останову службы, а «S» – запуску. Числа, следующие перед именем службы задают порядок запуска скриптов в директории. Например, скрипт `/etc/rc.d/rc3.d/S34syslogd` будет запущен до скрипта `/etc/rc.d/rc3.d/S40crond`, тогда как `/etc/rc.d/rc3.d/K60crond` до `/etc/rc.d/rc3.d/K66syslogd`. Можно заметить, что сумма чисел для одной службы равна 100 – это позволяет упорядочить все скрипты в порядке старта, обратном порядку завершения.

Для установления связи между службами и уровнями выполнения используется утилита `chkconfig`.

Примеры служб

Рассмотрим примеры служб, существующих в том или ином виде практически во всех UNIX-системах:

cron

системный планировщик заданий – демон, запускающий определённые программы с заданными интервалами времени;

syslog

системный журнал – демон, организующий единый интерфейс для журналирования событий в системе;

network

служба инициализация сети – производит автоматическую настройку интерфейсов, маршрутизации и т.п. ;

iptables

служба инициализации межсетевого экрана в Linux;

sshd, xinetd, ftpd

набор сетевых служб, запускающих различные сетевые серверы;

sendmail

почтовый сервер – демон, обеспечивающий отправку и доставку почты по протоколу SMTP;

modules

служба, загружающая и инициализирующая дополнительные модули ядра;

local

служба, которая обычно запускается в последнюю очередь и позволяет администратору стартовать дополнительные программы при загрузке системы;

checkroot

служба, инициализирующая проверку корневой файловой системы (с использованием утилиты, специализированной для типа файловой системы).

Практическое занятие «Изучение семафоров в UNIX как средства синхронизации процессов»

Цель занятия: Изучение семафоров в UNIX как средства синхронизации процессов.

Содержание работы:

При разработке средств System V IPC семафоры вошли в их состав как неотъемлемая часть. Следует отметить, что набор операций над семафорами System V IPC отличается от классического набора операций {P, V}, предложенного Дейкстрой. Он насчитывает три операции:

· A(S, n) - увеличить значение семафора S на величину n;

- $D(S, n)$ - пока значение семафора $S < n$ процесс блокируется. Далее $S = S - n$;
- $Z(S)$ - процесс блокируется до тех пор, пока значение семафора S не станет равным 0.

Изначально все IPC семафоры иницируются нулевым значением.

Легко видеть, что классической операции $P(S)$ соответствует операция $D(S, 1)$, а классической операции $V(S)$ соответствует операция $A(S, 1)$. Аналогом ненулевой инициализации Дейкстровских семафоров значением n может служить выполнение операции $A(S, n)$ сразу после создания семафора S , с обеспечением атомарности создания семафора и ее выполнения посредством другого семафора. Мы показали, что классические семафоры реализуются через семафоры System V IPC. Обратное не является верным. Используя операции $P(S)$ и $V(S)$, мы не сумеем реализовать операцию $Z(S)$.

IPC семафоры являются средством связи с непрямой адресацией, требуют инициализации для организации взаимодействия процессов и специального действия для освобождения системных ресурсов по его окончании. Пространством имен IPC семафоров является множество значений ключа, генерируемых с помощью функции `ftok()`. Для совершения операций над семафорами системным вызовом в качестве параметра передаются IPC дескрипторы семафоров, однозначно идентифицирующих их во всей вычислительной системе, а вся информация о семафорах располагается в адресном пространстве ядра операционной системы. Это позволяет организовывать через семафоры взаимодействие процессов, даже не находящихся в системе одновременно.

Практическое занятие «Изучение очередей сообщений в UNIX и работа с ними»

Цель занятия: Изучение очередей сообщений в UNIX и работа с ними.

Содержание работы:

Для обеспечения возможности обмена сообщениями между процессами механизм очередей поддерживается следующими системными вызовами:

`msgget()`

для образования новой очереди сообщений или получения дескриптора существующей очереди;

`msgsnd()`

для постановки сообщения в указанную очередь сообщений;

`msgrcv()`

для выборки сообщения из очереди сообщений;

`msgctl()`

для выполнения ряда управляющих действий Прототипы перечисленных системных вызовов описаны в файлах

```
#include <sys/ipc.h>
```

```
#include <sys/msg.h>
```

По системному вызову `msgget()` в ответ на ключ(key) и набор флагов (полностью аналогичны флагам в системном вызове `semget()`)

ядро создает новую очередь сообщений и возвращает пользователю идентификатор созданной очереди, либо находит элемент таблицы очередей сообщений, содержащий указанный ключ, и возвращает соответствующий идентификатор очереди:

```
int msgqid = msgget(key_t key, int flag).
```

Для помещения сообщения в очередь служит системный вызов

`msgsnd()`:

```
int msgsnd (int msgqid, void *msg, size_t size, int flag),
```

где `msg` – это указатель на структуру длиной `size`, содержащую определяемый пользователем целочисленный тип сообщения и символьный массив-сообщение.

Структура `msg` имеет вид:

```
struct msg {long mtype; /* тип сообщения*/
```

```
char mtext[SOMEVALUE]; /* текст сообщения
```

(SOMEVALUE - любое*/);

Параметр flag

определяет действия ядра при выходе за пределы допустимых размеров внутренней буферной памяти (флаг IPC_NOW

AIT со значением, рассмотренным выше).

Условиями успешной постановки сообщения в очередь являются:

- наличие прав процесса по записи в данную очередь сообщений;
- не превышение длиной сообщения заданного системой верхнего предела;
- положительное значение указанного в сообщении типа сообщения.

Если же оказывается, что новое сообщение невозможно буферизовать в ядре по причине превышения верхнего предела суммарной длины сообщений, находящихся в данной очереди сообщений (флаг IPC_NOW

AIT при этом отсутствует), то обратившийся процесс откладывается (усыпляется) до тех пор, пока очередь сообщений не разгрузится процессами, ожидающими получения сообщений. Для приема сообщения используется системный вызов

msgrcv():

int msgrcv (int msgqid, void *msg, size_t size, long msg_type, int flag);

Системный вызов

msgctl()

int msgctl (int msgqid, int command, struct msqid_ds *msg_stat)

используется - для опроса состояния описателя очереди сообщений (command = IPC_STAT) и помещения его в структуру msg_stat (детали опускаем);

- изменения его состояния (command = IPC_SET), например, изменения прав доступа к очереди;
- для уничтожения указанной очереди сообщений (command = PC_RMID).

Практическое занятие «Организация файловой системы в UNIX»

Цель занятия: Организация файловой системы в UNIX.

Содержание работы:

1. Получить имя текущего каталога с помощью команды **pwd**:
2. user@desktop ~ \$ pwd
/home/user/
3. Перейти в корневой каталог с помощью команды **cd /**:
4. user@desktop ~ \$ cd /
5. user@desktop / \$ pwd
/
Отметить, как изменилась строка приглашения.
6. Посмотреть содержимое корневого каталога с помощью команды **ls**:
7. user@desktop / \$ ls
8. bin dev home mnt proc sbin tmp var
boot etc lib opt root sys usr
9. Сравнить с использованием «раскарашенного» вывода команды **ls -F --color**:
10. user@desktop / \$ ls -F --color
11. bin/ dev/ home/ mnt/ proc/ sbin/ tmp/ var/
boot/ etc/ lib/ opt/ root/ sys/ usr/
Директории отмечаются синим цветом и знаком «/» после имени.
12. Посмотреть содержимое домашнего каталога с помощью команды **ls -F --color ~**:
13. user@desktop / \$ ls -F --color ~
user@desktop / \$

Возможно, домашний каталог пуст.

14. Вернуться в домашний каталог с помощью **cd** без параметров:
15. user@desktop / \$ cd
user@desktop ~ \$
16. Создать директорию test с помощью команды **mkdir test**:
user@desktop ~ \$ mkdir test
17. Посмотреть обновлённое содержимое домашнего каталога **ls -F --color**:
18. user@desktop ~ \$ ls -F --color
test/
19. Создать поддиректорию subtest в директории test командой **mkdir test/subtest**:
user@desktop ~ \$ mkdir test/subtest
20. Посмотреть содержимое домашнего каталога и его подкаталогов с помощью ключа рекурсивного просмотра **-R** в команде **ls -F --color -R**:
21. user@desktop ~ \$ ls -F --color -R
22. .:
23. test/
24. test:
25. subtest/
26. test/subtest:

Сценарий: Изучение типов файлов в UNIX

1. Создать пустой файл с помощью команды **touch test/first.txt**:
2. user@desktop ~ \$ touch test/first.txt
3. user@desktop ~ \$ ls -F --color test/
first.txt subtest/
4. Добавить строку текста в конец файла с помощью команды **echo "Hello, world" >> test/first.txt** и перенаправления вывода:
user@desktop ~ \$ echo "Hello, world" >> test/first.txt
5. Посмотреть содержимое файла с помощью команды **cat test/first.txt**:
6. user@desktop ~ \$ cat test/first.txt
7. Hello, world
user@desktop ~ \$
8. Посмотреть расширенную информацию о каталоге, используя ключ **-l** команды **ls**.
ls -F --color -l test:
9. user@desktop ~ \$ ls -F --color -l test
10. итого 4
11. -rw-r--r-- 1 user users 13 Окт 15 20:48 first.txt
drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
12. Добавить псевдоним командной оболочки, чтобы сократить размер команды, с помощью команды **alias ls='ls -F --color -l'**:
13. user@desktop ~ \$ alias ls='ls -F --color -l'
14. user@desktop ~ \$ ls test
15. итого 4
16. -rw-r--r-- 1 user users 13 Окт 15 20:48 first.txt
drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
17. Посмотреть глубже на директорию test, используя ключи **-a** и **-i**. **ls -a -i test**
18. user@desktop ~ \$ ls -a -i test
19. итого 7
20. drwxr-xr-x 3 user users 104 Окт 15 20:49 ./
21. drwx----- 67 user users 3352 Окт 15 20:42 ../
22. -rw-r--r-- 1 user users 13 Окт 15 20:48 first.txt
drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/

- Первый столбец – номера индексных узлов файловой системы. Третий столбец – число жёстких ссылок файла.
23. Перейти в каталог test с помощью команды **cd test**
user@desktop ~ \$ cd test
 24. Скопировать файл с помощью команды **cp first.txt copy1.txt**:
 25. user@desktop test \$ cp first.txt copy1.txt
 26. user@desktop test \$ ls -a -i
 27. итого 11
 28. 5522107 drwxr-xr-x 3 user users 136 Окт 15 20:54 ./
 29. 384269 drwx----- 67 user users 3352 Окт 15 20:42 ../
 30. 5938127 -rw-r--r-- 1 user users 13 Окт 15 20:54 copy1.txt
 31. 5937954 -rw-r--r-- 1 user users 13 Окт 15 20:48 first.txt
 - 6292625 drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
 - Новый файл имеет свой собственный индексный узел.
 32. Переименовать файл с помощью команды **mv first.txt orig.txt. mv first.txt orig.txt**:
 33. user@desktop test \$ mv first.txt orig.txt
 34. user@desktop test \$ ls -a -i
 35. итого 11
 36. 5522107 drwxr-xr-x 3 user users 128 Окт 15 20:59 ./
 37. 384269 drwx----- 67 user users 3352 Окт 15 20:42 ../
 38. 5938127 -rw-r--r-- 1 user users 13 Окт 15 20:54 copy1.txt
 39. 5937954 -rw-r--r-- 1 user users 13 Окт 15 20:48 orig.txt
 - 6292625 drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
 - Отметим, что изменилось только имя файла, все остальные атрибуты остались прежними.
 40. Создать жёсткую ссылку командой **ln orig.txt copy2.txt**:
 41. user@desktop test \$ ln orig.txt copy2.txt
 42. user@desktop test \$ ls -a -i
 43. итого 15
 44. 5522107 drwxr-xr-x 3 user users 160 Окт 15 21:00 ./
 45. 384269 drwx----- 67 user users 3352 Окт 15 20:42 ../
 46. 5938127 -rw-r--r-- 1 user users 13 Окт 15 20:54 copy1.txt
 47. 5937954 -rw-r--r-- 2 user users 13 Окт 15 20:48 copy2.txt
 48. 5937954 -rw-r--r-- 2 user users 13 Окт 15 20:48 orig.txt
 - 6292625 drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
 - Добавилась ещё одна ссылка на тот же файл, число ссылок увеличилось на 1.
 49. Создать символическую ссылку командой **ln -s orig.txt orig.lnk**:
 50. user@desktop test \$ ln -s orig.txt orig.lnk
 51. user@desktop test \$ ls -a -i
 52. итого 15
 53. 5522107 drwxr-xr-x 3 user users 184 Окт 15 21:18 ./
 54. 384269 drwx----- 67 user users 3352 Окт 15 21:02 ../
 55. 5938127 -rw-r--r-- 1 user users 13 Окт 15 20:54 copy1.txt
 56. 5937954 -rw-r--r-- 2 user users 13 Окт 15 20:48 copy2.txt
 57. 5938189 lrwxrwxrwx 1 user users 8 Окт 15 21:18 orig.lnk -> orig.txt
 58. 5937954 -rw-r--r-- 2 user users 13 Окт 15 20:48 orig.txt
 - 6292625 drwxr-xr-x 2 user users 48 Окт 13 21:33 subtest/
 - Новый файл имеет новый индексный узел и размер, равный имени файла orig.txt.
 59. Сравним содержимое файлов при обращении к ним по имени:
 60. user@desktop test \$ cat orig.txt
 61. Hello, world
 62. user@desktop test \$ cat copy2.txt

63. Hello, world
64. user@desktop test \$ cat orig.lnk
Hello, world

К одним и тем же данным можно обратиться через жесткую или символическую ссылку.

Практическое занятие «Изучение аппарата прерываний. Сигналы в операционной системе UNIX»

Цель занятия: Изучение аппарата прерываний и сигналов в операционной системе UNIX.

Содержание работы:

С точки зрения пользователя получение процессом сигнала выглядит как возникновение прерывания. Процесс прекращает регулярное исполнение, и управление передается механизму обработки сигнала. По окончании обработки сигнала процесс может возобновить регулярное исполнение. Типы сигналов (их принято задавать номерами, как правило, в диапазоне от 1 до 31 включительно или специальными символьными обозначениями) и способы их возникновения в системе жестко регламентированы.

Процесс может получить сигнал от:

1. hardware (при возникновении исключительной ситуации);
2. другого процесса, выполнившего системный вызов передачи сигнала
- ;
3. операционной системы (при наступлении некоторых событий);
4. терминала (при нажатии определенной комбинации клавиш);
5. системы управления заданиями (при выполнении команды kill – мы рассмотрим ее позже).

Передачу сигналов процессу в случаях его генерации источниками 2, 3 и 5, т.е., в конечном счете, каким-либо другим процессом, можно рассматривать как реализацию в UNIX сигнальных средств связи, о которых рассказывалось в лекции 4.

Существует три варианта реакции процесса на сигнал:

1. Принудительно проигнорировать сигнал.
2. Произвести обработку по умолчанию: проигнорировать, остановить процесс (перевести в состояние ожидания до получения другого специального сигнала), либо завершить работу с образованием core файла или без него.
3. Выполнить обработку сигнала, специфицированную пользователем.

Изменить реакцию процесса на сигнал можно с помощью специальных системных вызовов, которые мы рассмотрим позже. Реакция на некоторые сигналы не допускает изменения, и они могут быть обработаны только по умолчанию. Так, например, сигнал с номером 9 – SIGKILL обрабатывается только по умолчанию и всегда приводит к завершению процесса.

Важным вопросом при программировании с использованием сигналов является вопрос о сохранении реакции на них при порождении нового процесса или замене его пользовательского контекста. При системном вызове (fork) все установленные реакции на сигналы наследуются порожденным процессом.

