ОПЕРАЦИОННЫЕ СИСТЕМЫ

Учебное пособие

Аннотация

Функции и архитектуры операционных систем, основные инструменты управления ОС Windows

Телепова Т. П.

TelepovaTP@e1.ru

Российский государственный профессионально-педагогический университет

Учебное пособие составлено для студентов направления подготовки 09.03.02 Информационные системы и технологии бакалаврской программы «Информационные технологии в медиаиндустрии»

Автор: Т.П. Телепова

Содержание

1 Введение в операционные системы

1.1 Понятие и классификация и операционной системы

По определению ГОСТ под операционной системой (ОС) понимают систему программ, предназначенную для обеспечения определенного уровня эффективности вычислительной системы за счет автоматизированного управления ее работой и предоставляемых пользователям определенного рода услуг.

В логической структуре типичной вычислительной системы операционная система занимает положение между устройствами с их микроархитектурой, машинным языком и, возможно, собственными (встроенными) микропрограммами – с одной стороны – и прикладными программами с другой.

С 1990-х годов наиболее распространёнными операционными системами являются системы семейства Microsoft Windows и системы класса UNIX (особенно Linux и MacOS).

Классификация операционных систем

Различные операционные системы могут различаться алгоритмами управления основными устройствами компьютера (процессорами, памятью, устройствами), особенностями использованных методов проектирования, типами аппаратных платформ, областями использования и многими другими свойствами.

Рассмотрим развитие ОС с точки зрения исторического развития вычислительных ресурсов (опуская историю механических и электромеханических устройств).

Первый период (1945-1955 гг., ламповые машины) характеризовался:

1. Операционных систем нет.
2. Программирование осуществлялось исключительно на машинном языке.
3. Все задачи организации вычислительного процесса решались вручную каждым программистом с пульта управления.
4. Вычислительная система выполняла одновременно только одну операцию (ввод-вывод или собственно вычисления).
5. Программы выполнялись строго последовательно.

Такой режим работы вычислительной системы называется последовательной обработкой данных. В целом первый период характеризуется крайне высокой стоимостью вычислительных систем, их малым количеством и низкой эффективностью использования.

Второй период (1955 г. – начало 1960-х гг., компьютеры на основе транзисторов). Этот период с точки зрения развития технологии вычислительного процесса характеризуется:

1. Развитие алгоритмических языков (LISP, COBOL, ALGOL-60, PL-1 и т.д.).

2. Появляются первые компиляторы, редакторы связей, библиотеки математических и служебных подпрограмм.

3. Именно в этот период происходит разделение персонала на программистов и операторов, разработчиков вычислительных машин и специалистов по эксплуатации.

4. Изменяется сам процесс прогона программ. Текст программы оформляется в виде колоды перфокарт – *задание,* с указанием необходимых ресурсов.

Основной недостаток такого процесса обработки информации: смена запрошенных ресурсов вызывает приостановку выполнения программ, в результате процессор часто простаивает. Для повышения эффективности использования ВМ задания с похожими ресурсами начинают собирать вместе, создаётся пакет заданий.

5. Появляются первые системы пакетной обработки (СПО). Программа, постоянно находящаяся в памяти компьютера, автоматизирует запуск одной программы за другой из пакета и тем самым увеличивает коэффициент загрузки процессора. Программу можно назвать простейшей операционной системой, обеспечивающей обработку программ в однопрограммном пакетном режиме.

6. Появление магнитного диска (для которого не важен порядок чтения информации, в отличие от магнитных лент, которые были устройствами последовательного доступа) привело к возможности выбора определённого задания на исполнение.

7. СПО начинают решать задачи планирования заданий в зависимости от наличия запрошенных ресурсов, приоритетов, срочности вычислений и т. д.

Третий период (начало 1960-х – 1980 г.), компьютеры на основе интегральных микросхем) и многозадачных ОС.

Многозадачные ОС позволяют параллельно выполнять несколько задач, распределяя между ними вычислительные ресурсы. Такой процесс обработки данных достигается за счет реализации принципа мультипрограммирования.

Многозадачность называется условной если между задачами делится только оперативная память, но не процессорное время, так как реально работает только одна активная задача, остальные ждут или ее завершения, или внешней команды на переключение (активизацию) другой задачи. Типичный представитель такой ОС – Windows 3.x.

Вытесняющая многозадачность – решение о переключении на другую задачу принимает сама ОС (например, на основе квантования процессорного времени между выполняемыми процессами). В качестве таких ОС можно назвать Windows NT, UNIX.

Роль операционной системы в поддержке многозадачности:

1. Организация интерфейса между прикладной программой и ОС при помощи системных вызовов.

2. Планирование использования процессора, т.е. организация очереди из заданий.

3. Создание и ведение контекста задания при переключении с одного задания на другое для обеспечения правильного продолжения вычислений;

4. Стратегии управления памятью, как ограниченным ресурсом ВС, т.е. реализация процессов размещения, замещения, выборки информации из памяти, реализация виртуализации памяти.

5. Организация хранения информации на внешних носителях и обеспечение доступа к конкретному файлу. Поскольку программам может потребоваться произвести санкционированный обмен данными, необходимо их обеспечить средствами коммуникации.

6. Для корректного обмена данными необходимо разрешать конфликтные ситуации, возникающие при работе с различными ресурсами, и предусмотреть координацию программами своих действий, т. е. снабдить систему средствами синхронизации.

Особенности аппаратной поддержки многозадачности:

1. Реализация защитных механизмов. Появление привилегированных для ОС (например, команды ввода-вывода) и непривилегированных команд для ограничения доступа программ пользователя к распределению ресурсов.

2. *Наличие прерываний.* Внешние прерывания оповещают ОС о том, что произошло асинхронное событие (т.е. событие происходящее независимо от других условий), например, завершилась операция ввода-вывода. Внутренние прерывания (сейчас их принято называть исключительными ситуациями,) возникают, когда выполнение программы привело к ситуации, требующей вмешательства ОС, например, деление на ноль или попытка нарушения защиты.

3. *Развитие параллелизма* в архитектуре ВМ за счёт прямого доступа к памяти и организации каналов ввода-вывода. КВВ – самостоятельные в логическом отношении устройства, которые работают под управлением собственных программ, находящихся в памяти. В современных машинах КВВ называют периферийными процессорами или процессорами ввода-вывода.

КВВ и интерфейсы выполняют следующие функции:

1. Выбор и подготовка к обмену того или иного ВУ, управление обменом.

2. Осуществляют определенную обработку данных, подлежащих обмену: изменение форматов, перемещаемых данных, формирование адресов, контроль количества передаваемых байтов и т. д.

КВВ позволяют иметь машины с переменным составом периферийных устройств, обеспечивают параллельную работу периферийных устройств как между собой, так и по отношению к процессору, обеспечивают автоматическое распознавание и реакцию процессора на различные ситуации, возникающие в периферийных устройствах.

Итак, реализация многозадачности привела к включению в состав ВМ отдельной системы управления, которая обеспечивает одновременную автоматическую работу процессора, оперативной памяти, каналов, внешних устройств. Эта система управления не исключает, а дополняет систему управления процессора (УУ), которая обеспечивает автоматическое выполнение команд программы в процессоре.

В настоящее время структура ВМ состоит из двух частей. Первая часть – аппаратная (hardware - процессор, оперативная память, каналы, внешние устройства). Вторая часть – операционная система (software, основная часть системного программного обеспечения ВМ, куда кроме ОС входят и другие системные программы), обеспечивающая управление многозадачности.