При системном вызове (exec) сохраняются реакции только для тех сигналов, которые игнорировались или обрабатывались по умолчанию. Получение любого сигнала, который до вызова (exec) обрабатывался пользователем, приведет к завершению процесса. Системный вызов (kill) и команда kill

Из всех перечисленных ранее в разделе " Аппаратные прерывания (interrupt), исключения (exception), программные прерывания (trap, software interrupt). Их обработка" источников сигнала пользователю доступны только два – команда kill и посылка сигнала

процессу с помощью системного вызова `kill()`. Команда `kill` обычно используется в следующей форме:

```
kill [-номер] pid
```

Здесь `pid` – это идентификатор процесса, которому посылается сигнал, а номер – номер сигнала, который посылается процессу. Послать сигнал (если у вас нет полномочий суперпользователя) можно только процессу, у которого эффективный идентификатор пользователя совпадает с идентификатором пользователя, посылающего сигнал. Если параметр – номер отсутствует, то посылается сигнал `SIGTERM`, обычно имеющий номер 15, и реакция на него по умолчанию – завершить работу процесса, который получил сигнал.

При использовании системного вызова `kill()` послать сигнал (не имея полномочий суперпользователя) можно только процессу или процессам, у которых эффективный идентификатор пользователя совпадает с эффективным идентификатором пользователя процесса, посылающего сигнал.

Практическое занятие «Изучение семейства протоколов TCP/IP»

Цель занятия: изучение инструментов конфигурирования сети в UNIX, включающих настройку параметров TCP/IP-сети.

Содержание работы:

Изучаемые команды: `arp`, `ifconfig`, `netstat`, `ping`, `route`, `ssh`, `telnet`, `traceroute`

Сценарий: Настройка сетевого интерфейса

Сценарий посвящен сетевым интерфейсам – прослойке между канальным и сетевым уровнем в UNIX. В сценарии показано, как получать информацию о настроенных в системе сетевых интерфейсах и как производится их диагностика и конфигурирование.

Начальные условия: Командная строка суперпользователя после входа в систему.

1. Получить сведения обо всех настроенных сетевых интерфейсах с помощью команды **`ifconfig -a`**:

```
2. desktop ~ # ifconfig -a
3. eth0 Link encap:Ethernet HWaddr 00:0D:60:8D:42:AA
4.  inet addr:192.168.1.5 Bcast:192.168.1.255 Mask:255.255.255.0
5.    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
6.    RX packets:204779 errors:0 dropped:0 overruns:0 frame:0
7.    TX packets:107606 errors:0 dropped:0 overruns:0 carrier:0
8.    collisions:0 txqueuelen:1000
9.    RX bytes:302429520 (288.4 Mb) TX bytes:9177476 (8.7 Mb)
10.   Base address:0x8000 Memory:c0220000-c0240000
11.
12. lo  Link encap:Local Loopback
13.   inet addr:127.0.0.1 Mask:255.0.0.0
14.   UP LOOPBACK RUNNING MTU:16436 Metric:1
15.   RX packets:228 errors:0 dropped:0 overruns:0 frame:0
16.   TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
17.   collisions:0 txqueuelen:0
```

```
RX bytes:17724 (17.3 Kb) TX bytes:17724 (17.3 Kb)
```

18. Проверить возможность соединения с локальной машиной с помощью команды **`ping 127.0.0.1`**.

```
19. desktop ~ # ping 127.0.0.1
20. PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
21. 64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.052 ms
22. 64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.051 ms
```

```
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.055 ms
```

23. Перед конфигурированием интерфейса *eth0* необходимо убедиться, что он отключен. Отключение сетевого интерфейса *eth0* производится командой **ifconfig eth0 down**.

```
24. desktop ~ # ifconfig eth0 down
25. desktop ~ # ifconfig -a
26. lo Link encap:Local Loopback
27.  inet addr:127.0.0.1 Mask:255.0.0.0
28.  UP LOOPBACK RUNNING MTU:16436 Metric:1
29.  RX packets:228 errors:0 dropped:0 overruns:0 frame:0
30.  TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
31. collisions:0 txqueuelen:0
```

```
RX bytes:17724 (17.3 Kb) TX bytes:17724 (17.3 Kb)
```

32. Для связи сетевого интерфейса *eth0* с IP-адресом 192.168.1.1 выполним команду **ifconfig eth0 192.168.1.1 up**.

```
33. desktop ~ # ifconfig eth0 192.168.1.1 up
34. desktop ~ # ifconfig -a
35. eth0 Link encap:Ethernet HWaddr 00:0C:F1:2E:0E:F9
36. inet addr:192.168.1.1 Bcast:192.168.1.255 Mask:255.255.255.0
37.  UP BROADCAST MULTICAST MTU:1500 Metric:1
38.  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
39.  TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
40. collisions:0 txqueuelen:1000
41.  RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
42.  Interrupt:11 Base address:0x2000 Memory:c0210000-c0210fff
43.
44. lo Link encap:Local Loopback
45.  inet addr:127.0.0.1 Mask:255.0.0.0
46.  UP LOOPBACK RUNNING MTU:16436 Metric:1
47.  RX packets:228 errors:0 dropped:0 overruns:0 frame:0
48.  TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
49. collisions:0 txqueuelen:0
```

```
RX bytes:17724 (17.3 Kb) TX bytes:17724 (17.3 Kb)
```

При этом по умолчанию используется сеть класса C, т.е. маска сети «255.255.255.0».

50. Для задания специфической маски подсети используется параметр *netmask*. Например, данная команда задаёт параметры сети класса A: **ifconfig eth0 10.10.1.1 netmask 255.0.0.0 up**.

```
51. desktop ~ # ifconfig eth0 10.10.1.1 netmask 255.0.0.0 up
52. desktop ~ # ifconfig -a
53. eth0 Link encap:Ethernet HWaddr 00:0C:F1:2E:0E:F9
54. inet addr:10.10.1.1 Bcast:10.255.255.255 Mask:255.0.0.0
55.  UP BROADCAST MULTICAST MTU:1500 Metric:1
56.  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
57.  TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
58. collisions:0 txqueuelen:1000
59.  RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
60.  Interrupt:11 Base address:0x2000 Memory:c0210000-c0210fff
61.
62. lo Link encap:Local Loopback
```

```

63. inet addr:127.0.0.1 Mask:255.0.0.0
64. UP LOOPBACK RUNNING MTU:16436 Metric:1
65. RX packets:228 errors:0 dropped:0 overruns:0 frame:0
66. TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
67. collisions:0 txqueuelen:0

```

```

RX bytes:17724 (17.3 Kb) TX bytes:17724 (17.3 Kb)

```

68. С помощью команды **arp** можно узнать текущую ARP-таблицу операционной системы (соответствие MAC-адресов канального уровня IP-адресам). Таблица автоматически поддерживается операционной системой в процессе сетевого обмена.

```

69. desktop ~ # arp
70. Address HWtype HWaddress Flags Mask Iface
gate.localnet ether 00:02:44:8F:16:B7 C eth0

```

Сценарий: Настройка таблицы маршрутизации

В сценарии производится изучение и настройка таблицы маршрутизации IP. С помощью специальной программы производится изучение маршрута следования пакетов.

Начальные условия: Командная строка суперпользователя, сетевой интерфейс настроен на статический IP-адрес.

1. Для просмотра таблицы маршрутизации воспользуемся командой **route -n**:

```

2. desktop ~ # route -n
3. Kernel IP routing table
4. Destination Gateway Genmask Flags Metric Ref Use Iface
5. 10.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 eth0
6. 127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo

```

```

0.0.0.0 10.10.1.254 0.0.0.0 UG 0 0 0 eth0

```

7. Без использования ключа **-n** для всех имён будут использоваться символичные значения: **route**

```

8. desktop ~ # route
9. Kernel IP routing table
10. Destination Gateway Genmask Flags Metric Ref Use Iface
11. localnet * 255.0.0.0 U 0 0 0 eth0
12. loopback * 255.0.0.0 U 0 0 0 lo

```

```

default gate.localnet 0.0.0.0 UG 0 0 0 eth0

```

13. Для добавления новой строки в таблицу нужно воспользоваться параметром **add**: **route add -host 10.10.2.1 dev eth0**.

```

14. desktop ~ # route add -host 10.10.2.1 dev eth0
15. desktop ~ # route -n
16. Kernel IP routing table
17. Destination Gateway Genmask Flags Metric Ref Use Iface
18. 10.10.2.1 0.0.0.0 255.255.255.255 UH 0 0 0 eth0
19. 10.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 eth0
20. 127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo

```

```

0.0.0.0 10.10.1.254 0.0.0.0 UG 0 0 0 eth0

```

Эта команда добавляет явный маршрут до отдельного хоста с указанным IP-адресом через интерфейс **eth0**.

21. Аналогичным образом маршрут удаляется, используется параметр *del*: **route del -host 10.10.2.1**.

```
22. desktop ~ # route add -host 10.10.2.1 dev eth0
23. desktop ~ # route -n
24. Kernel IP routing table
25. Destination Gateway Genmask Flags Metric Ref Use Iface
26. 10.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 eth0
27. 127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
```

```
0.0.0.0 10.10.1.254 0.0.0.0 UG 0 0 0 eth0
```

28. В качестве назначения маршрута можно указывать также целую сеть (параметр - *net*. Рассмотрим команду, которая задаёт маршрут в сеть «192.168.1.0» через шлюз «10.10.1.253»: **route add -net 192.168.1.0 gw 10.10.1.253**.

```
29. desktop ~ # route add -net 192.168.1.0 gw 10.10.1.253
30. desktop ~ # route -n
31. Kernel IP routing table
32. Destination Gateway Genmask Flags Metric Ref Use Iface
33. 192.168.1.0 10.10.1.253 255.255.255.255 UG 0 0 0 eth0
34. 10.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 eth0
35. 127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
```

```
0.0.0.0 10.10.1.254 0.0.0.0 UG 0 0 0 eth0
```

36. Рассмотрим простой маршрут движения пакетов до хоста в Internet с помощью программы **traceroute ya.ru**:

```
37. desktop ~ # traceroute ya.ru
38. traceroute to ya.ru (213.180.204.8), 64 hops max, 40 byte packets
39. 1 10.10.1.254 (10.10.1.254) 3.418 ms 2.67 ms 0.719 ms
40. 2 cs7206.rinet.ru (195.54.192.28) 1.34 ms 1.378 ms 0.647 ms
41. 3 ix2-m9.yandex.net (193.232.244.93) 1.554 ms 1.457 ms 1.420 ms
42. 4 c3-vlan4.yandex.net (213.180.210.146) 2.137 ms 2.154 ms 1.842 ms
```

```
5 ya.ru (213.180.204.8) 2.646 ms 2.183 ms 2.220 ms
```

Сценарий: Изучение службы доменных имён

Сценарий посвящен изучению службы доменных имён – её использованию и конфигурированию.

Начальные условия: Командная строка суперпользователя, сетевой интерфейс настроен на статический IP-адрес.

1. Просмотрим содержимое файла /etc/hosts, содержащего имена локальных хостов: **cat /etc/hosts**

```
2. desktop ~ # cat /etc/hosts
3. #
4. # hosts This file describes a number of hostname-to-address
5. # mappings for the TCP/IP subsystem. It is mostly
6. # used at boot time, when no name servers are running.
7. # On small systems, this file can be used instead of a
8. # "named" name server. Just add the names, addresses
9. # and any aliases to this file...
10. #
11.
12. # Localhost
```

```
13. 127.0.0.1 localhost
14.
15. # Home LAN
16. 10.10.1.254 gate.localnet gate
```

```
10.10.1.20 boss.localnet boss
```

17. Проверим работоспособность DNS с помощью команды обращения к хосту в Internet по имени **ping ya.ru**:

```
18. desktop ~ # ping ya.ru
19. PING ya.ru (213.180.204.8) 56(84) bytes of data.
20. 64 bytes from ya.ru (213.180.204.8): icmp_seq=1 ttl=54 time=3.56 ms
```

```
64 bytes from ya.ru (213.180.204.8): icmp_seq=2 ttl=54 time=2.22 ms
```

21. Для корректной работы службы доменных имен необходимо прописать используемые серверы DNS в файле `/etc/resolv.conf`. Просмотрим его содержимое **cat /etc/resolv.conf**:

```
22. desktop ~ # cat /etc/resolv.conf
23. domain localnet
```

```
nameserver 10.10.1.17
```

24. С помощью команды **host ya.ru** узнаем информацию DNS о хосте в Internet:

```
25. desktop ~ # host ya.ru
26. ya.ru has address 213.180.204.8
```

```
ya.ru mail is handled by 10 cmail.yandex.ru.
```

27. Вторым аргументом команды **host ya.ru ns1.yandex.ru** имя DNS-сервера, с которого необходимо получить информацию:

```
28. desktop ~ # host ya.ru ns1.yandex.ru
29. ya.ru has address 213.180.204.8
30. Using domain server:
31. Name: ns1.yandex.ru
32. Address: 213.180.193.1#53
33. Aliases:
34.
35. Using domain server:
36. Name: ns1.yandex.ru
37. Address: 213.180.193.1#53
38. Aliases:
39.
```

```
ya.ru mail is handled by 10 cmail.yandex.ru.
```

Сценарий: Простая диагностика работы сети

Сценарий рассматривает самые простые способы диагностики работы сети.

Начальные условия: Командная строка суперпользователя, сетевой интерфейс настроен на статический IP-адрес.

1. Для проверки работоспособности сетевых служб воспользуемся командой удалённого терминала: **telnet ya.ru 80**. В данном случае будет установлено соединение с хостом в Internet по порту 80 (HTTP):

```
2. desktop ~ # telnet ya.ru 80
3. Trying 213.180.204.8...
```

```

4. Connected to ya.ru.
5. Escape character is '^]'.
6. GET / HTTP/1.0
7.
8. HTTP/1.0 200 OK
9. Server: thttpd/2.25b 29dec2003
10. Content-Type: text/html; charset=windows-1251
11. Date: Wed, 23 Nov 2005 05:40:33 GMT
12. Last-Modified: Mon, 07 Nov 2005 15:13:14 GMT
13. Accept-Ranges: bytes
14. Connection: close
15. Content-Length: 2005
16.
17. <html>
18. <head>

```

...

19. Если во время соединения с удалённым узлом ввести команду **netstat -t**, то можно увидеть, что состояние этого соединения – «ESTABLISHED»:

```

20. desktop ~ # netstat -t
21. Active Internet connections (servers and established)

```

```

tcp 0 0 desktop:42639 ya.ru:http ESTABLISHED

```

22. Информацию обо всех соединениях в системе можно получить с помощью команды **netstat -a**. В этом случае будет выводиться информация обо всех TCP-, UDP- и локальных сокетах:

```

23. desktop ~ # netstat -a
24. Active Internet connections (servers and established)
25. Proto Recv-Q Send-Q Local Address Foreign Address State
26. tcp 0 0 *:32769 *:.* LISTEN
27. tcp 0 0 *:32770 *:.* LISTEN
28. tcp 0 0 *:sunrpc *:.* LISTEN
29. tcp 0 0 *:ssh *:.* LISTEN
30. tcp 0 0 desktop:42639 ya.ru:http ESTABLISHED
31. udp 0 0 *:32768 *:.*
32. udp 0 0 *:32769 *:.*
33. udp 0 0 *:sunrpc *:.*
34. Active UNIX domain sockets (servers and established)
35. Proto RefCnt Flags Type State I-Node Path
36. unix 2 [ ACC ] STREAM LISTENING 8344 /var/run/acpid.socket
37. unix 2 [ ACC ] STREAM LISTENING 8866 /var/run/sdp

```

...

Сценарий: Работа по удалённому терминалу

Сценарий рассматривает работу по удалённому сетевому терминалу с использованием программы ssh.

Начальные условия: Командная строка суперпользователя, сетевой интерфейс настроен на статический IP-адрес.