В зависимости от областей использования многозадачные ОС подразделяются на три типа:

1. Системы пакетной обработки, СПО (ОС ЕС).
2. Системы с разделением времени (Unix, Linux, Windows).
3. Системы реального времени.

В *системах пакетной обработки* идея многозадачности заключается в следующем: пока одна программа выполняет операцию ввода-вывода, процессор не простаивает, как это происходило при однопрограммном режиме, а выполняет другую программу. Когда операция ввода-вывода заканчивается, процессор возвращается к выполнению первой программы.

Недостаток*:* эти системы обеспечивают высокую производительность при обработке больших объемов информации, но снижают эффективность работы пользователя в интерактивном режиме.

Системы с разделением времени

В системах с разделением времени специальный компонент ОС, называемый планировщиком заданий (процессов), делит процессорное время на короткие отрезки и предоставляет их поочерёдно различным исполняющимся программам (процессам). При этом создается видимость одновременного выполнения нескольких задач.

Развитием идеи системы с разделением времени является:

1. Возможность одновременной работы нескольких пользователей на одной компьютерной системе. Отсюда ОС могут рассматриваться как с т. з. числа одновременно работающих пользователей как однопользовательские (MS DOS) и многопользовательские (Unix, Linux, Windows).

Эти системы обладают меньшей пропускной способностью, но обеспечивают высокую эффективность работы пользователя в интерактивном режиме.

2. Механизм виртуальной памяти, который позволяет уменьшить ограничения на количество работающих пользователей за счёт гибкого распределения всей памяти ВМ (как основной, так и внешней) для исполняемой программы. Основная часть программы находится на диске и фрагмент, который необходимо в данный момент выполнять, может быть загружен в оперативную память, а ненужный — выкачан обратно на диск.

3. В системах разделения времени пользователь получил возможность эффективно производить отладку программы в интерактивном режиме и записывать информацию на диск, не используя перфокарты, а непосредственно с клавиатуры.

Системы реального времени

ВМ под управлением соответствующей ОС контролирует и управляет внешними по отношению к ВМ событиями, происходящими в некоторых внешних объектах. Такой способ организации работы специфичен тем, что контроль и управление происходят в темпе, согласованном со скоростью поступления данных от объекта управления. Применяются для управления технологическим процессом или техническим объектом, например, летательным объектом, станком и т.д.

По *количеству поддерживаемых процессоров* операционные системы делятся на многопроцессорные и однопроцессорные.

Четвертый период (с 1980 г.), появление персональных компьютеров, как результат развития микропроцессорных технологий). Одним из важнейших признаков является разделение ОС на локальные и сетевые. Компьютеры стали использоваться не только специалистами, что потребовало разработки «дружественного» программного обеспечения, функционирующих в *сетевых* или *распределенных* операционных систем.

Каждая машина в сети работает под управлением своей локальной операционной системы, отличающейся от операционной системы автономного компьютера наличием дополнительных средств (программной поддержкой для сетевых интерфейсных устройств и Доступа к удаленным ресурсам).

Распределенная система, напротив, внешне выглядит как обычная автономная

система. Пользователь не знает и не должен знать, где хранятся его файлы — на локальной или удаленной машине – и где его программы выполняются.

1.2 Функции операционных систем

Просмотрев этапы развития вычислительных систем, мы можем выделить шесть основных функций, которые выполняли классические операционные системы в процессе эволюции:

1. Планирование заданий и использования процессора.
2. Обеспечение программ средствами коммуникации и синхронизации.
3. Управление памятью.
4. Управление файловой системой.
5. Управление вводом-выводом.
6. Обеспечение безопасности.

Каждая из приведенных функций обычно реализована в виде подсистемы, являющейся структурным компонентом ОС. В каждой операционной системе эти функции, конечно, реализовывались по-своему, в различном объеме. Они не были изначально придуманы как составные части операционных систем, а появились в процессе развития, по мере того как вычислительные системы становились все более удобными, эффективными и безопасными.

Это дает основание предложить следующую трактовку определения ОС.

Операционная система – это упорядоченная последовательность системных управляющих программ совместно с необходимыми информационными массивами, предназначенная для планирования исполнения пользовательских программ и управления всеми ресурсами вычислительной машины (программами, данными, аппаратурой, оператором и другими распределяемыми и управляемыми объектами) с целью предоставления возможности пользователям эффективно (в некотором смысле) решать задачи, сформулированные в терминах вычислительной системы

В процессе эволюции ОС возникло несколько важных концепций, которые стали неотъемлемой частью теории и практики ОС. Рассмотрим одну из них, а именно системные вызовы.

Системные вызовы как функция ОС

В любой операционной системе поддерживается механизм, который позволяет пользовательским программам обращаться к услугам ядра ОС. В операционных системах IBM они назывались системными макрокомандами, в ОС Unix такие средства называют системными вызовами.

Системные вызовы (system calls) иногда еще называют программными преры-

ваниями, в отличие от аппаратных прерываний, которые чаще называют просто прерываниями. Системные вызовы операционных систем (Unix, Windows, MacOS OS, и т. д.) реализуются программами пользователей с помощью интерфейса прикладного программирования API (Application Programming Interface).

Это набор готовых классов, процедур, функций, структур и констант, предоставляемых приложением (библиотекой, сервисом) ОС для использования во внешних программных продуктах (программах пользователя).

Например, если пользовательскому процессу необходимо считать данные из файла, он должен выполнить команду системного вызова, т.е. выполнить прерывание с переключением в режим ядра и активизировать функцию операционной системы для считывания данных из файла.

У большинства современных ОС, концепция, лежащая в основе интерфейса прикладного программирования практически одинакова, хотя детали могут быть различны. Примеры API операционных систем – POSIX, Windows API, Linux Kernel API.

Наиболее часто применяемые процедуры API: *fork* – создание нового процесса, *exit* – завершение процесса, *open* – открывает файл, close - закрывает файл, *read* – читает данные из файла в буфер, *write* – пишет данные из буфера в файл, *stat* – получает информацию о состоянии файла, *mkdir* – создает новый каталог, *rmdir* – удаляет каталог и т.д.

В ОС Win32 API существует более 1000 системных вызовов, это связано с тем, что графический интерфейс пользователя в Windows встроен в ядро. Поэтому Win32 API имеет много вызовов для управления окнами, текстом, шрифтами т.д. Кроме того, в Win32 API отделен от процедур, непосредственно обрабатывающие системные вызовов. Это позволяет в разных версиях ОС W менять системные вызовы, не переписывая программы.

В общем виде процесс системного вызова можно описать следующим образом:

- пользовательская программа запрашивает сервис у операционной системы, осуществляя системный вызов;
- библиотеки процедур ОС, загружают машинные регистры определенными параметрами и осуществляют прерывание процессора, после чего управление передается обработчику данного вызова, входящему в ядро операционной системы. Цель таких библиотек – сделать системный вызов похожим на обычный вызов подпрограммы.
- при системном вызове задача переходит в привилегированный режим или режим ядра (kernel mode). В этом режиме работает код ядра операционной системы. Ядро операционной системы имеет полный доступ к памяти пользовательской программы и к его контексту.

В большинстве операционных систем системный вызов осуществляется командой программного прерывания (INT).

1.3 Архитектуры операционных систем

1.3.1 Монолитные операционные системы

Система, которая характеризуется отсутствием структуры как таковой (рис. 1).

В общем виде операционная система монолитного типа представляет собой набор процедур, каждая из которых может вызывать другие. Процедуры ОС компилируются, а затем компонуются в единый объектный файл. Однако, предполагает всё-таки следующую структуру:

1 Главная программа, которая осуществляет обработку системных прерываний.

2 Набор служебных процедур, реализующие системные вызовы.

3 Набор утилит, обслуживающие служебные процедуры.

В ОС программа, которая инициирует и прекращает процессы обработки системных прерываний, называется супервизором (supervisor). Это управляющая резидентная программа в составе операционной системы, координирующая распределение и использование ресурсов вычислительной системы.

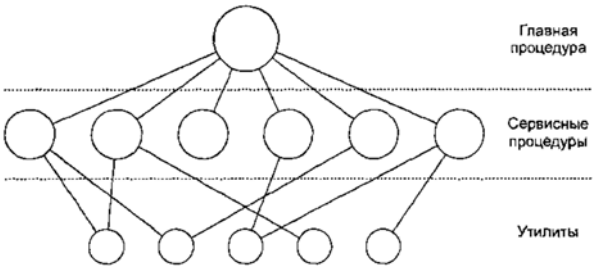

Рисунок 1 – Монолитная операционная система

В операционной системе может быть несколько супервизоров. Например, супервизор ввода-вывода инициирует и прекращает процессы ввода-вывода, супервизор основной памяти осуществляет учет и динамическое распределение области оперативной памяти и т.д. В качестве примера монолитной ОС можно привести ОС MS-DOS, ОС Windows 95/98. В современных ВС в основном речь идёт о модульных ОС с монолитным ядром и в качестве примеров приводятся большинство UNIX-систем Linux.

Монолитные ядра имеют долгую историю развития и, на данный момент, являются наиболее архитектурно зрелыми. Вместе с тем имеют следующие недостатки:

1. Монолитные ядра требуют перекомпиляции при любом изменении состава оборудования.

2. «Разбухание» кода монолитных ядер делает их малопригодными для систем, ограниченных по объёму ОЗУ, например, встраиваемых системах, микроконтроллерах и т. д.

1.3.2 Модульные операционные системы

В отличие от архитектуры ОС рассмотренной выше, модульные ОС структурно состоят из модулей (рис. 2), каждый из которых реализует определённый набор функций. Наиболее общим подходом к структуризации операционной системы является разделение всех ее модулей на две группы:

1. Ядро – модули, выполняющие основные функции ОС.
2. Модули, выполняющие вспомогательные функции ОС.

Модули ядра выполняют такие базовые функции как управление процессами, памятью, устройствами ввода вывода, обработка прерываний.

Вспомогательные модули как правило подразделяются на следующие:

✓ утилиты – программы, решающие задачи сопровождения ВС (сжатие дисков, архивация);

✓ системные обрабатывающие программы (редакторы, отладчики, компиляторы и пр.)

✓ программные модули, обеспечивающие графический пользовательский интерфейс.

✓ библиотеки процедур различного назначения, упрощающие разработку приложений (библиотека математических функций, функций ввода-вывода).

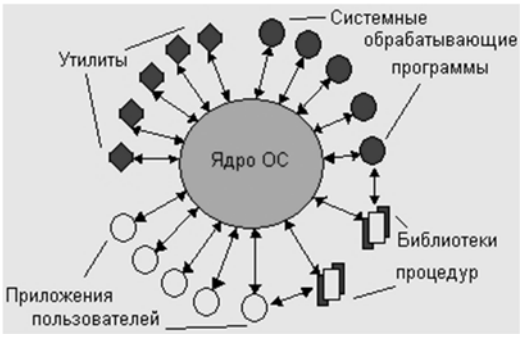

Рисунок 2 – Модульные операционные системы

Как и обычные пользовательские приложения, для выполнения своих задач вспомогательные модули, обращаются к функциям ядра посредством системных вызовов. Вспомогательные модули являются транзитными загружаются в оперативную память только на время выполнения своих функций. Такая организация ОС экономит оперативную память компьютера.

1.3.3 Модульное ядро

Современная, усовершенствованная модификация архитектуры монолитных ядер операционных систем компьютеров. Модульность реализуется за счёт механизма подгрузки модулей поддерживающих то или иное аппаратное обеспечение

(например, драйверов). При этом не требуются полной перекомпиляции ядра при изменении состава аппаратного обеспечения компьютера.

Подгрузка модулей может быть, как динамической (без перезагрузки ОС,), так и статической (выполняемой при перезагрузке ОС). Подгружаемые модули загружаются в адресное пространство ядра и в дальнейшем работают как интегральная (т.е. единая) часть ядра. Динамическая подгрузка модулей помогает сократить размер кода, работающего в пространстве ядра до минимума, что актуально для встраиваемых устройств с ограниченными аппаратными ресурсами.

Важным свойством ядерной архитектуры, является возможность защиты кодов и данных операционной системы. Защита реализуется за счет выполнения функций ядра в привилегированном режиме.

Привилегированный режим ядра

Реализуются средствами аппаратной поддержки. Аппаратура компьютера поддерживает как правило 2 режима привилегий – пользовательский режим (user mode) и привилегированный режим, или режим ядра (kernel mode), или режим супервизора (supervisor mode) (рис. 3). При этом приложениям запрещается выполнение в пользовательском режиме команд, связанных с использованием системных ресурсов. Возможно, кроме ситуации, когда инструкция обращается к области памяти, отведенной данному приложению.

Рисунок 3 – Операционная система с ядром в привилегированном режиме

ОС может на этой основе создать программным способом развитую систему защиты, образующую иерархию уровней привилегий. В этом случае режим супервизора реализуется на нулевом уровне с максимальным доступом к ресурсам, режим пользователя, – с ограниченным доступом.

1.3.4 Многослойная структура ОС

Функциональность операционной системы разделяется на уровни. Например, уровень управления аппаратурой (это уровень ядра ОС), уровень управления памятью, уровень управления процессами, уровень файловой системы.

Каждый уровень может обращаться за услугами только к соседнему нижележащему уровню через его интерфейс. Внутренние структуры данных и процедуры уровня не доступны другим уровням и не зависят от реализации их процедур.

Такая организация системы имеет много достоинств:

- упрощается разработка системы. Сначала определяются функции слоев и их интерфейсы, а затем наращивается мощность слоев по направлению «снизу-вверх»;
- при модернизации ОС изменение модулей внутри слоя не влияет на какие-либо изменения в остальных слоях.

По мере дальнейшего усложнения операционных систем обнаружились недостатки многоуровневой архитектуры:

- интерфейс уровня со временем становится громоздким, увеличивается сложность реализации отдельных уровней
- деление функциональности ОС на уровни является сдерживающим фактором в развитии операционной системы, так как изначально созданную структуру уровней не поменять из соображений совместимости.

Многослойный подход обычно распространяется и на структуру ядра ОС. Ядро может состоять из следующих слоев (рис. 4):

1 Средства аппаратной поддержки ОС. К ним относятся средства поддержки привилегированного режима, система прерываний, те, которые участвуют в организации вычислительного процесса.

2. Машинно-зависимые компоненты ОС. Этот слой экранирует вышележащие слои ядра от особенностей аппаратуры, что позволяет разрабатывать вышележащие слои на основе машинно-независимых модулей. Например, слой HAL (Hardware Abstraction Layer, HAL) операционной системы Windows NT позволяет обеспечить переносимость 99% кода на системы с различным оборудованием.