1. С помощью команды **ssh user@10.10.1.222**

```

2. desktop ~ # ssh user@10.10.1.222
3. Password:

```

```
4. Last login: Sat Nov 21 15:56:20 2005 from 10.10.1.5
```

```
5. user@remote ~ $
```

```
user@remote ~ $ exit
```

6. Выполним команду **who**, чтобы убедиться, что находимся на удалённой машине. Для всех пользователей, работающих удалённо, указывается IP-адрес.

```
7. user@remote ~ $ who
```

```
8. user vc/1 Nov 14 14:04
```

```
user pts/0 Nov 22 10:55 (10.10.1.5)
```

9. Для завершения сеанса удалённого терминала нужно выйти из командной оболочки с помощью команды **exit**.

```
10. user@remote ~ $ exit
```

```
11. logout
```

```
12. Connection to 10.10.1.5 closed.
```

```
desktop ~ #
```

Задания для самоподготовки

1. Настройте сетевой интерфейс eth0 на сеть с адресом 192.168.77.0, состоящую из 16 машин.
2. С помощью программы netstat определите какие значения принимают локальные порты в исходящих TCP-соединениях? В какой диапазон они попадают?

Практическое занятие «Изучение сокетов в UNIX и работа с ними»

Цель занятия: Изучение сокетов.

Содержание работы:

Для создания *сокета* в операционной системе служит *системный вызов* `socket()`. Для транспортных протоколов *семейства TCP/IP* существует два вида *сокетов*: *UDP-сокеты* – *сокеты* для работы с *датаграммами*, и *TCP-сокеты* – *поточный сокет*. Однако понятие *сокета* (см. лекцию 14, раздел "Полные адреса. Понятие *сокета (socket)*") не ограничивается рамками только этого семейства протоколов. Рассматриваемый *интерфейс* сетевых системных вызовов (`socket()`, `bind()`, `recvfrom()`, `sendto()` и т. д.) в операционной системе UNIX может применяться и для других стеков протоколов (и для протоколов, лежащих ниже *транспортного уровня*).

При создании *сокета* необходимо точно специфицировать его тип. Эта спецификация производится с помощью трех параметров вызова `socket()`. Первый *параметр* указывает, к какому семейству протоколов относится создаваемый *сокеты*, а второй и третий параметры определяют конкретный протокол внутри данного семейства.

Второй *параметр* служит для задания вида интерфейса работы с *сокеты* – будет это *поточный сокет*, *сокеты* для работы с *датаграммами* или какой-либо иной. Третий *параметр* указывает протокол для заданного типа интерфейса. В *стеке протоколов TCP/IP* существует только один протокол для *поточных сокетов* – *TCP* и только один протокол для *датаграммных сокетов* – *UDP*, поэтому для транспортных протоколов *TCP/IP* третий *параметр* игнорируется.

В других стеках протоколов может быть несколько протоколов с одинаковым видом интерфейса, например, *датаграммных*, различающихся по степени надежности.

Для транспортных протоколов *TCP/IP* мы всегда в качестве первого параметра будем указывать предопределенную константу `AF_INET` (*Address family – Internet*) или

ее синоним PF_INET (*Protocolfamily – Internet*).

Второй *параметр* будет принимать предопределенные значения SOCK_STREAM для потоковых *сокетов* и SOCK_DGRAM – для датаграммных.

Поскольку третий *параметр* в нашем случае не учитывается, в него мы будем подставлять значение 0.

Ссылка на информацию о созданном *сокете* помещается в *таблицу открытых файлов процесса* подобно тому, как это делалось для *pip*'ов и *FIFO* (см. семинар 5). Системный вызов возвращает пользователю файловый *дескриптор*, соответствующий заполненному элементу таблицы, который далее мы будем называть дескриптором *сокета*. Такой способ хранения информации о *сокете* позволяет, во-первых, процессам-детям наследовать ее от процессов-родителей, а, во-вторых, использовать для *сокетов* часть системных вызовов, которые уже знакомы нам по работе с *pip*'ами и *FIFO*: close(), read(), write().

Системный вызов для создания сокета

Прототип системного вызова

```
#include <sys/types.h>
#include <sys/socket.h>
int socket(int domain, int type,
int protocol);
```

Описание системного вызова

Системный вызов *socket* служит для создания виртуального коммуникационного узла в операционной системе. Данное описание не является полным описанием системного вызова, а предназначено только для использования в нашем курсе. За полной информацией обращайтесь к UNIX Manual.

Параметр *domain* определяет семейство протоколов, в рамках которого будет осуществляться передача информации. Мы рассмотрим только два таких семейства из нескольких существующих. Для них имеются предопределенные значения параметра:

- PF_INET – для *семейства протоколов TCP/IP* ;
- PF_UNIX – для семейства внутренних протоколов UNIX, иначе называемого еще UNIX domain.

Параметр *type* определяет семантику обмена информацией: будет ли осуществляться связь через сообщения (datagrams), с помощью установления *виртуального соединения* или еще каким-либо способом. Мы будем пользоваться только двумя способами обмена информацией с предопределенными значениями для параметра *type*:

- SOCK_STREAM – для связи с помощью установления *виртуального соединения* ;
- SOCK_DGRAM – для обмена информацией через сообщения.

Параметр *protocol* специфицирует конкретный протокол для выбранного семейства протоколов и способа обмена информацией. Он имеет значение только в том случае, когда таких протоколов существует несколько. В нашем случае семейство протоколов и тип обмена информацией определяют протокол однозначно. Поэтому данный параметр мы будем полагать равным 0.

Возвращаемое значение

В случае успешного завершения системный вызов возвращает файловый дескриптор (значение большее или равное 0), который будет использоваться как ссылка на созданный коммуникационный узел при всех дальнейших сетевых вызовах. При возникновении какой-либо ошибки возвращается отрицательное значение.

Адреса сокетов. Настройка адреса сокета. Системный вызов bind()

Когда *сокет* создан, необходимо настроить его *адрес*. Для этого используется системный вызов *bind()* . Первый *параметр* вызова должен содержать *дескриптор сокета*, для которого производится настройка адреса. Второй и третий параметры задают этот *адрес*.

Во втором параметре должен быть *указатель* на структуру structsockaddr, содержащую удаленную и локальные части полного адреса.

Указатели типа structsockaddr * встречаются во многих сетевых системных вызовах; они используются для передачи информации о том, к какому адресу привязан или должен быть привязан *сокет*. Рассмотрим этот *тип данных* подробнее. Структура structsockaddr описана в файле <sys/socket.h> следующим образом:

```
structsockaddr {  
    shortsa_family;  
    charsa_data[14];  
};
```

Такой состав структуры обусловлен тем, что сетевые системные вызовы могут применяться для различных семейств протоколов, которые *по-разному* определяют адресные пространства для удаленных и локальных *адресов сокета*. По сути дела, этот *тип данных* представляет собой лишь общий *шаблон* для передачи системным вызовам структур данных, специфических для каждого семейства протоколов. Общим элементом этих структур остается только *поле* shortsa_family (которое в разных структурах, естественно, может иметь разные имена, важно лишь, чтобы все они были одного типа и были первыми элементами своих структур) для описания семейства протоколов. Содержимое этого поля *системный вызов* анализирует для точного определения состава поступившей информации.

Системный вызов для привязки сокета к конкретному адресу

Прототип системного вызова

```
#include<sys/types.h>  
#include <sys/socket.h>  
int bind(intsockd,  
structsockaddr *my_addr,  
intaddrlen);
```

Описание системного вызова

Системный вызов *bind* служит для привязки созданного *сокета* к определенному полному адресу вычислительной сети.

Параметр sockd является дескриптором созданного ранее коммуникационного узла, т. е. значением, которое вернул системный вызов *socket()*.

Параметр my_addr представляет собой адрес структуры, содержащей информацию о том, куда именно мы хотим привязать наш *сокет*— то, что принято называть *адресом сокета*. Он имеет тип указателя на структуру-шаблон struct sockaddr, которая должна быть конкретизирована в зависимости от используемого семейства протоколов и заполнена перед вызовом.

Параметр addrlen должен содержать фактическую длину структуры, адрес которой передается в качестве второго параметра. Эта длина в разных семействах протоколов и даже в пределах одного семейства протоколов может быть различной (например, для UNIX Domain).

Возвращаемое значение

Системный вызов возвращает значение 0 при нормальном завершении и отрицательное значение — в случае ошибки.

Системные вызовы sendto() и recvfrom()

Для отправки *датаграмм* применяется *системный вызов sendto()*. В число параметров этого вызова входят:

- дескриптор *сокета*, через который отсылается *датаграмма* ;
- адрес области памяти, где лежат данные, которые должны составить содержательную часть *датаграммы*, и их длина;
- флаги, определяющие поведение системного вызова (в нашем случае они всегда будут иметь значение 0);

- указатель на структуру, содержащую *адрес сокета* получателя, и ее фактическая длина. *Системный вызов* возвращает отрицательное значение при возникновении ошибки и количество реально отосланных *байт* при нормальной работе. **Нормальное завершение системного вызова не означает, что датаграмма уже покинула ваш компьютер!** Датаграмма сначала помещается в системный сетевой *буфер*, а ее реальная *отправка* может произойти после возврата из системного вызова. *Вызов sendto()* может блокироваться, если в сетевом буфере не хватает места для *датаграммы*. Для чтения принятых *датаграмм* и определения адреса получателя (при необходимости) служит *системный вызов recvfrom()*. В число параметров этого вызова входят:
- Дескриптор *сокета*, через который принимается *датаграмма*.
- Адрес области памяти, куда следует положить данные, составляющие содержательную часть *датаграммы*.
- Максимальная длина, допустимая для *датаграммы*. Если количество данных *датаграммы* превышает заданную максимальную длину, то вызов по умолчанию рассматривает это как ошибочную ситуацию.
- Флаги, определяющие поведение системного вызова (в нашем случае они будут полагаться равными 0).
- Указатель на структуру, в которую при необходимости может быть занесен *адрес сокета* отправителя. Если этот адрес не требуется, то можно указать значение NULL.
- Указатель на переменную, содержащую максимально возможную длину адреса отправителя. После возвращения из системного вызова в нее будет занесена фактическая длина структуры, содержащей адрес отправителя. Если предыдущий параметр имеет значение NULL, то и этот параметр может иметь значение NULL.

Системный вызов recvfrom() по умолчанию блокируется, если отсутствуют принятые *датаграммы*, до тех пор, пока *датаграмма* не появится. При возникновении ошибки он возвращает отрицательное значение, при нормальной работе – длину принятой *датаграммы*.

Наберите и откомпилируйте программу. Перед запуском "**узнайте у своего системного администратора**", запущен ли в системе стандартный *UDP*-сервис *echo* и если нет, попросите стартовать его. Запустите программу с запросом к сервису своего компьютера, к сервисам других компьютеров. Если в качестве *IP-адреса* указать несуществующий *адрес*, *адрес* выключенной машины или машины, на которой не работает сервис *echo*, то *программа* бесконечно блокируется в вызове *recvfrom()*, ожидая ответа. *Протокол UDP* не является надежным протоколом.

Если *датаграмму* доставить по назначению не удалось, то отправитель никогда об этом не узнает!

Пример программы *UDP*-сервера

Поскольку *UDP*-сервер использует те же самые системные вызовы, что и *UDP*-клиент, мы можем сразу приступить к рассмотрению примера *UDP*-сервера (*программа 15–16-2.с*) для сервиса *echo*.

/* Простой пример *UDP*-сервера для сервиса *echo* */

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <string.h>
#include <stdio.h>
#include <errno.h>
#include <unistd.h>
int main()
{
    int sockfd; /* Дескриптор сокета */
```

```

int clilen, n; /* Переменные для различных длин
и количества символов */
char line[1000]; /* Массив для принятой и
отсылаемой строки */
struct sockaddr_in servaddr, cliaddr; /* Структуры
для адресов сервера и клиента */
/* Заполняем структуру для адреса сервера: семейство
протоколов TCP/IP, сетевой интерфейс – любой, номер порта
51000. Поскольку в структуре содержится дополнительное не
нужное нам поле, которое должно быть нулевым, перед
заполнением обнуляем ее всю */
bzero(&servaddr, sizeof(servaddr));
servaddr.sin_family = AF_INET;
servaddr.sin_port = htons(51000);
servaddr.sin_addr.s_addr = htonl(INADDR_ANY);
/* Создаем UDP сокет */
if((sockfd = socket(PF_INET, SOCK_DGRAM, 0)) < 0){
perror(NULL); /* Печатаем сообщение об ошибке */
exit(1);
}
/* Настраиваем адрес сокета */
if(bind(sockfd, (struct sockaddr *)&servaddr,
sizeof(servaddr)) < 0){
perror(NULL);
close(sockfd);
exit(1);
}
while(1) {
/* Основной цикл обслуживания */
/* В переменную clilen заносим максимальную длину
для ожидаемого адреса клиента */
clilen = sizeof(cliaddr);
/* Ожидаем прихода запроса от клиента и читаем его.
Максимальная допустимая длина датаграммы – 999
символов, адрес отправителя помещаем в структуру
cliaddr, его реальная длина будет занесена в
переменную clilen */
if((n = recvfrom(sockfd, line, 999, 0,
(struct sockaddr *)&cliaddr, &clilen)) < 0){
perror(NULL);
close(sockfd);
exit(1);
}
/* Печатаем принятый текст на экране */
printf("%s\n", line);
/* Принятый текст отправляем обратно по адресу
отправителя */
if(sendto(sockfd, line, strlen(line), 0,
(struct sockaddr *)&cliaddr, clilen) < 0){
perror(NULL);
close(sockfd);
exit(1);
}
}

```



```

} /* Уходим ожидать новую датаграмму*/
}
return 0;
}

```

Листинг 15-16.2. Программа 15–16-2.с . Простой пример UDP-сервера для сервиса echo. Наберите и откомпилируйте программу. Запустите ее на выполнение. Модифицируйте текст программы *UDP*-клиента (программа 15–16-1.с), заменив номер *порта* с 7 на 51000. Запустите клиента с другого виртуального терминала или с другого компьютера и убедитесь, что клиент и *сервер* взаимодействуют корректно.

Практическое занятие «Настройка сетевых параметров»

Цель занятия: Настройка сетевых параметров.

Содержание работы:

Проверьте работоспособность стека протоколов TCP/IP.

Запустите виртуальную машину VM-1 и загрузите ОС Windows.

Запустите консоль (*Пуск/Программы/Стандартные/Командная строка*).

В командной строке введите `ipconfig /all / more`.

Используя приведенную ниже информацию, создайте в своей папке текстовый документ со следующими данными:

- имя компьютера;
- основной DNS-суффикс;
- описание DNS-суффикса для подключения;
- физический адрес;
- DHCP включен;
- автоконфигурация включена;
- IP-адрес автоконфигурации;
- маска подсети;
- шлюз по умолчанию.

Убедитесь в работоспособности стека *TCP/IP*, отправив эхо-запросы на IP-адреса. Для этого воспользуйтесь командой `ping`:

- отправьте эхо-запросы на локальный адрес компьютера (loopback) `ping 127.0.0.1` (на экране должны появиться сообщения о полученном ответе от узла 127.0.0.1);
- отправьте эхо-запрос по другому IP-адресу, например 172.21.5.1.

2. Настройте стек протоколов TCP/IP для использования статического IP-адреса.

Откройте окно Сетевые подключения (*Пуск/Панель управления/Сетевые подключения*). Вызовите свойства подключения по локальной сети. Для этого можно воспользоваться контекстным меню.

В появившемся диалоговом окне на вкладке Общие откройте свойства Протокол Интернета TCP/IP.

Щелкните переключатель *Использовать следующий IP-адрес* и введите в соответствующие поля данные: IP_адрес; Маску подсети; Основной шлюз; Предпочитаемый DNS.

Примените параметры кнопкой *ОК*.

Закройте окно свойств подключения кнопкой *ОК* (если потребуется, то согласитесь на перезагрузку компьютера).

Проверьте работоспособность стека протоколов *TCP/IP*.

3. Настройте TCP/IP для автоматического получения IP-адреса.

Откройте окно Сетевые подключения.

Вызовите свойства Подключения по локальной сети.

Откройте свойства Протокол Интернета TCP/IP.

Установите переключатель *Получить IP-адрес автоматически*.

Закройте диалоговое окно Свойства: Протокол Интернета TCP/IP кнопкой *OK*.

Примените параметры кнопкой *OK*.

Проверьте настройку стека протоколов *TCP/IP*.

Получите другой адрес для своего компьютера. Для этого:

- запустите консоль (командную строку);
- введите команду для сброса назначенных адресов - `ipconfig /release`;
- введите команду для получения нового адреса `ipconfig / renew`;

Проверьте работоспособность стека протоколов *TCP/IP*.

Контрольные вопросы:

1. Опишите параметры, используемые при настройке статического адреса TCP/IP.
2. Какие преимущества дает применение стека протоколов TCP/IP .
3. Дайте определение понятию стек протоколов TCP/IP.

Практическое занятие «Управление разделением ресурсов в локальной сети»

Цель занятия: Управление разделением ресурсов в сети.

Содержание работы:

1. Загрузить операционную систему Windows 2000 Server.
2. Запустить **VMware** и загрузить одну виртуальную машину с ОС Windows XP. Убедиться, что оба компьютера доступны по сети (при помощи утилиты **ping**) и находятся в одной рабочей группе. Если этого не произошло, то проверить настройку сетевых интерфейсов и сетевую идентификацию обеих машин.
3. На виртуальной машине открыть оснастку **Локальные пользователи и группы** (Путь: **Пуск – Настройки – Панель управления – Администрирование – Управление компьютером**) (рис. 3.1)

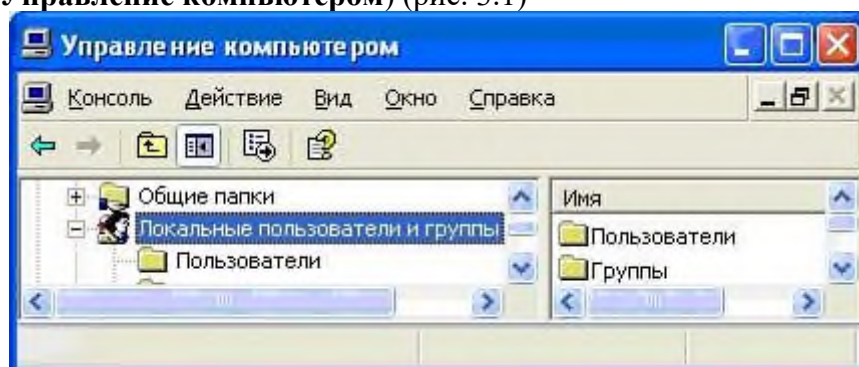


Рис. 3.1. Оснастка **Локальные пользователи и группы**

4. Познакомиться с описанием встроенных групп пользователей.
5. Создать группу пользователей **Моя группа**, в которую включить следующих пользователей: себя и двух студентов-соседей своей подгруппы (рис. 3.2 ... 3.5), а также создать группу пользователей **Преподаватели**, в которую включить пользователя **Гришин**. Всем пользователям назначить пароль **123** с неограниченным сроком действия.

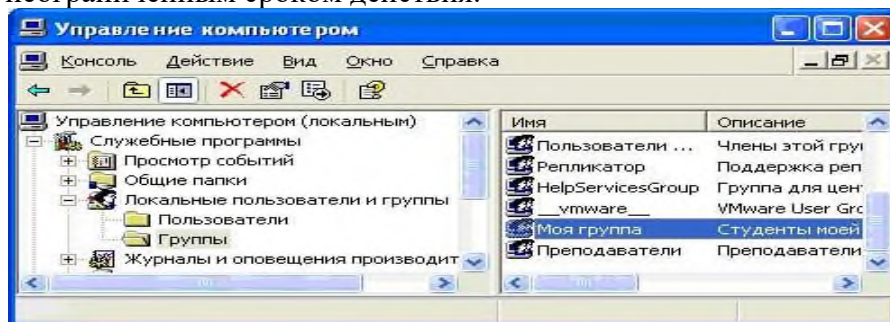


Рис. 3.2. Создание групп пользователей

Новый пользователь

Пользователь: Иванов

Полное имя: Иванов Иван Иванович

Описание: студент

Пароль: ...

Подтверждение: ...

☐ Потребовать смену пароля при следующем входе в систему

☐ Запретить смену пароля пользователем

☒ Срок действия пароля не ограничен

☐ Отключить учетную запись

Создать Закреть

Рис. 3.3. Создание учетной записи пользователя **Иванов** с неограниченным сроком действия пароля

Свойства: Иванов

Общие Членство в группах Профиль

Член групп:

Пользователи

Добавить... Удалить

ОК Отмена Применить

Рис. 3.4. Свойства новой учетной записи пользователя по умолчанию

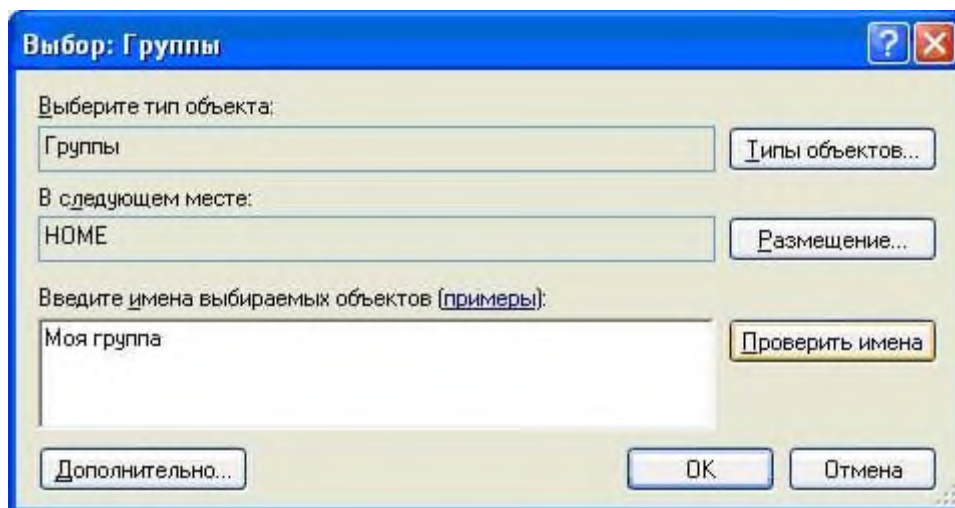


Рис. 3.5. Пример включения пользователя **Иванов** в группу **Моя группа**

6. Результаты создания пользователей продемонстрировать преподавателю – каждый новый пользователь должен быть членом двух групп: **Пользователи** и **Моя группа** (или **Преподаватели**).

Изменение параметров входа пользователей в систему

7. Завершить сеанс текущего пользователя с именем **student**, который является членом группы **Администраторы** и имеет пустой пароль. Убедиться, что в окне приветствия Windows XP появился список всех пользователей, кроме пользователя **Администратор**. Новые пользователи имеют имена, которые были введены в поле **Полное имя** в свойствах соответствующей учетной записи. Если это поле осталось пустым, то отображается **Имя пользователя** – логин. Вновь зайти в систему пользователем **student**.
8. Убрать окно приветствия для осуществления стандартного входа в систему с вводом имени пользователя и пароля (**Панель управления – Учетные записи пользователей – Изменение входа пользователей в систему** – снять галочку **Использовать страницу приветствия**) (рис. 3.6) и проверить результат – выйти из системы и вновь зайти пользователем **student**. **Внимание!** Нажатие клавиш **Ctrl + Alt + Delet** вызывает реакцию как основной, так и гостевой ОС, поэтому в виртуальной машине следует пользоваться комбинацией **Ctrl + Alt + Insert**.

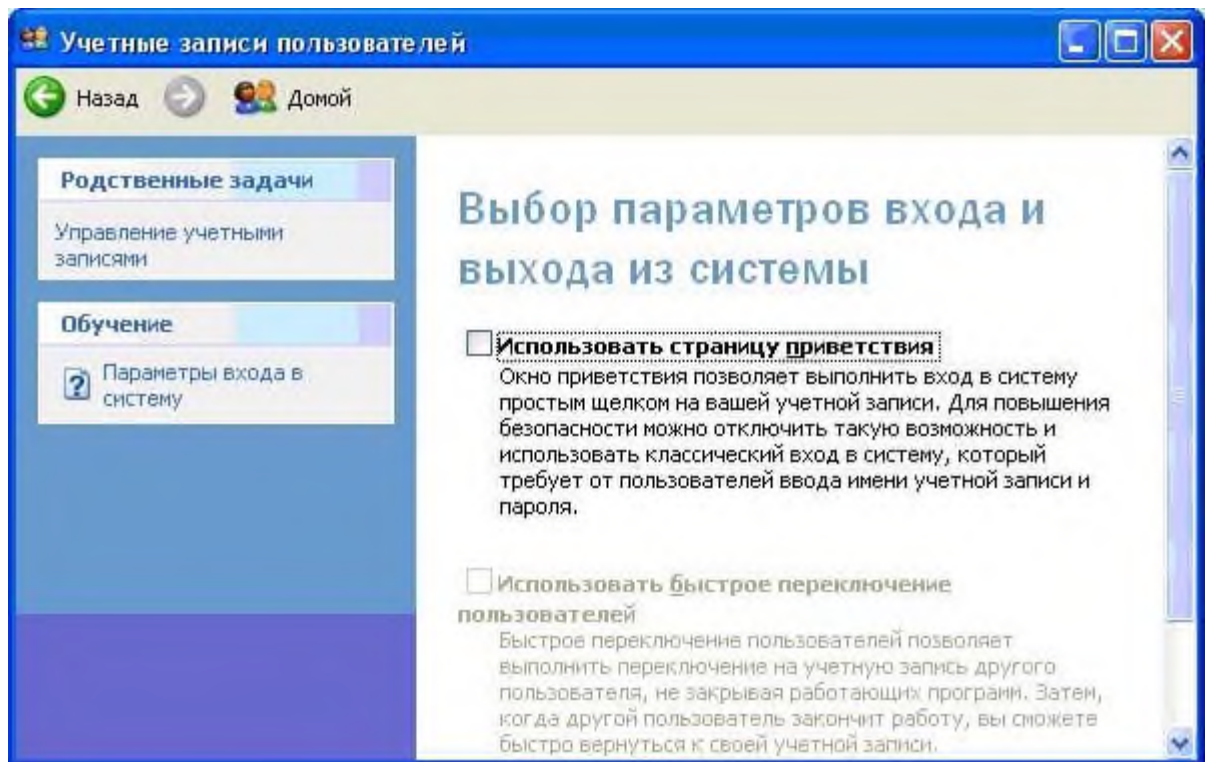


Рис. 3.6. Окно выбора параметров входа в систему

Назначение разрешений на разделяемые файловые ресурсы

9. На логическом диске с файловой системой NTFS создать папку **Контрольные работы**, в свойствах которой открыть закладку **Безопасность**. Если такой вкладки нет, то в свойствах папки (**Сервис – Свойства папки – Вид**) убрать флажок у опции **Использовать простой общий доступ к файлам** (рис. 3.7).

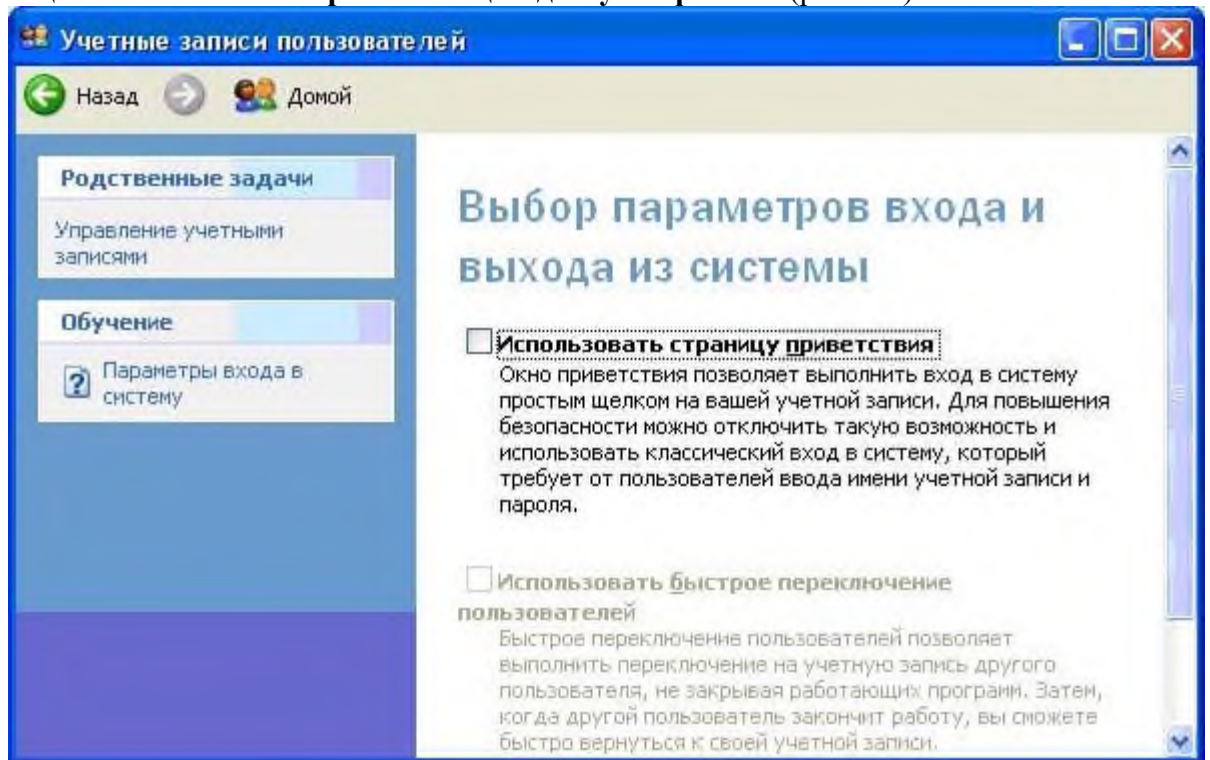


Рис. 3.7. Включение доступа к параметрам безопасности объектов файловой системы NTFS – снять флажок у опции **Использовать простой общий доступ к файлам**

10. Для папки **Контрольные работы** назначить следующие разрешения:
 - администраторам разрешен полный доступ;

- членам группы **Моя группа** разрешено размещать свои файлы и при необходимости исправлять их, но они не должны иметь возможности читать чужие контрольные работы и удалять файлы (в том числе и свои);
- членам группы **Преподаватели** разрешено только читать все файлы.
- Самостоятельно назначить требуемые разрешения и лишь в случае неудачи сверить свои настройки с рис. 3.8 ... 3.10, которые иллюстрируют ход выполнения задания.

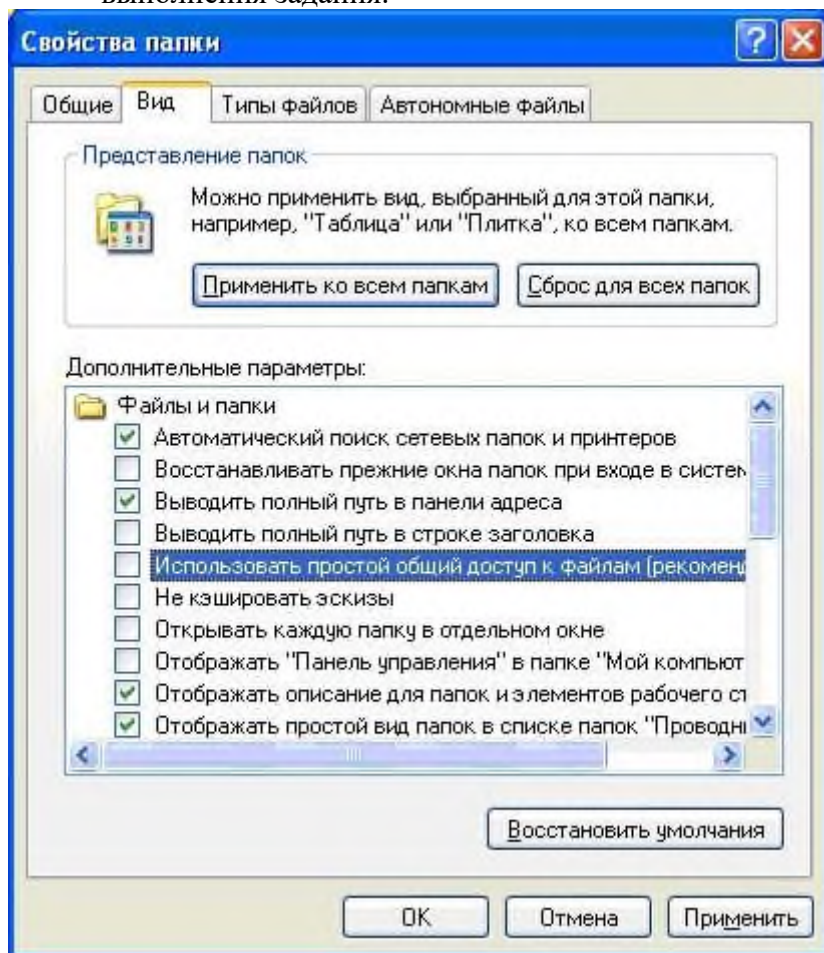


Рис. 3.8. Состав допущенных к каталогу групп пользователей и набор основных разрешений для группы **Преподаватели**

Для получения возможности изменять разрешения необходимо при помощи кнопки **Дополнительно** снять флажок **Наследовать от родительского объекта ...** (рис. 3.9)

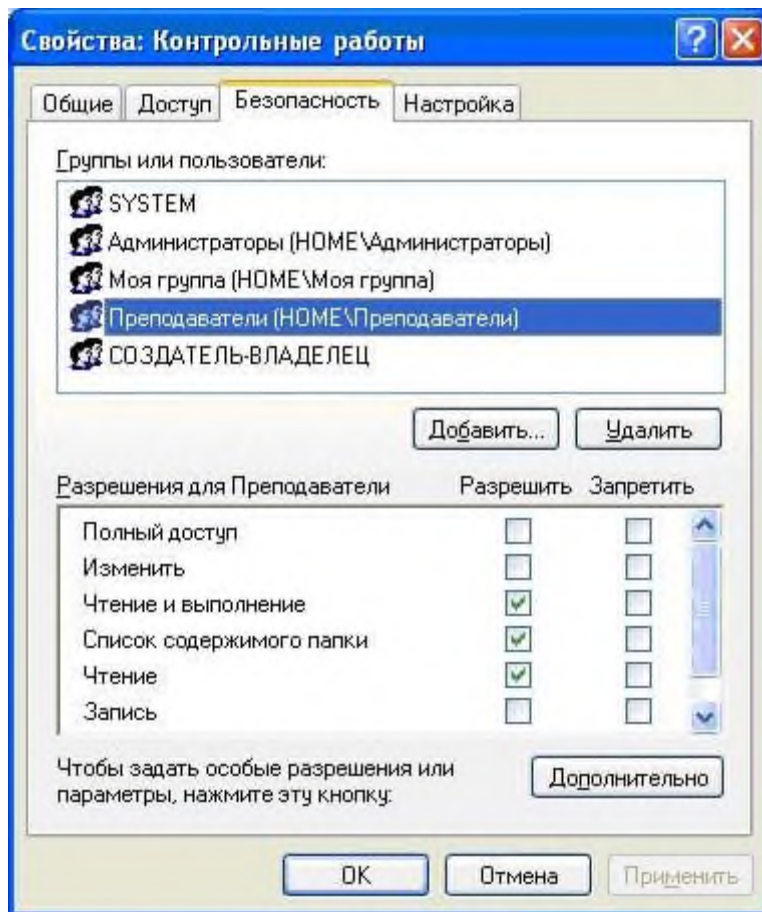


Рис. 3.9. Интерфейс оснастки управления индивидуальными разрешениями

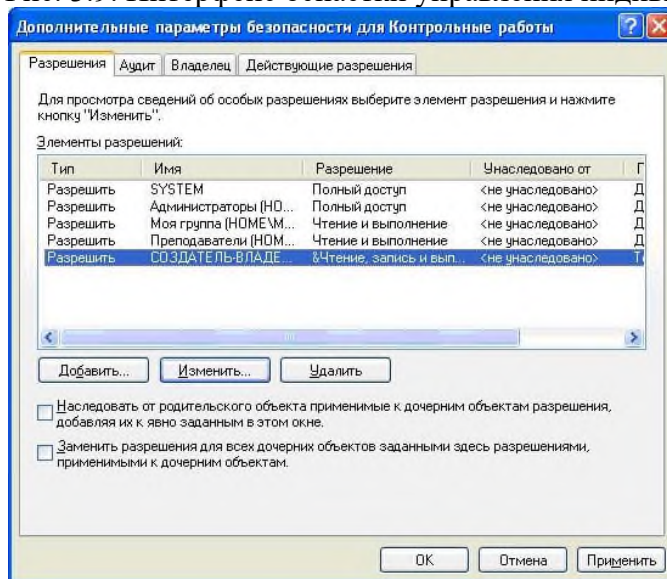


Рис. 3.10. Набор разрешений для группы **Моя группа** (только для этой папки!!!)

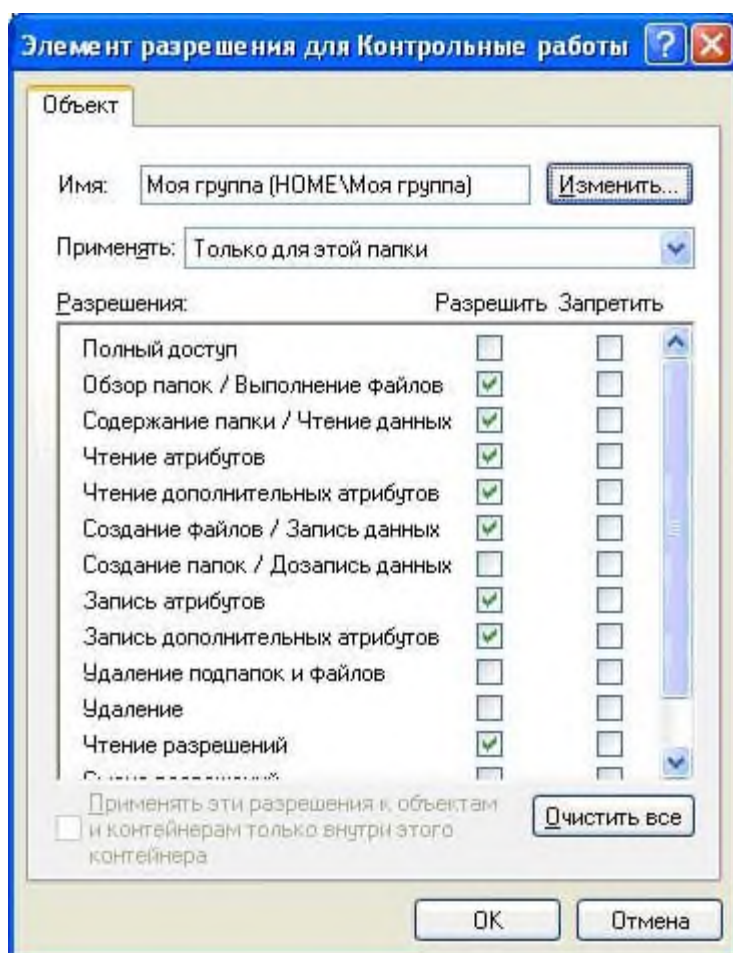


Рис. 3.11. Набор разрешений для группы **СОЗДАТЕЛЬ-ВЛАДЕЛЕЦ** (для этой папки и ее файлов!!!)

Проверка выполнения разрешений

11. По очереди зайти в систему созданными пользователями и от своего имени и имени обоих соседей создать на рабочем столе текстовые документы (имя файлу документа присвоить по фамилии пользователя) и поместить их в папку **Контрольные работы**. Проверить выполнение всех созданных разрешений на папку **Контрольные работы**, заходя в систему от имени различных пользователей.