3. Базовые механизмы ядра. Этот слой выполняет операции, связанные с управлением процессами, обработкой прерываний, управлением памятью.

4. Менеджеры или диспетчера ресурсов. Этот слой состоит из функциональных модулей, управляющие файловой системой, системой ввода-вывода, виртуальной памятью и процессами.

5. Интерфейс системных вызовов. Это самый верхний слой, взаимодействует непосредственно с приложениями и системными утилитами, образуя прикладной программный интерфейс операционной системы API.

Классическая многоуровневая архитектура в наиболее современных операционных системах в настоящее время вытесняется клиент-серверной архитектурой, реализованной на принципе микроядерности.

Рисунок 4 – Многослойная структура ядра ОС

1.3.5 Концепция микроядерной архитектуры ОС

Суть микроядерной архитектуры ОС состоит в следующем:

- в привилегированном режиме остается работать небольшая часть ОС, называемая микроядром (рис. 5).

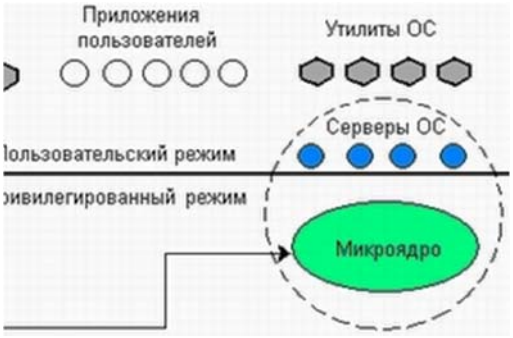

Рисунок 5 – Микроядерная архитектура ОС

- в состав микроядра входят машинно-зависимые модули, а также модули, выполняющие базовые функции ядра, микроядро защищено от остальных частей ОС и приложений;
- остальные более высокоуровневые функции ядра оформляются в виде приложений, работающие в пользовательском режиме. Например, менеджеры ресурсов (файловая система, система ввода-вывода, виртуальная память и процессы).
- модули ОС, представленные в пользовательском режиме в виде приложений, вызываются пользовательскими приложениями для выполнения определённых функций с помощью специально разработанного механизма вызова приложений.
- менеджеры ресурсов, вынесенные в пользовательский режим, называются серверами ОС, то есть модулями, основным назначением которых является обслуживание запросов локальных приложений и других модулей ОС.

Механизм обращения к функциям ОС, оформленным в виде серверов, выглядит следующим образом (рис. 6):

- клиент (прикладная программа, либо другой компонент ОС), запрашивает выполнение функции у сервера, посылая ему сообщение;

- так как прямая передача сообщений между приложениями невозможна из-за изолированности их адресных пространств в качестве посредника выступает микроядро, которое в привилегированном режиме имеет доступ к адресным пространствам приложений;
- микроядро передает сообщение, содержащее имя и параметры вызываемой процедуры нужному серверу;
- сервер выполняет запрошенную операцию, ядро возвращает результаты клиенту с помощью другого сообщения.

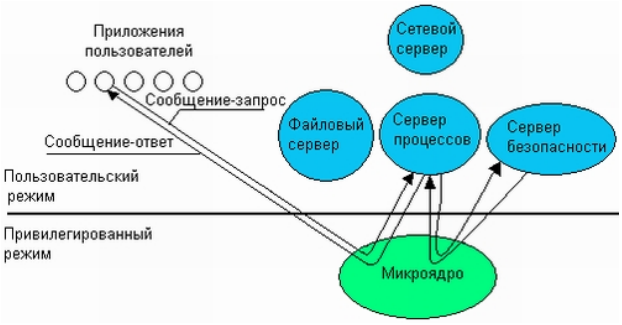

Рисунок 6 – Механизм обращения к функциям ОС

Таким образом, микроядерная операционной система соответствует известной модели клиент-сервер, в которой роль транспортных средств выполняет микроядро.

1.3.6 Архитектура Windows

Windows представляет собой операционную систему с *гибридным ядром* (рис. 7). В состав микроядра входят машинно-зависимые модули. Основные системные функции реализованы в режиме ядра (управление процессами, памятью, устройствами, файловой системой и безопасностью). Однако ряд важных системных компонентов реализованы в пользовательском режиме. Например, процессы входа в систему, аутентификации, диспетчера сеансов, а также подсистемы окружения.

Компоненты режима ядра

Диспетчер системных сервисов работает в режиме ядра, перехватывает вызовы функций от Ntdll.dll, проверяет их параметры и вызывает соответствующие функции из Ntoskrnl.exe.

Исполнительная система и ядро содержатся в Ntoskrnl.exe (NT Operating System Kernel – ядро операционной системы NT).

Исполнительная система представляет собой совокупность компонентов (называемых диспетчерами – manager), которые реализуют основные задачи операционной системы:

- диспетчер процессов (process manager) – управление процессами и потоками;
- диспетчер памяти (memory manager) – управление виртуальной памятью и

отображение её на физическую;

- монитор контроля безопасности – управление безопасностью;
- диспетчер ввода вывода (I/O manager), диспетчер кэша (cache Manager), диспетчер Plug and Play (PnP Manager) – управление внешними устройствами и файловыми системами;
- диспетчер электропитания (power manager) – управление электропитанием и энергопотреблением;
- диспетчер объектов (object manager) – управление служебными процедурами и структурами данных, которые необходимы остальным компонентам.

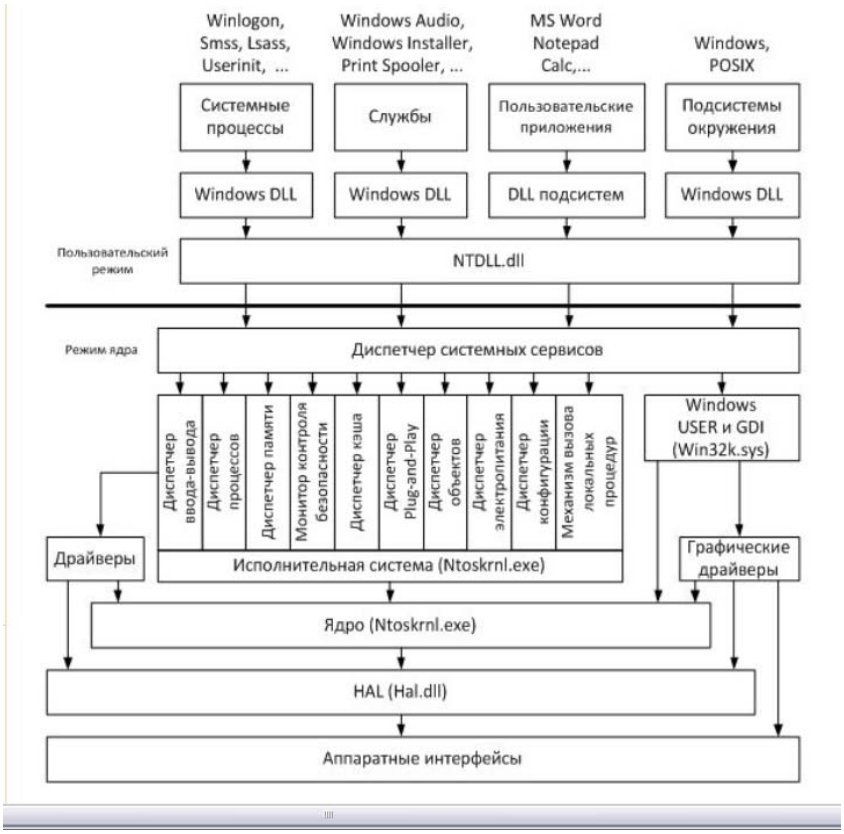

Рисунок 7 – Архитектура Windows

Ядро (Kernel) содержит функции, обеспечивающие поддержку компонентам исполнительной системы и осуществляющие планирование потоков, механизмы синхронизации, обработку прерываний. Компонент Windows USER и GDI отвечает за пользовательский графический интерфейс (окна, элементы управления в окнах – меню, кнопки и т. п., рисование), является частью подсистемы Windows и реализован в драйвере Win32k.sys.