12. Открыть к папке **Контрольные работы** доступ по сети, для этого в свойствах папки **открыть общий доступ** по сети (рис. 3.12).

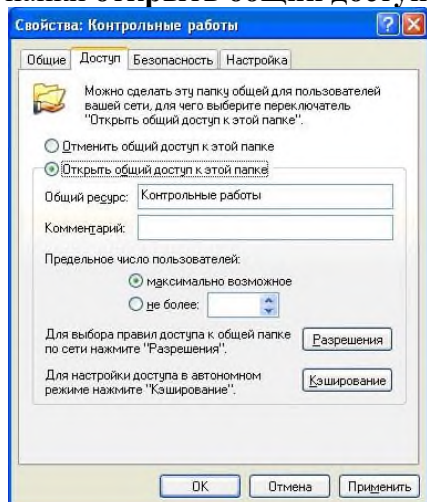


Рис. 3.12. Открытие общего доступа к папке по сети

13. Проверить доступность по сети виртуального компьютера соседа, на котором он создал аналогичную папку **Контрольные работы**, доступную в сети. Убедиться,

что на соседнем компьютере доступ к этой папке по сети имеют только те пользователи, которые обладают на соседнем компьютере аналогичными учетными записями.

Практическое занятие «Восстановление файловой системы»

Цели занятия: научиться выполнять восстановление файловой системы.

Содержание работы:

1. В программе **WinWord** выберите пункты меню “Файл” – “Сохранить как”.
2. В открывшемся окне в поле “Тип файла” выберите “Текст в формате RTF” (рис.1).

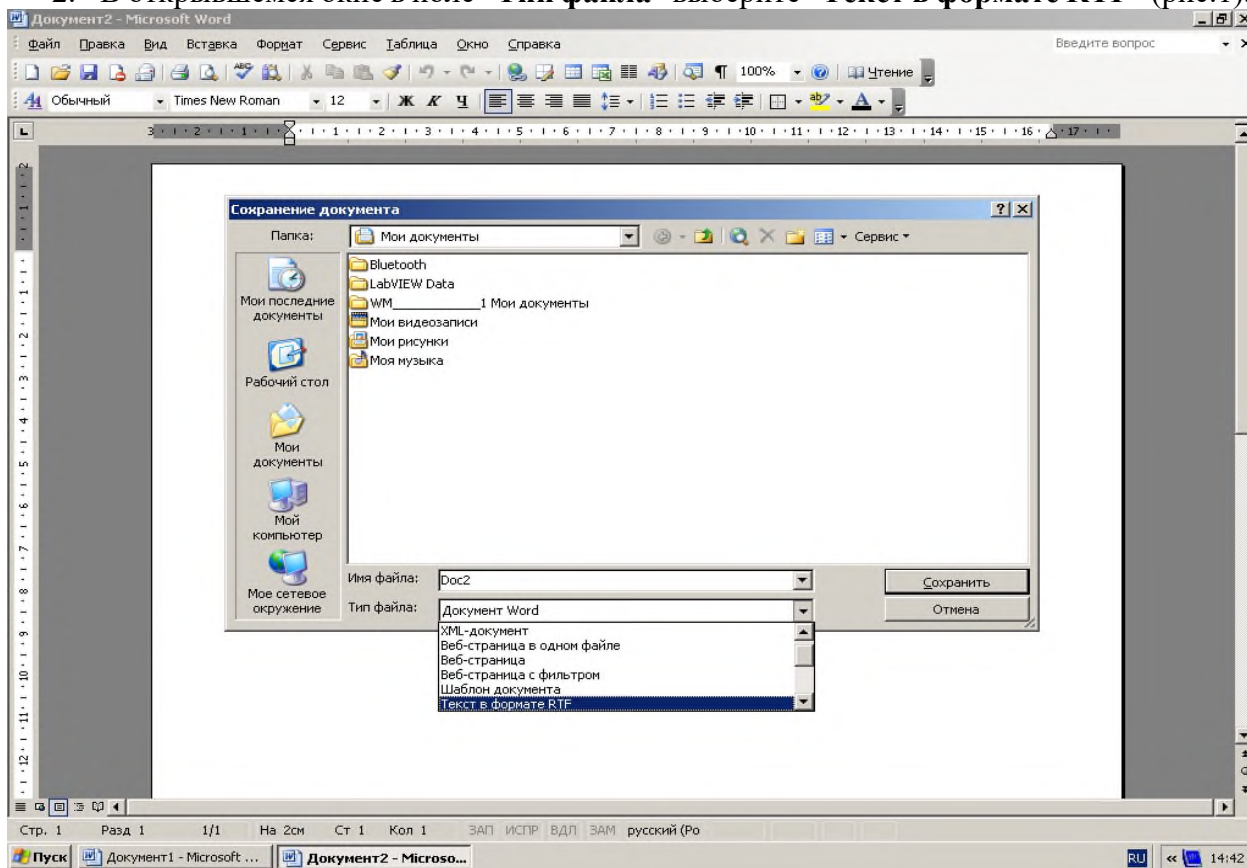


Рис. 1

3. Выберите команду “**Сохранить**”, при этом имя файла оставьте прежним.
4. В результате появится новый файл с именем существующего, но с другим расширением.
5. Далее закройте **WinWord** и удалите все зараженные Word-документы и обязательно удалите файл-шаблон **Normal.dot** в папке **WinWord**.
6. Запустите WinWord и восстановите документы из RTF-файлов в соответствующий формат файла (рис. 2) с расширением (.doc).
7. В результате этой процедуры макровирус будет удален из системы, а практически вся информация останется без изменений.

Примечание:

1. Этот метод рекомендуется использовать, если нет соответствующих антивирусных программ.
2. При конвертировании происходит потеря невирусных макросов, используемых в данном файле, поэтому перед запуском вышеописанной процедуры, рекомендуется сохранить их исходный текст, а после обезвреживания вируса – восстановить необходимые макросы в первоначальном виде.

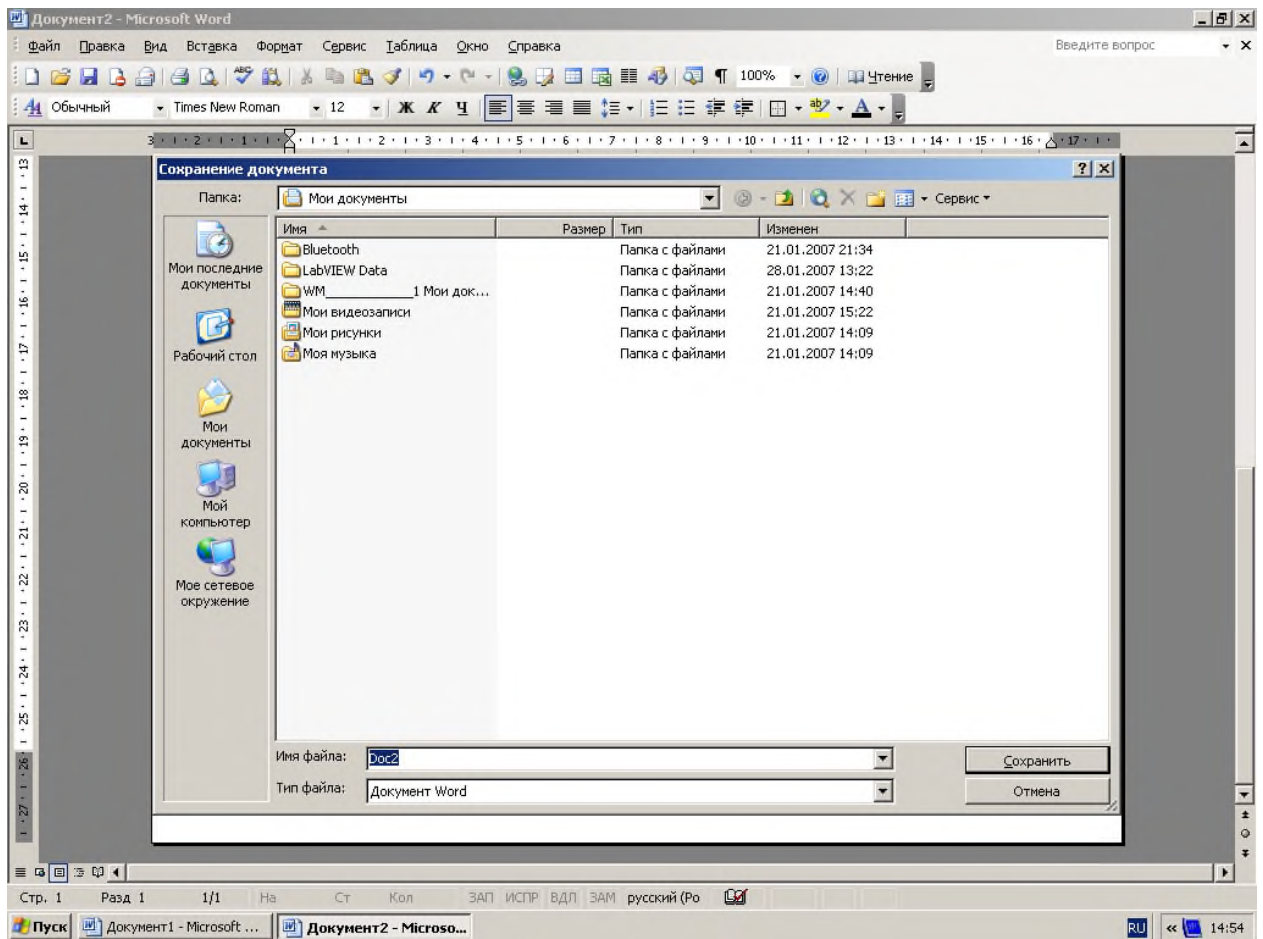


Рис. 2

8. Для последующей защиты файлов от макровирусов включите защиту от запуска макросов.
9. Для этого в **WinWord** выберите последовательно пункты меню: **Сервис-Макрос-Безопасность** (рис. 3).

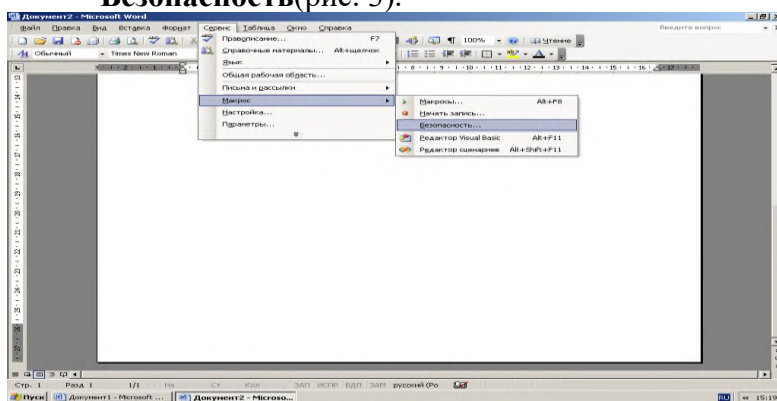


Рис. 3

10. В открывшемся окне на вкладке **Уровень безопасности** отметьте пункт **“Очень высокая”** (рис. 4).

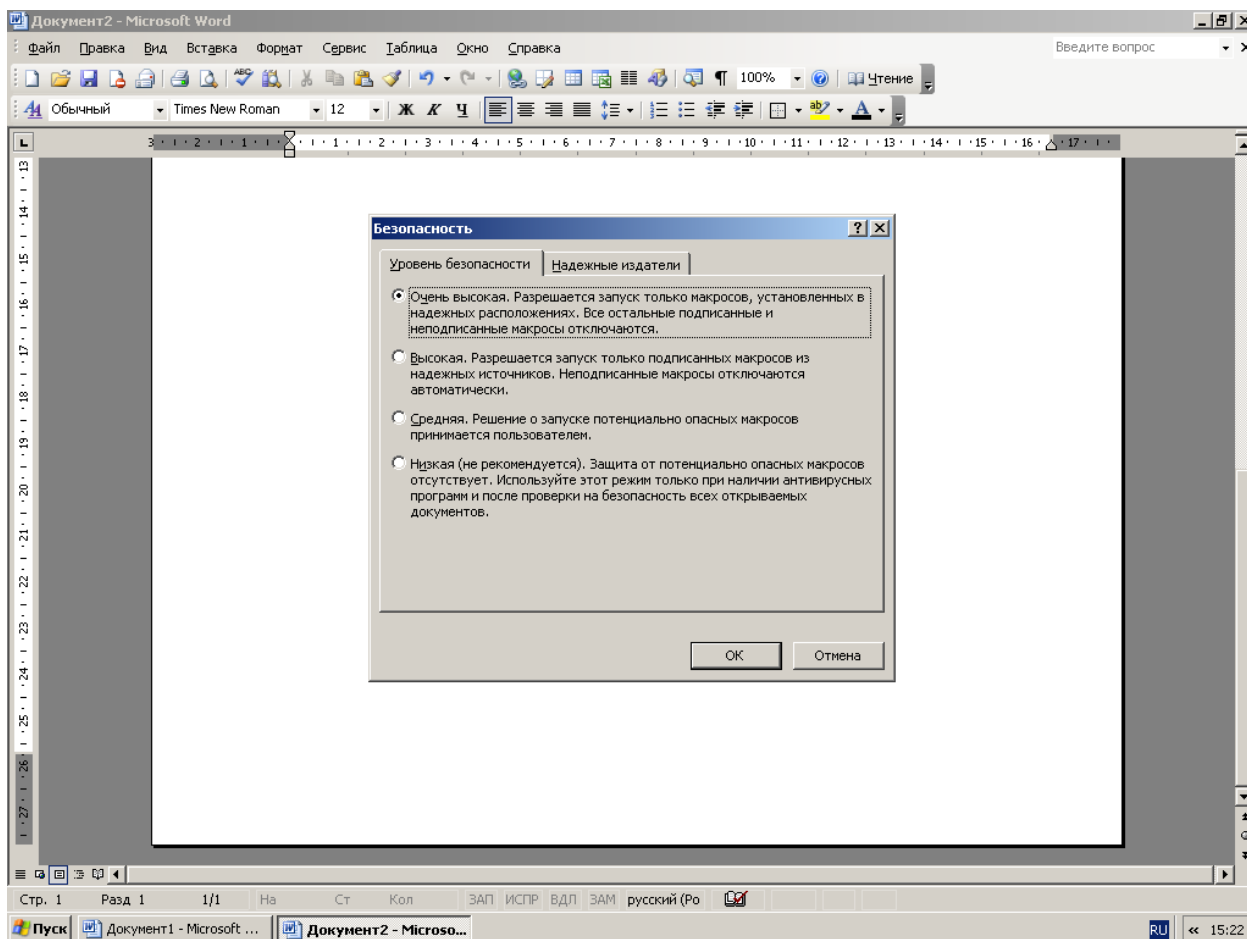


Рис. 4

Практическое занятие «Настройка Norton Commander»

Цель занятия: научиться настраивать Norton Commander.

Содержание работы:

1. Оболочка Norton Commander

Norton Commander (NC) обеспечивает выполнение следующих функций:

- редактирование файлов;
- упрощение выполнения конкретного набора команд MS DOS.

Достоинством Norton Commander является то, что в нем у вас остается возможность выполнения любых команд MS DOS (кроме PATH и SET).

ВЫЗОВ Norton Commander

[d:] [path] NC

либо [d:] [path] NCSMALL,

где d - диск, path - путь доступа.

На экране появляются панели Norton Commander, внизу экрана - меню функциональных клавиш NC (максимальное количество панелей - 2, одна - активная).

ИСПОЛЬЗОВАНИЕ ФУНКЦИОНАЛЬНЫХ КЛАВИШ

Клавиша	Назначение
F1	Помощь (Help)
F2	Вызов созданного пользователем меню
F3	Просмотр (View) указанного файла

F4	Редактирование (Edit) указанного файла
F5	Копирование (Copy) выбранной группы файлов
F6	Переименование или перемещение (Ren/Mov) выбранной группы файлов, переименование каталога
F7	Создание (Mkdir) каталога
F8	Удаление (Delete) выбранной группы файлов или каталога
F9	Удаление (Delete) выбранной группы файлов или каталога
F10	Выход (Quit) в режим команд MS-DOS
Shift-F3	Работает в любом режиме так, как F3 работает в режиме без панелей
Shift-F4	Работает в любом режиме так, как F4 работает в режиме без панелей
Shift-F5	Работает в любом режиме так, как F5 работает в режиме без панелей
Shift-F6	Работает в любом режиме так, как F6 работает в режиме без панелей
Shift-F7	Работает в любом режиме так, как F7 работает в режиме без панелей
Shift-F8	Работает в любом режиме так, как F8 работает в режиме без панелей
Shift-F9	Сохранить настройку в файле NC.INI

Инструкция по Norton Commander

Alt-F1	Выбрать диск (левая панель)
Alt-F2	Выбрать диск (правая панель)
Alt-F3	Просмотр (View) указанного файла, но для файлов типов dbf и wk? не вызываются программы dbview.exe и 123view.exe
Alt-F4	Редактирование (Edit) указанного файла альтернативным редактором
Alt-F5	Не задействована
Alt-F6	Не задействована
Alt-F7	Поиск файла(ов) по всем каталогам диска
Alt-F8	Выбор команды из стека сохраненных команд
Alt-F9	Переход в 43-строчный режим (для EGA,VGA)

Alt-F10	Создать окно с деревом каталогов активной панели (NCD-Tree)
Ctrl-F1	Зажечь/погасить левую панель
Ctrl-F2	Зажечь/погасить правую панель

Если вы выполняете команду из меню NC, то прервать работу можно с помощью клавиш F10 или ESC.

УКАЗАНИЕ И ИСПОЛНЕНИЕ ФАЙЛОВ

В данной конструкции встречаются термины - "указанный файл" и "выбранный файл или группа файлов". "Указанный файл" в конкретный момент времени может быть только один и от остальных файлов он визуально отличается (выделен фоновой полосой).

"Выбранный файл или группа файлов" визуально отличаются от остальных файлов (выделено цветом, яркостью или вертикальной яркой полоской справа). Исполняется всегда указанный файл. Для работы в NC используют клавиатуру, "мышь" либо и то и другое.

При использовании клавиатуры:

/УКАЗАТЬ ФАЙЛ/ - использовать клавиши направлений;

/ИСПОЛНИТЬ/ - нажать клавишу ENTER.

При использовании "мыши":

/УКАЗАТЬ ФАЙЛ/ - нажать левую кнопку "мыши";

/ИСПОЛНИТЬ/ - дважды быстро нажать левую кнопку.

ВЫБОР И ОТМЕНА ФАЙЛОВ

С использованием клавиатуры выбор и отмена группы файлов производится двумя способами:

1. Нажать серую клавишу (на цифровой клавиатуре) "+" (для выбора) или "-" (для отмены выбора), после чего в диалоговом "окне" указать конкретное имя файла (группы файлов), используя при необходимости символы * и ?;

2. Указать файл. Нажать клавишу INSERT (если файл уже выбран, то эти действия приведут к отмене выбора).

При использовании "мыши":

/ВЫБРАТЬ ФАЙЛ/ - нажать правую кнопку "мыши", повторное нажатие приведет к отмене выбора.

Над выбранными файлами могут производиться групповые операции копирования, удаления, пересылки и переименования.

ПАНЕЛИ Norton Commander

Активна только одна панель. Настройка панелей производится в меню NC (F9) по вашему желанию.

Комбинации клавиш	Назначение
ТАВ или Ctrl+I	Переключение активной панели
Ctrl+O	Удаление панелей с экрана
Ctrl+P	Удаление неактивной панели
Ctrl+U	Обмен панелей местами
Ctrl+L	Просмотр статуса активной панели число и суммарный объём файлов текущего каталога, ёмкость ОЗУ, текущего диска, свободное место в них

Повторное использование указанных комбинаций клавиш имеет обратное действие.

РАБОТА С КАТАЛОГАМИ В ПАНЕЛЯХ Norton Commander.

Войти в указанный каталог: указать его и нажать ENTER или исполнить, используя манипулятор.

Выход в родительский каталог: указать файл(две точ.) и нажать ENTER или исполнить, используя манипулятор, или нажать Ctrl+PgUp без указания каталога.

Повторное чтение каталога - Ctrl+R (выбранные группы файлов при этом не сохраняются), возможен переход на другой диск.

Комбинации клавиш	Назначение
----------------------	------------

Ctrl+\	Переход в корневой каталог
--------	----------------------------

ПРИМЕЧАНИЕ: используя NC легко удалять каталоги файлов.

Процедура удаления состоит в следующем:

- войти в каталог;
- произвести выбор всех файлов каталога (см.4)
- выполнить F8 (Delete);
- выйти в родительский каталог (при этом каталог, который вы хотите удалить указан полосой);
- выполнить F8 (Delete)

ОПИСАНИЕ КОМАНД Norton Commander.

Команды View (просмотреть) и Edit (редактировать) из меню NC выполняются с указанным файлом.

Команды COPY(копировать), REN/ MOV(заменить имя / переместить),DELETE (удалить)работают либо с указанным файлом, либо с выбранной группой файлов. При этом появляется соответствующее диалоговое "окно". Для команд COPY и REN/ MOV указывают путь и имя файла. Для команды DELETE, подтверждают удаление, выбрав "delete", и отменяют, выбрав "cancel". Возможно удаление файлов с атрибутом только чтение", при этом запрашивается дополнительное подтверждение.

Возможен ввод из командной строки любой команды MS-DOS, при этом введенные команды помещаются в стек и могут быть впоследствии извлечены из него в область ввода. При редактировании команды в области ввода используются приведенные в следующей таблице клавиши.

Клавиши редактирования командной строки и вспомогательные клавиши

Комбинации клавиш	Назначение
Ctrl+B	Удалить/зажечь меню клавиш
Ctrl+E	Показать в области ввода предыдущую команду из стека
Ctrl+X	Показать в области ввода следующую команду из стека
Ctrl+ J	Поместить в область ввода наименование указанного файла (курсор должен находиться в конце области ввода и предшествоваться пробелом)
Ctrl+ D	Перемещение курсора на символ вправо

Ctrl+ S	Перемещение курсора на символ влево
Ctrl+ → Ctrl+ F	Перемещение курсора на слово вправо
Ctrl+ ← Ctrl+ A	Перемещение курсора на слово влево
Ctrl-Home	Перемещение курсора в начало строки
Ctrl-End	Перемещение курсора в конец строки

ПРОСМОТР(View) УКАЗАННОГО файла.

Укажите файл.

Выполните View (F3) из NC меню.

Клавиши, используемые в режиме просмотра файла

Комбинации клавиш	Назначение
F7	Поиск(Search) цепочки, указанной в диалоговом "окне" (регистр не имеет значения).
Shift-F7	Повторить последний поиск
↑	Экран перемещается вверх по тексту
↓	Экран перемещается вниз по тексту
→	Экран перемещается вверх по тексту на страницу
←	Экран перемещается вниз по тексту на страницу
→	Текст на экране перемещается влево
←	Текст на экране перемещается вправо
Ctrl+	Текст на экране перемещается влево на 40 символов
Ctrl+ ←	Текст на экране перемещается вправо на 40 символов
Home	Просмотр начала файла
End	Просмотр конца файла
F10 или Esc	Выход (Quit) в NC

- Для просмотра файла используйте клавиши PgUp и PgDn;
- Для поиска указанной цепочки используйте функциональную клавишу F7. В диалоговом "окне" наберите требуемую цепочку (регистр не имеет значения). В результате поиска

первая найденная строка с указанной цепочкой перемещается в начало экрана.

Практическое занятие «Изучение структуры файла nc.ext и его редактирование»

Цель занятия: Изучение структуры файла nc.ext и его редактирование

Содержание работы:

На практике очень удобно, чтобы файл с соответствующим расширением был бы связан с соответствующей ему программой, например:

текстовый файл *.txt - с текстовым редактором, файл базы данных

*.dbf - с системой управления базами данных, программы *.bas ,

*.pas - со своей средой программирования, архивный файл - со своим архиватором.

Тогда при нажатии Enter Norton Commander будет автоматически загружать соответствующий выделенный файл в соответствующую ему программу. Это экономит время пользователя ПК и облегчает его работу. Для создания такого режима работы служит файл nc.ext. Он находится в каталоге NC со всеми остальными файлами Norton Commander и как любой другой текстовый файл может быть отредактирован в любом текстовом редакторе для DOS, в том числе и в редакторе, встроенном в Norton Commander. Для редактирования файла nc.ext проще всего найти его курсором и нажать F4.

Файл nc.ext может иметь, например, следующую структуру:

txt: lexicon !.!

pas: turbo !.!

bas: qbasic !.!

dbf: c:\clipper\dbu\dbu !.!

rar: rar en !.!

ain: ain l !.!

В файле nc.ext сперва указывают расширение файла, например txt, а затем командный файл той программы, с которой он связывается, например, lexicon. В общем случае надо указывать полный путь этого файла, например, c:\clipper\dbu\dbu. В конце каждой команды ставится !.!. В этом случае Norton Commander при нажатии Enter будет загружать соответствующий файл в соответствующую ему программу, согласно записи в файле nc.ext. Длина файла nc.ext в NC4 не должна превышать величины около 700 байт. Иначе последние команды файла nc.ext не будут выполняться. Последние две команды позволяют удобно просматривать RAR и AIN архивы при помощи NC.

Практическое занятие «Изучение меню пользователя и редактирование файла nc.mnu»

Цель занятия: Изучение меню пользователя и редактирование файла nc.mnu

Содержание работы:

Меню пользователя Norton Commander значительно облегчает и ускоряет запуск прикладных программ и поэтому широко применяется на практике. Меню пользователя содержится в файле nc.mnu. Если файл nc.mnu находится в каталоге Norton Commander (NC), то такое меню пользователя является Главным (Main). Длина файла nc.mnu в NC4 ограничена величиной около 1.6 Кб. Иначе последние пункты меню не будут выполняться. Главное меню вызывается при нажатии F2 из любого каталога, если в нем нет своего файла nc.mnu. В последнем случае будет вызываться Локальное меню (когда файл nc.mnu находится в любом каталоге, кроме каталога NC). Файл nc.mnu является текстовым и редактируется в любом текстовом редакторе, но чаще всего для этого применяется встроенный в Norton Commander редактор. Структура файла nc.mnu может иметь в простейшем случае следующий вид:

P: Парковка машины

park
L: Лексикон
lexicon
W: Windows-95
win
E: Фотон
c:\ft\ft
N: Norton Utilities
norton

Сначала в файле ps.mni указывают ту клавишу (латинскую букву или цифру), которую надо будет нажимать для запуска программы. Затем на этой же строчке после двоеточия дают описание запускаемой программы на русском или английском языке. На второй строчке после двух пробелов записывается команда DOS, в общем случае с указанием полного пути командного файла. Иногда продолжение этой команды записывается на третьей, четвертой и т.д. строчке, если используется команда DOS cd. В нашем примере при нажатии клавиши W будет запущен Windows. Расширения файлов exe, com, bat можно в командах не указывать.

Практическое занятие «Администрирование операционной системы»

Цель занятия: Администрирование операционной системы

Содержание работы:

Перед установкой Intranet сервера необходимо хотя бы приблизительно оценить требования, накладываемые им на оборудование. И хотя известно, что учебные ПК заранее соответствуют данным требованиям, в реальности данный шаг очень важен при создании сервера Intranet. Один из существенных аспектов технического оформления узла Intranet - потенциальные затраты. Перед выбором оборудования необходимо правильно определить потребности, исходя из ожидаемых характеристик создаваемого узла. Если существующее оборудование по каким-либо причинам не подходит для выполнения планируемых сервисов, то имеет смысл подумать о его модернизации либо о создании менее сложного узла Intranet.

Планирование потребностей

Правильно определить свои потребности в аппаратуре для создания узла Интернета очень важно, но и очень трудно. Их невозможно рассчитать по строгим формулам, кроме того, они изменяются во времени. Специфика работы любого сервера Intranet не накладывает жестких ограничений на оборудование. Исключение составляет только размер дискового пространства. Здесь проявляются основные отличия узла Intranet от узла Internet. Intranet, как правило, строится на основе быстрых локальных сетей. При этом ширина полосы пропускания не является критичным фактором, чего не скажешь об Internet. Планируя потребности необходимо ответить на следующие вопросы:

- Какие сервисы Intranet будут использоваться?
- Каковы виды материалов? Много ли графики? Каковы требования к хранилищу файлов и объему жесткого диска?
- Прогнозируется ли рост узла в ближайшем будущем? Какая аппаратура при этом необходима?
- Нужен ли дополнительный компьютер?
- Кто будущие пользователи? Новички ли они или имеют опыт?
- Немаловажным также является вопрос обеспечения надежности и отказоустойчивости работы сервера.

Только ответив на все данные вопросы можно переходить к выбору аппаратуры.

Выбор аппаратуры

В идеальном случае имеет смысл воспользоваться только той аппаратурой, которая у Вас уже есть. В общем случае следует ставить IIS на уже существующие серверы локальной

сети, работающие под управлением WindowsNTServer/Workstation 4.0 либо Windows 95. В таблице 2.1 представлены минимальные требования к оборудованию, составленные на основе некоторых эмпирических данных.

Таблица 2.1. Минимальные требования к аппаратуре

	<i>Трафик</i>		
Сервисы (ниже)	низкий (менее 3 пользователей одновременно)	средний (от 3 до 15 пользователей одновременно)	высокий (более 15 пользователей одновременно)
WWW	486/66 32 Мб ОЗУ 1 Гб диск	486/100 32 Мб ОЗУ 1 Гб диск	586/90 48 Мб ОЗУ 2 Гб диск
FTP	486/66 32 Мб ОЗУ 1 Гб диск	486/66 32 Мб ОЗУ 1 Гб диск	486/66 32 Мб ОЗУ 1 Гб диск
Электронная почта	586/75 48 Мб ОЗУ 1 Гб диск	586/75 48 Мб ОЗУ 1 Гб диск	586/90 64 Мб ОЗУ 2 Гб диск
Все сервисы	586/75 48 Мб ОЗУ 2 Гб диск	586/90 64 Мб ОЗУ 1,5 Гб диск	586/90 64+ Мб ОЗУ 2 Гб диск

Вышеперечисленные требования являются минимальными, но не критическими, и узел будет работать удовлетворительно при указанных нагрузках.

Если создаваемый сервер Intranet критичен для бизнеса, то необходимо продумать обеспечение его отказоустойчивости. Для этого можно в состав оборудования включить источники бесперебойного питания. Также возможно применение аппаратных дисковых массивов RAID.