Взаимодействие диспетчера ввода вывода с устройствами обеспечивают драйверы (drivers) – программные модули, работающие в режиме ядра, обладающие максимально полной информацией о конкретном устройстве. Однако, и драйверы, и ядро не взаимодействуют с физическими устройствами напрямую – посредником между программными компонентами режима ядра и аппаратурой является HAL. Слой HAL

позволяет скрыть от всех программных компонентов особенности аппаратной платформы (например, различия между материнскими платами), на которой установлена операционная система.

Компоненты пользовательского режима

В пользовательском режиме работают следующие виды процессов:

- системные процессы (system processes) – компоненты Windows, отвечающие за решение критически важных системных задач (т. е. аварийное завершение одного из этих процессов вызывает крах или нестабильную работу всей системы), но выполняемые в пользовательском режиме. Некоторые системные процессы:
  - Winlogon.exe – процесс входа в систему и выхода из неё;
  - Smss.exe (Session Manager – диспетчер сеансов) – процесс выполняет важные операции при инициализации системы (загрузка необходимых DLL, запуск процессов Winlogon и Csrss и др.);
  - Lsass.exe – процесс проверяет правильность введенных имени пользователя и пароля;
- службы (services) – приложения, работающие в фоновом режиме и не требующие взаимодействия с пользователем. Службы могут быть как частью операционной системы (Windows Audio – служба для работы со звуком), так и частью пользовательского приложения (служба СУБД SQL Server);
- пользовательские приложения (user applications) ¬– прикладные программы, запускаемые пользователем;
- подсистемы окружения (environment subsystems) – компоненты, предоставляющие доступ приложениям к некоторому подмножеству системных функций. Windows поддерживает две подсистемы окружения:

Все перечисленные процессы пользовательского режима (кроме подсистемы POSIX) для взаимодействия с модулями режима ядра используют библиотеки Windows DLL (Dynamic Link Library – динамически подключаемая библиотека).

Основные Windows DLL следующие:

Kernel32.dll – базовые функции, в том числе работа с процессами и потоками, управление памятью и вводом выводом; Advapi32.dll – функции, в основном связанные с управлением безопасностью и доступом к реестру; User32.dll – функции, отвечающие за управление окнами и их элементами в GUI приложениях (Graphical User Interface – графический интерфейс пользователя); Gdi32.dll – функции графического пользовательского интерфейса (Graphics Device Interface, GDI), обеспечивающие рисование на дисплее и принтере графических примитивов и вывод текста.

2 Инструменты управления в операционной системе Windows

2.1 Системный реестр

Реестр операционной системы Windows представляет собой централизованную базу данных параметров настройки системы и работающих в ней приложений, хранящихся в папках (%SystemRoot% – переменная среды Windows, задающая путь к реестру, для просмотра наберите в командной строке команду Set) и папке пользовательских профилей (Ntuser.dat). На рис. 8 представлено содержимое папки %SystemRoot%\System32\Config. В этом смысле реестр аналогичен разнообразным INI-файлам, а также файлам Autoexec.bat и Config.sys, которые использовались в MS-DOS. Реестр содержит информацию обо всех аппаратных средствах, программном обеспечении, операционной системе и сетевых параметрах компьютера.

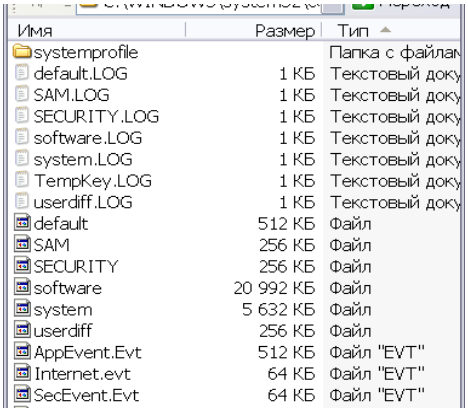

Рисунок 8 – Содержание папки %SystemRoot%\System32\Config

При работе в командной строке необходимо перейти в каталог C:\Windows\system32\config (рис. 9).

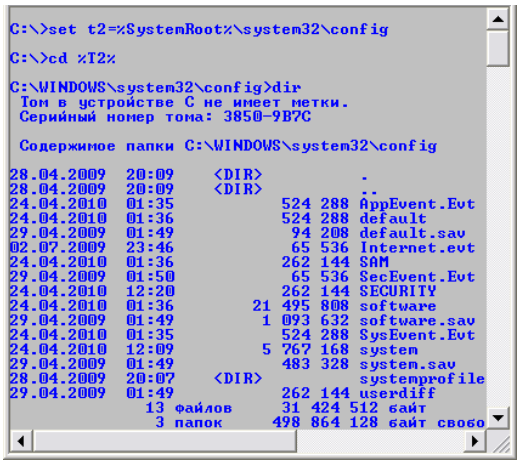

Рисунок 9 – Содержимое каталога C:\Windows\system32\config в командной строке

Структура реестра

Для работы с реестром используется простая и понятная утилита Regedit – редактор реестра (C:\WINDOWS\system32\regedt32.exe). Для ее запуска нажмите:

Пуск → Выполнить → введите команду "regedit"

Или по адресу C:\WINDOWS\system32\regedt32.exe откроется окно программы (рис. 10), в которой слева отображается дерево реестра, похожее по виду на отображение структуры диска в Проводнике, а справа выводятся ключи, содержащиеся в выбранном (активном) разделе.

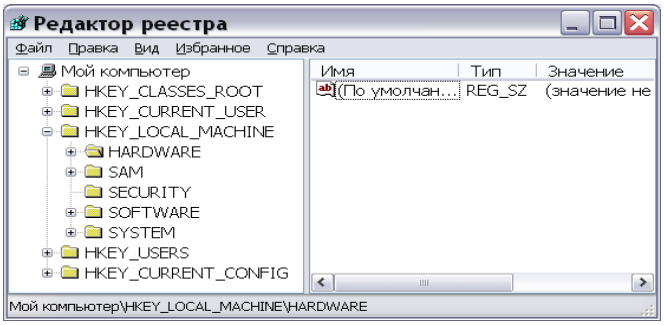

Рисунок 10 – Редактор реестра

С помощью редактора вы можете редактировать значения, импортировать или экспортировать реестр, осуществлять поиск. Реестр Windows имеет иерархическую древовидную структуру, состоящую из корневых разделов, вложенных подразделов и ключей.

Разделы и подразделы – это папки в левом окне regedit. *Ключ реестра или параметр* – это переменная, которой присвоено определённое значение, это то, что мы видим в правом окне regedit.

Все имена корневых разделов начинаются со строки HKEY_, что указывает разработчикам программного обеспечения на то, что это – дескриптор, который может использоваться программой. Описания корневых разделов реестра Windows приведены в таблице 1.