После выбора требуемой аппаратуры можно переходить к установке и настройке IIS.

Установка и настройка сервера Intranet

Перед установкой сервера IIS необходимо решить ряд вопросов, связанных с установкой и настройкой WindowsNT 4.0/95. В частности, необходимо:

- установить WindowsNTServer/Workstation 4.0 либо Windows 95;
- настроить сетевой протокол TCP/IP для ЛВС, в которой создается Intranet.

Также, наряду с этим, на WindowsNT 4.0 для возможности установления более надежной защиты информации IIS желательно использование файловой системы NTFS, а не FAT.

После установки в IIS все параметры принимают значения по умолчанию. При необходимости можно изменить текущие установки IIS.

Отдельно также следует рассмотреть вопросы защиты информации на IIS, выходящие за рамки данного практикума. При этом пользователи, работающие с WindowsNT могут пользоваться мощными системными средствами такими, как пользовательские бюджеты, права доступа к каталогам и т.д.

Установка IIS

IIS может поставляться как в составе WindowsNT 4.0, так и отдельно. Для установки IIS необходимо запустить программу установки INETSRV\INETSTP.EXE. Процесс установки начинается с диалога, в котором необходимо выбрать устанавливаемые компоненты (рисунок 2.1).

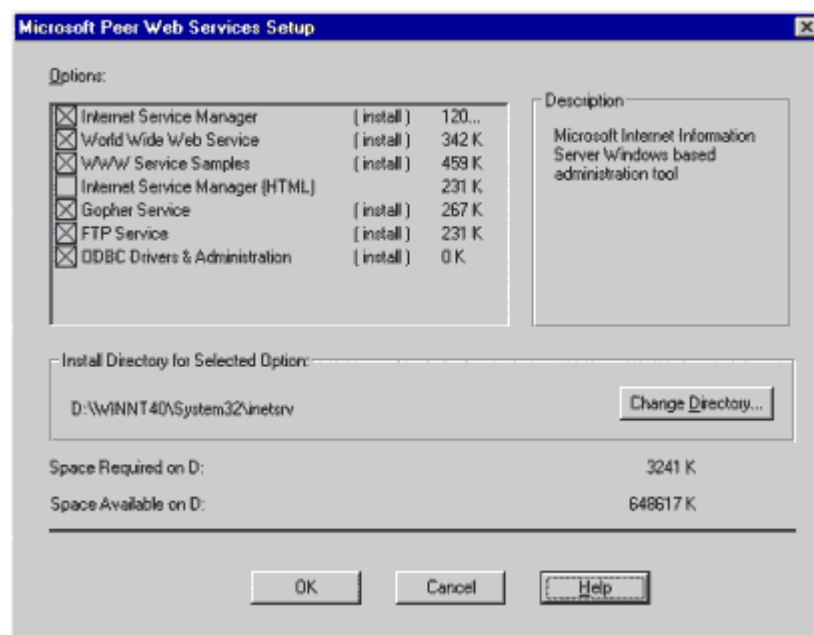


Рис. 2.1. Выбор устанавливаемых компонент

Для каждого компонента в диалоге приведены его символическое название, планируемая над ним операция (install - установка и remove - удаление) и его размер в килобайтах. В нижней части диалога показано требуемое и доступное дисковое пространство. Для установки компонента необходимо его отметить. Для уже установленных компонент можно их удалить просто убрав отметку рядом с ними. Возможна работа со следующими компонентами:

- *InternetServiceManager* - административные программы для управления сервисами Intranet;
- *WorldWideWeb* - WWW сервис для публикации гипертекстовых документов;
- *WWWServiceSamples* - различные примеры гипертекстовых документов;
- *InternetServiceManager (HTML)* - специальная версия административных программ, работающих в формате гипертекстовых документов через произвольный Web-браузер;
- *GopherService* - Gopher сервис;
- *FTPService* - FileTransferProtocol (FTP) сервис (В составе системы Windows также поставляется стандартный FTP сервис. Но лучше использовать FTP сервис, поставляемый в составе IIS, так как он удобнее и лучше в обращении.);
- *ODBCDriversandAdministration* - ODBC драйверы, необходимые для обеспечения доступа к БД посредством ODBC из WWW сервиса (Не устанавливайте их в случае, если уже установлена поддержка ODBC в системе.);

При необходимости можно сменить каталог, в который будет установлен IIS (кнопка *ChangeDirectory:*). По окончании нажмите кнопку *OK*.

Далее нужно определить корневые каталоги для сервисов WWW, FTP и Gopher (рисунок 2.2).

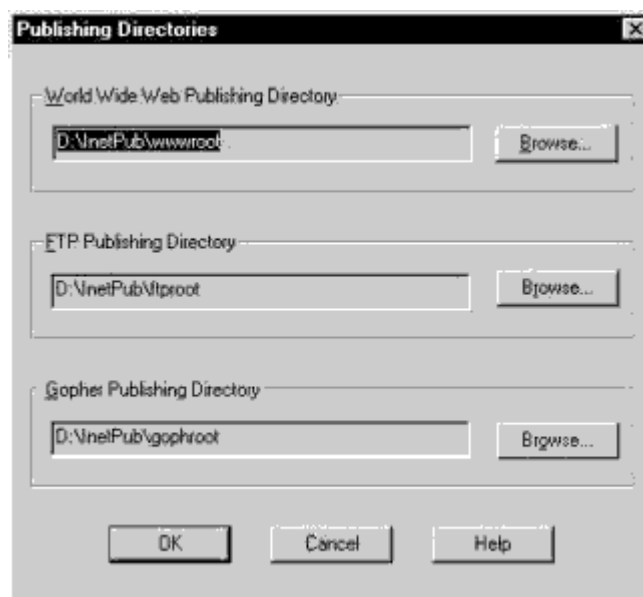


Рис. 2.2. Определение корневых каталогов для сервисов

Заметьте, что на этапе установки IIS Вы не можете в качестве диска корневого каталога устанавливать сетевой диск. По окончании нажмите кнопку *OK*. Дальнейшая установка IIS выполняется автоматически.

В дополнение к сервисам IIS отдельно можно установить пакет электронной почты. Для WindowsNTServer существует несколько систем электронной почты, рассмотрение которых выходит за рамки данного практикума.

Настройка IIS

После установки всех сервисов IIS, а именно WWW, FTP и Gopher, необходимо настроить каждый из них. Ниже полностью рассмотрен процесс конфигурирования сервиса WWW. Для FTP и Gopher описываются только те опции, которые отличают эти сервисы; информация о других их опциях, аналогичных опциям WWW, не приводится. Настройку можно производить как из стандартного приложения Internet Service Manager, так и из приложения Internet Service Manager (HTML), осуществляющего общение с пользователями через интерфейс гипертекстовых документов. Вариант работы с гипертекстом практически ничем не отличается от стандартного варианта. Далее приводятся примеры экранных форм только для стандартного варианта программ.

Настройка WWW

Для отображения параметров установленного WWW сервиса выполните двойной щелчок мышью на строке сервиса WWW в Internet Service Manager (рисунок 2.3).

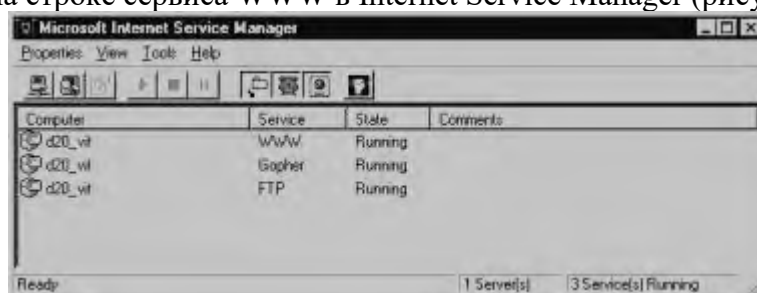


Рис. 2.3. Панель управления сервиса
Либо для варианта Internet Service Manager (HTML) (рисунок 2.4).



Рис. 2.4. Управление сервисами через браузер
В диалоге настройки три вкладки: *Service*, *Directories*, *Logging*.

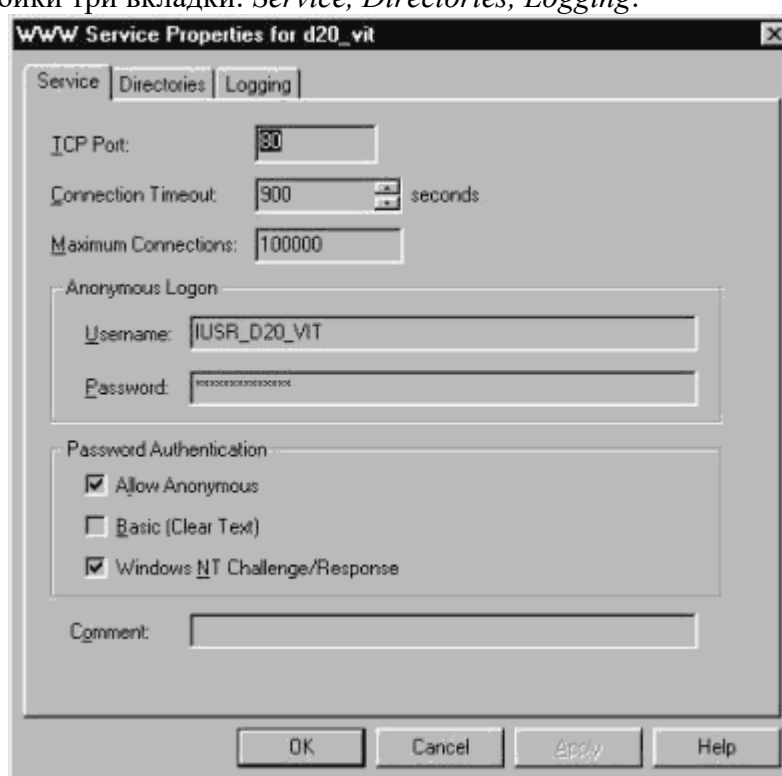


Рис. 2.5. Настройка свойств сервиса

На вкладке *Service* можно изменить перечисленные далее опции (рисунок 2.5):

- Опция *ConnectionTimeout* - время простоя сессии, после которой сервер прервет ее работу. В большинстве случаев подходит значение по умолчанию - 900 секунд. Его можно уменьшить при интенсивном трафике.
- Опция *MaximumConnections* - количество сессий, обрабатываемых сервером одновременно. Оптимальное значение данного параметра в значительной степени зависит от скорости работы оборудования и размера оперативной памяти. Данное значение необходимо уменьшить, если пользователи жалуются на медленную работу.
- Два поля группы *Anonymous Logon* - *Username* и *Password* - идентификатор и пароль пользовательского бюджета, созданного и используемого IIS для обращения к файлам узла Web от имени пользователей Web. По умолчанию *Username* имеет форму *IUSR_имямашины*.
- Группа *Password Authentication* включает три допускаемых варианта подключения пользователей IIS-серверу: *Allow anonymous*, *Basic (ClearText)*,

WindowsNTChallenge/Response. Опция *Allow anonymous* означает, что пользователям не обязательно идентифицировать себя при подключении к серверу, опция *Basic (ClearText)* - у пользователя будут запрошены идентификатор и пароль для передачи на IIS. Опция позволяет клиентам WindowsNT/95, использующим Microsoft Internet Explorer версии 2.0 или более поздней, идентифицировать себя для сервера автоматически.

- Поле *Comment* используется для ввода комментария, отображаемого в Internet Information Manager. Можно поместить сюда произвольную информацию.

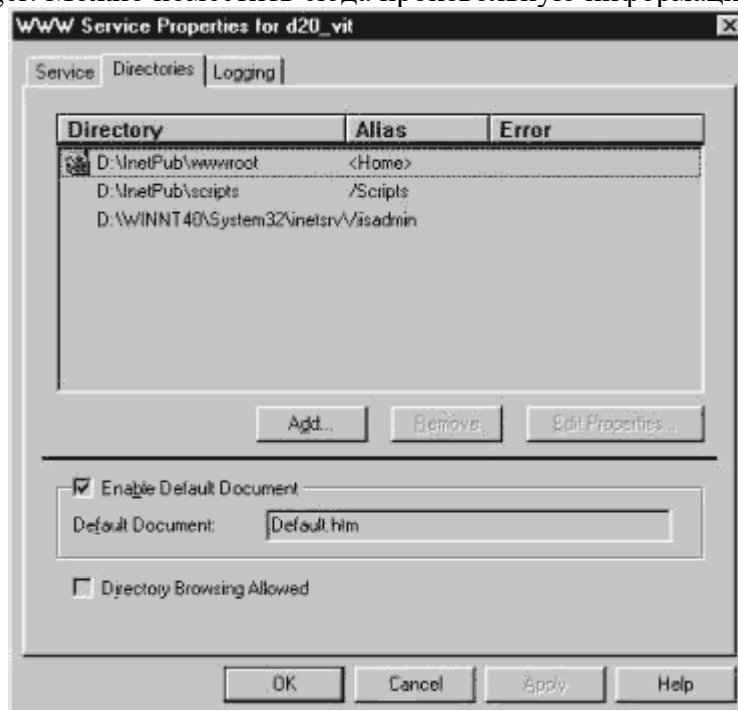


Рис. 2.6. Каталоги Web-сервера

По кнопкам *Add:* и *EditProperties:* для каждого каталога выводится диалог его настройки:

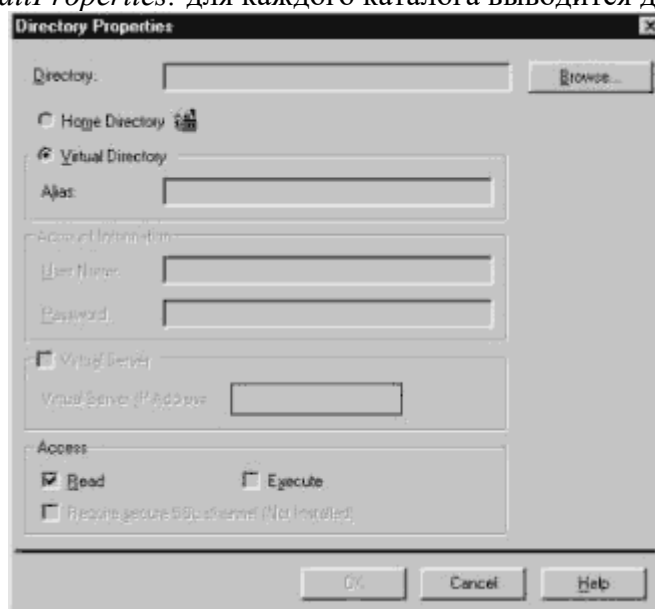


Рис. 2.7. Свойства каталога

Здесь для каждого каталога можно установить его виртуальный образ (*Alias*). Можно ограничить права доступа к нему (*Read, Execute*). Если каталог берется из сетевого ресурса, то можно задать параметры доступа к данному ресурсу (*Account Information: User Name, Password*).

Для сервиса WWW в системе WindowsNTServer 4.0 также имеется очень важная возможность организации виртуального Web-сервера. Для этого системе назначаются

дополнительные IP-адреса и домены (Технология их назначения достаточно сложна и выходит за рамки данного практического курса.). Далее каталог просто назначается корневым каталогом для виртуального Web-сервера (*Virtual Server*, *Virtual Server IP Address*). Это позволяет физически на одной машине вести сразу несколько Web-серверов с разными IP-адресами.

На вкладке *Directories* также можно изменить перечисленные далее опции:

- Список содержит имена реальных и виртуальных каталогов, доступных IIS. Виртуальные каталоги *www root* и *scripts* заданы по умолчанию.
- Опция *Enable Default Document* означает, что пользователю, подключившемуся к узлу Web без указания конкретного файла WWW, будет предоставлен указанный в этом поле документ. Установлена по умолчанию в *Default.htm*.
- При установленном флажке *Directory Browsing Allowed* пользователь сможет просматривать список файлов на сервере Web и выбирать из него. По умолчанию отключен. (Рекомендуется не включать его во избежание непредусмотренного доступа к файлам.)

Виртуальный каталог *www root* используется для хранения гипертекстовых файлов .HTM, шаблонов гипертекста .HTX, графических изображений и др. файлов гипермедиа. Для него обязателен доступ Read (чтение). Виртуальный каталог *Scripts* служит для размещения серверных программ (CGI-скриптов), основной доступ: *Execute* (выполнение); если там будут располагаться гипертекстовые документы, то дополнительный надо установить доступ по *Read*. При смене режимов доступа в процессе работы следует перезапустить сервис *www*.

Использование интерфейса CGI может потребовать дополнительных установок в системном реестре (программа *Regedt32.exe*):

```
HKEY_LOCAL_MACHINE
SYSTEM
CurrentControlSet
W3SVC
Parameters
ScriptMap
```

Так для выполнения командных процедур нужно добавить значение (*Add Value*) соответствия типа файла .CMD и команды вызова *C:\WINNT\SYSTEM32\CMD.EXE /C %s %s*. Следующей идет вкладка *Logging* (рисунок 2.8).

На ней доступны следующие опции:

- Первая опция – *Enable Logging* - по умолчанию включает мониторинг. Если нет необходимости отслеживать использование сервера или выяснять потенциальные проблемы защиты, то его можно отключить, сохранив дополнительное дисковое пространство.
- При включенном мониторинге можно выбирать, куда помещается журнал: в файл или базу данных. (Мы рекомендуем использовать файл, так как большинство аналитических средств IIS работает только с файлами.)
- Опция *Automatically Open New Log* разрешает IIS создавать новый файл журнала так часто, как это необходимо; иначе создается один файл.
- *Log File Directory* задает каталог для файлов журнала.

Настройка FTP

В диалоге для сервиса FTP представлены следующие вкладки: *Service*, *Messages*, *Directories*, *Logging*. Некоторые из них содержат параметры, аналогичные параметрам WWW. Ниже будут рассмотрены только опции, отличающие FTP.

Рассмотрим закладку *Service* (рисунок 2.9):

- Опции *Allow anonymous connections*, *Allow only anonymous connections* разрешают только анонимное подключение. Если же эти опции отключены, то некоторым пользователям системой Windows могут быть предоставлены особые права доступа.

- Кнопка *Current Sessions* выводит список текущих пользователей FTP сервера IIS.

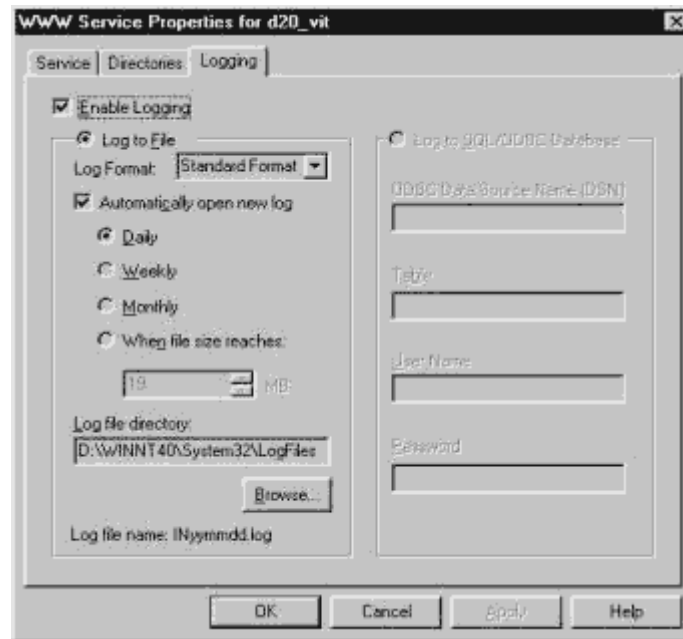


Рис. 2.8. Настройка протоколирования

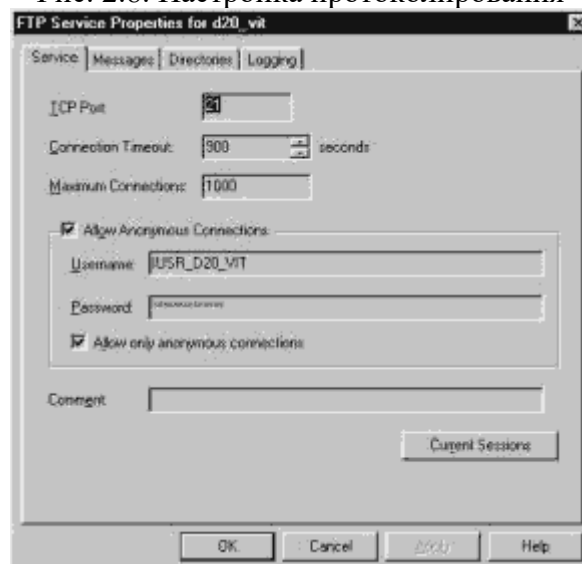


Рис. 2.9. Настройка подключения к FTP

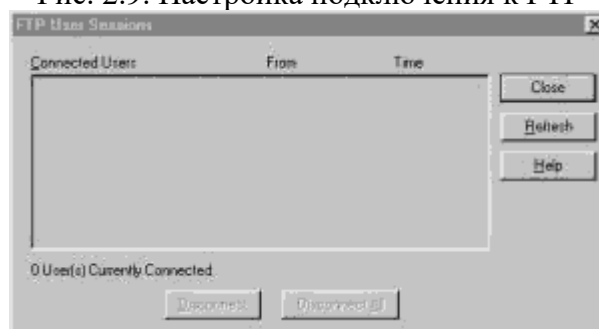


Рис. 2.10. Сеансы пользователей FTP

Закладка *Message* характерна только для сервиса FTP. Она позволяет задавать сообщения, получаемые пользователем, при его подключении, отключении или если сервер занят.

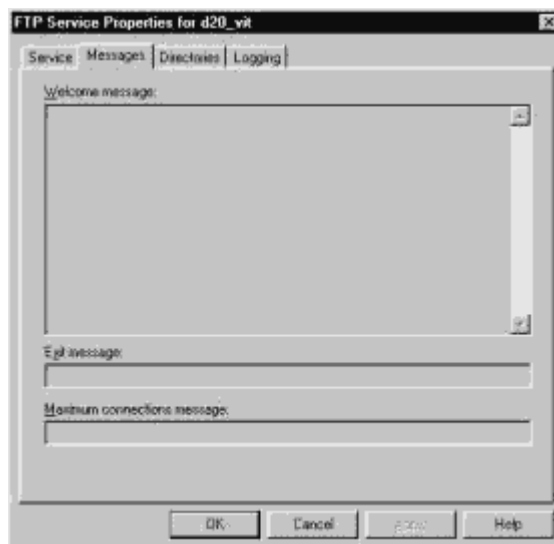


Рис. 2.11. Настройка сообщений

В закладке *Directories* добавлен раздел , позволяющий представлять виртуальные каталоги в стиле *Unix* либо *MS-DOS*. Закладка *Logging* аналогична закладке из WWW.

2.2.Перечень вопросов и заданий для промежуточной аттестации

Форма: дифференцированный зачет

Список вопросов к дифференцированному зачету

ОП.02 Операционные системы и среды

1. «Операционное окружение»
2. «Состав операционного окружения»
3. «Назначение операционного окружения»
4. «Программы»
5. «Виды программ»
6. «Сервисные программы»
7. «Сервисные программы поддержки операционного окружения»
8. «Режим пользователя»
9. «Базовая машина»
10. «Расширенная машина»
11. «Принцип работы ядра операционной системы»
12. «Сравнительный анализ операционных систем»
13. «Операционные системы семейства Windows»
14. «Операционные системы семейства Unix»
15. «Функции ОС»
16. «Состав ОС»
17. «Свойства ОС»
18. «Настройка операционного окружения»
19. «Утилиты операционного окружения»
20. «Параметры загрузки операционной системы»
21. «Эволюция ОС Windows»,
22. «Эволюция ОС Unix »
23. «Способы защиты памяти»
24. «Сегментная организация памяти»
25. «Установка ОС Windows»

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п.п.	Содержание изменения	Дата, номер протокола заседания кафедры, подпись зав.кафедрой
1	2	3
1		
2		
3		
4		