Таблица 1 – Корневые разделы реестра

Имя корневого раздела	Описание
HKEY_LOCAL_ MACHINE	Здесь сосредоточены основные параметры системы, оборудования, программного обеспечения
HKEY_CLASSES_ROOT	Содержит сведения о зарегистрированных расширениях, технологии OLE, механизме drag-and-drop и других функциях интерфейса.
HKEY_CURRENT_CONFIG	Содержит данные для текущего аппаратного профиля. Представляют собой наборы изменений, внесенных в стандартную конфигурацию устройств.
HKEY_CURRENT_USER	Содержит настройки для текущего пользователя (рабочий стол, настройки сети, приложения)
HKEY_USERS	индивидуальные настройки среды для каждого пользователя системы и профиль по умолчанию для вновь создаваемых пользователей

Реестр подразделяется на составные части, которые разработчики системы назвали кустами, или ульями по аналогии с ячеистой структурой пчелиного улья.

Куст – это раздел реестра, отображаемый как файл на жестком диске, представляет собой дискретную совокупность разделов, вложенных разделов и параметров, берущую начало в вершине иерархии реестра.

Отличие кустов от других групп разделов состоит в том, что они являются постоянными компонентами реестра. Кусты не создаются динамически при загрузке операционной системы и не удаляются при ее остановке. Каждый куст реестра Windows NT ассоциирован с набором стандартных файлов.

По умолчанию большинство файлов кустов (Default, SAM, Security и System) сохраняются в папке %SystemRoot%\System32\Config.

Поскольку куст представляет собой файл, его можно перемещать из одной системы в другую. Некоторые ульи, такие, как HKLM\HARDWARE, не сохраняются в файлах, а создаются при каждой загрузке, то есть являются изменяемыми. При запуске системы реестр собирается из ульев в единую древовидную структуру с корневыми разделами.

Кроме этих файлов, есть ряд вспомогательных, со следующими расширениями: LOG — журнал транзакций, в котором регистрируются все изменения реестра. Благодаря этому реализуется атомарность реестра — то есть в данный момент времени в реестре могут быть либо старые значения, либо новые, даже после сбоя. SAV – копии ульев в том виде, в котором они были после завершения текстовой фазы установки (первоначальная установка).

С кустами ассоциируются файлы четырех типов:

alt – содержит резервную копию жизненно важного куста HKEY_LOCAL_MACHINE/System;

log – содержит журнал транзакций, в котором регистрируются все изменения, внесенные в разделы и значимые элементы куста;

sav – содержит копии файлов куста в том виде, который они имели на момент завершения текстовой фазы процесса установки;

без расширения имени файла – содержит копию куста.

Данные реестра хранятся в виде параметров, расположенных в разделах реестра. Каждый параметр характеризуется именем, типом данных и собственно значением. Три части параметра реестра всегда располагаются в следующем порядке:

Имя	Тип данных	Значение
RegistrySizeLimit	REG_DWORD	0x8000000

В таблице 2 перечислены, типы данных, определенные и используемые в реестре.

Таблица 2 – Типы данных для параметров реестра

Тип данных	Описание	Пример
REG_BINARY	Представляет двоичный параметр. Двоичные параметры хранятся как двоичные (только комбинация нулей и единиц), но отображаются и вводятся в шестнадцатеричном формате.	01 00 14 80 90 00 00 9c 00
REG_DWORD	Представляет параметр тина DWORD, который состоит из шестнадцатеричных данных с максимальной длиной в четыре байта.	0x00000002
REG_EXPAND_SZ	Представляет раскрываемый строковый параметр, который обычно используется в путях к каталогам.	%SystemRoot%\dns.exe
REG_MULTI_SZ	Представляет параметр из нескольких строк.	Tcpip Afd RpcSc
REG_SZ	Представляет строковый параметр, содержащий последовательность символов.	DNS Server

2.2 Консоль управления Windows

Microsoft Management Console (консоль управления) – компонент операционной системы Windows. Позволяет системным администраторам и пользователям с помощью гибкого интерфейса конфигурировать и отслеживать работу системы.

Основной принцип действия MMC заключается в оснастках – небольшие подпрограммы, позволяющие настроить разные аспекты операционной системы. Наиболее примечательной является оснастка «Групповые политики», по своему строению похожая на Active Directory. Интерфейсы программирования MMC позволяют интегрировать оснастки с консолью (рис. 11). Интерфейсы MMC позволяют оснасткам совместно использовать общую хост-среду и обеспечивают интеграцию между приложениями. Консоль MMC не выполняет никаких функций управления.

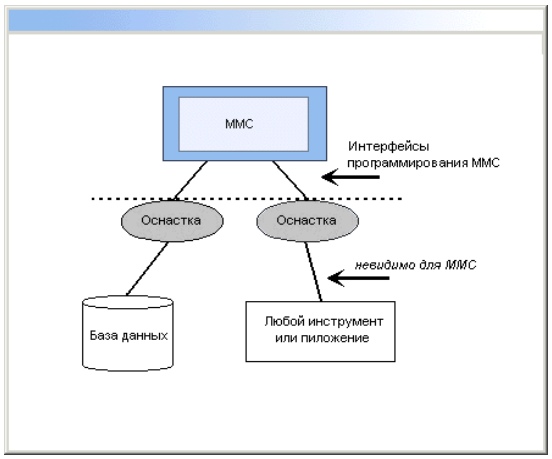

Рисунок 11 – Прикладные интерфейсы позволяют интегрировать оснастки с консолью

Компания Microsoft и независимые поставщики программного обеспечения могут разрабатывать инструменты управления для запуска в среде MMC.

Преимущества MMC

Возможность индивидуальной настройки и передачи полномочий. Помимо обеспечения интеграции и общей среды для административных инструментов, консоль MMC предоставляет возможность полностью индивидуальной настройки, так что администраторы могут создавать такие консоли управления, которые будут включать только необходимые им инструменты.

Интеграция и унификация. MMC обеспечивает общую среду, в которой могут запускаться оснастки, и администраторы могут управлять различными сетевыми продуктами, используя единый интерфейс, что упрощает изучение работы с различными инструментами.

Гибкость в выборе инструментов и продуктов. В среде MMC можно использовать различные инструменты и оснастки. Для использования в среде MMC оснастка должна поддерживать объектную модель компонентов (Component Object Model, COM) или распределенную COM (Distributed Component Object Model, DCOM). Это позволяет выбирать наиболее оптимальный продукт среди оснасток, причем гарантируется его полная совместимость со средой MMC.

Пользовательский интерфейс MMC

Консоль управления MMC имеет пользовательский интерфейс, позволяющий открывать множество документов (Multiple Document Interface, MDI). Интерфейс консоли MMC на примере оснастки *Управление компьютером* показан на рис. 12.

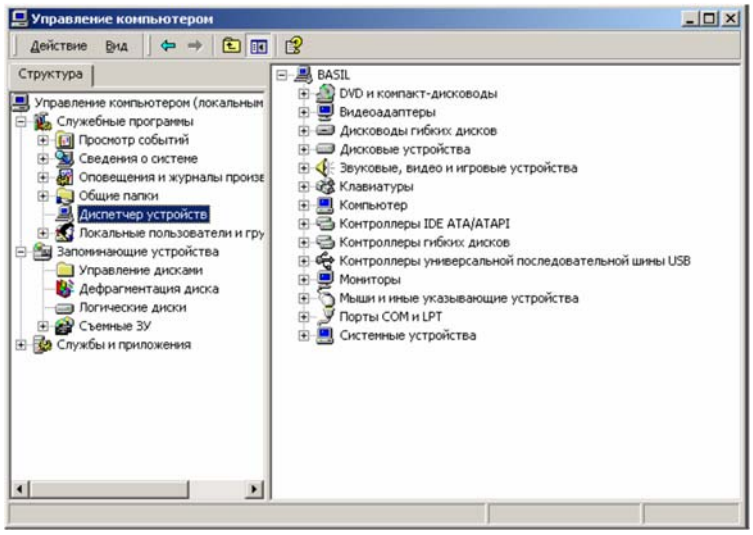

Рисунок 12 – Окно оснастки Управление компьютером

Родительское окно MMC имеет главное меню и панель инструментов. Главное

меню обеспечивает функции управления файлами и окнами, а также доступ к справочной системе.

Дочерние окна ММС представляют собой различные средства просмотра автономного документа консоли. Каждое из этих дочерних окон содержит панель управления, панель структуры и панель результатов, или сведений. Панель управления содержит меню и набор инструментов. Панель структуры отображает пространство имен инструментов в виде дерева, которое содержит все видимые узлы, являющиеся управляемым объектом, задачей или средством просмотра.

Панель результатов в дочернем окне отображает список элементов выбранного узла. Данный список может содержать папки, оснастки, элементы управления, веб-страницы, панели задач и другие элементы.

Оснастки и работа с ними

Все инструменты ММС состоят из совокупности оснасток. Каждая оснастка представляет собой минимальную единицу управления. С технической стороны оснастка представляет собой "OLE-сервер внутри процесса" (in-proc server – так часто называют DLL-библиотеки в модели COM), который выполняется в контексте процесса ММС. Оснастка может вызывать другие элементы управления и динамические библиотеки (DLL) для выполнения своей задачи.

Ряд оснасток могут быть объединены администратором в инструмент (также называется документом), который сохраняется в файле с расширением msc (Management Saved Console). Администратор использует инструменты для управления сетью. Файл *.msc можно затем передать другому администратору (например, по электронной почте), который сможет использовать содержащийся в нем инструмент на своем рабочем месте.

Благодаря возможности индивидуальной настройки ММС, администратор может создать идеальный инструмент на основе доступных оснасток. Каждый инструмент может иметь множество функций: например, возможности управления службой Active Directory, топологией репликации, доступом к файлам и т. д. В больших сетях администраторы могут иметь набор инструментов, организованных по категориям выполняемых с их помощью задач.

Типы оснасток

В ММС поддерживаются два типа оснасток:

Изолированная оснастка. Обеспечивает выполнение своих функций даже при отсутствии других оснасток, например, Управление компьютером.

Оснастка расширения. Может работать только после активизации родительской оснастки. Оснастки расширения могут предоставлять различные функциональные

возможности. Например, такие оснастки могут расширять пространство имен консоли, увеличивать число пунктов в меню или добавлять определенные мастера. В качестве примера можно привести оснастку Диспетчер устройств. Примеры оснасток, имеющиеся в системе Windows, описаны в таблице 4.

Таблица 3 – Оснастки Windows

Оснастка	Назначение
1. Анализ и настройка безопасности	Служит для управления безопасностью системы с помощью шаблонов безопасности
2. Групповая политика	Служит для назначения сценариев регистрации, групповых политик для компьютера и пользователей некоторого компьютера сети; позволяет просматривать и изменять политику безопасности, политику аудита и права пользователей
17. Управление дисками	Служит для управления дисками и защитой данных, для разбиения дисков на логические тома, форматирования, управления совместным доступом, квотами и т. д.
18. Управление компьютером	Предоставляет функции администрирования системы. Содержит в своем составе ряд изолированных оснасток и оснасток расширения
3. Дефрагментация диска	Служит для анализа и дефрагментации дисковых томов
4. Диспетчер устройств	Содержит список всех устройств, подключенных к компьютеру, и позволяет их конфигурировать
5. Локальные пользователи и группы	Служит для управления локальными учетными записями пользователей и групп
6. Общие папки	Отображает совместно используемые папки, текущие сеансы и открытые файлы
7. Оповещения и журналы производительности	Конфигурирует журналы данных о работе системы и службу оповещений
8. Папка	Служит для добавления новой папки в дерево
9. Просмотр событий	Служит для просмотра и управления системным журналом, журналами безопасности и приложений
10. Сведения о системе	Отображает информацию о системе
11. Сертификаты	Служит для управления сертификатами

3 Практикум по работе в операционной системе Windows

Практическое задание № 1 Работа с реестром

Теоретические сведения

Реестр (системный реестр) – это иерархическая база данных, содержащая записи, определяющие параметры и настройки операционных систем Microsoft Windows. Реестр в том виде, как он выглядит при просмотре редактором реестра, формируется из данных, источниками которых являются файлы реестра и информация об оборудовании, собранная в процессе загрузки.

В описании файлов реестра на английском языке используется термин "Hive". В некоторых работах его переводят на русский как "Улей". Microsoft в своих документах переводит это как "Куст".

Файлы реестра создаются в процессе установки операционной системы и хранятся в папке %SystemRoot%\system32\config (обычно C:\windows\system32\config). Для операционных систем Windows это файлы с именами: default, sam, security, software, system.

Папка хранения копии файлов реестра C:\windows\ Repair.

Для работы с содержимым системного реестра используется специальное программное обеспечение – редакторы реестра (REGEDIT.EXE, REGEDT32.EXE), являющиеся стандартными компонентами операционной системы. Для запуска реестра используется "Пуск">"Выполнить" – regedit.exe (рис. 13).

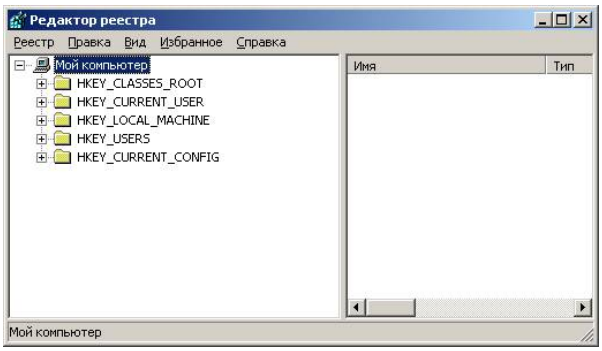

Рисунок 13 – Редактор реестра

В левой половине окна вы видите список корневых разделов (root keys) реестра. Каждый корневой раздел может включать в себя вложенные разделы (subkeys) и параметры (value entries).

Назначение корневых разделов представлено в таблице 1.

В процессе загрузки и функционирования операционной системы выполняется постоянное обращение к данным реестра как для чтения, так и для записи. Даже один

неверный параметр в реестре может привести к краху системы, как и нарушение целостности отдельных файлов. Поэтому, прежде чем экспериментировать с реестром, позаботьтесь о возможности его сохранения и восстановления.

Редактор реестра Windows (regedit.exe)

В состав ОС Windows входит программа для редактирования реестра – regedit.exe. Поскольку она располагается в системном каталоге, для ее запуска в командной строке не нужно указывать полный путь (например, достаточно будет такой последовательности: Пуск> Выполнить> regedit> OK (рис. 14).

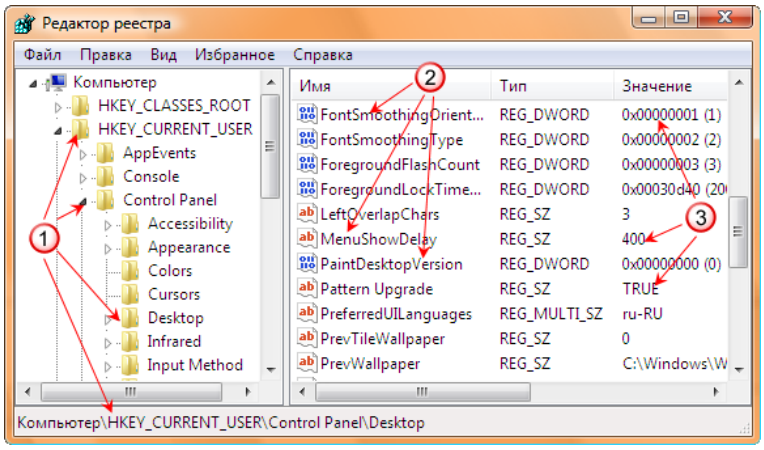

Рисунок 14 – Редактор реестра

① – разделы реестра;

② – параметры реестра;

③ – значения параметров.

В левой части окна находятся разделы реестра**,** их еще называют «ключи» (похожи на папки**)**, в правой части окна отображаются параметры (похожи на файлы) и их значения (рис. 15)**.**

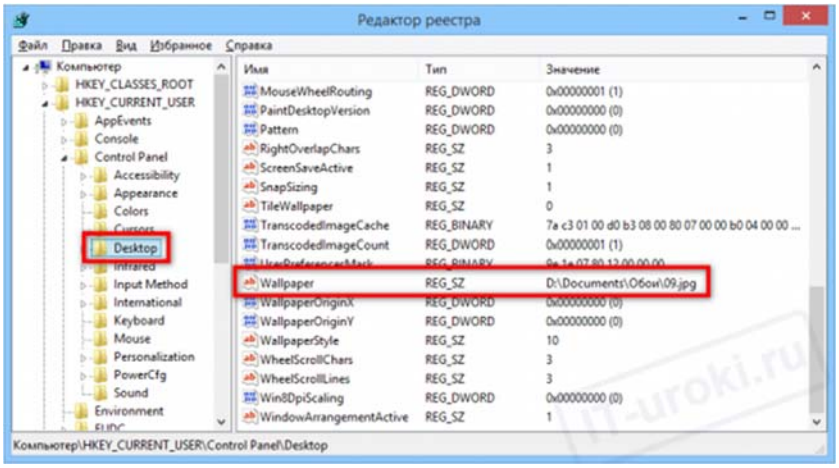

Рисунок 15 – Раздел реестра

Пример: в разделе «HKEY_CURRENT_USER\Control Panel\Desktop» есть параметр «Wallpaper» (обои рабочего стола), на изображении видно, что его значение «D:\\Documents\\Обои\\09.jpg». Таким образом, прописана картинка, которая отображается как фон рабочего стола.

Параметры могут быть разных типов, на изображении ниже пример всех возможных параметров для Windows 10 и 8 (для наглядности, имя параметра соответствует его типу) (рис. 16):

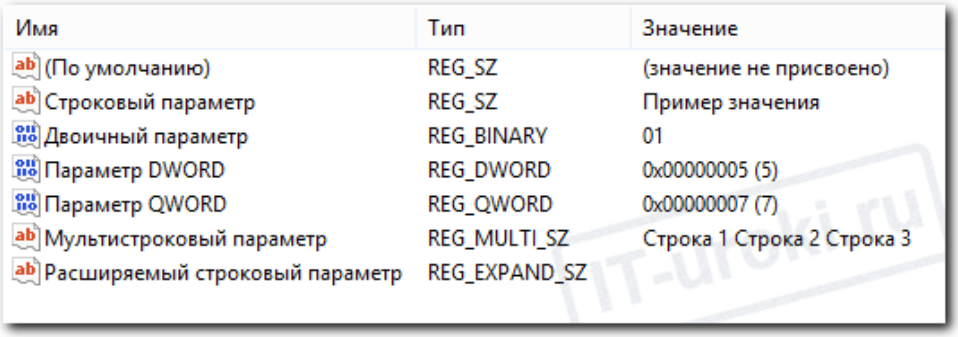

Рисунок 16 – Виды параметров реестра в Windows 10 и 8 (пример)

Изменяя значения параметров, как раз и производят изменения в настройках Windows и программ. Иногда приходится создавать новые параметры определенного типа, чтобы получить новую возможность в работе программы или операционной системы.

Запустите редактор реестра и просмотрите ветку реестра HKLM\SOFTWARE\Adobe\Acrobat Reader\5.0\InstalPath (рис. 17).

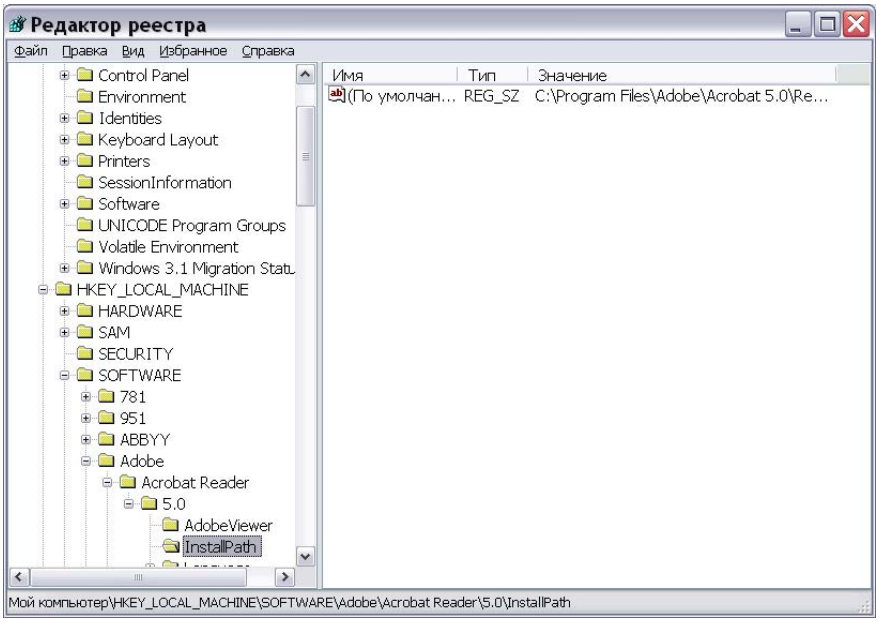

Рисунок 17 – Редактор реестра

Здесь хранится параметр с типом REG_SZ, так как это строковый параметр, и указан путь к файлу, который требуется запускать.

Создание ключей в реестре

Для создания ключа, необходимо сначала убедится в каком разделе мы находимся.

Обратите внимание на информационную панель редактора реестра (в самом низу). Там указан полный путь к ветке реестра в которой Вы находитесь:

Мой компьютер\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

Создать ключ реестра можно двумя способами:

1. В меню Правка выберите команду Создать, а затем укажите тип добавляемого параметра: Строковый параметр, Двоичный параметр, Параметр DWORD, Мульти-строковый параметр или Расширяемый строковый параметр. Введите имя нового параметра и нажмите клавишу ENTER.

2. Правой кнопки мышки щелкнуть в правой части редактора реестра и выбрать пункт меню "Создать". Далее выбираете тип добавляемого параметра: Строковый параметр, Двоичный параметр, Параметр DWORD, Мульти-строковый параметр или Расширяемый строковый параметр. Введите имя нового параметра и нажмите клавишу ENTER (рис. 18).

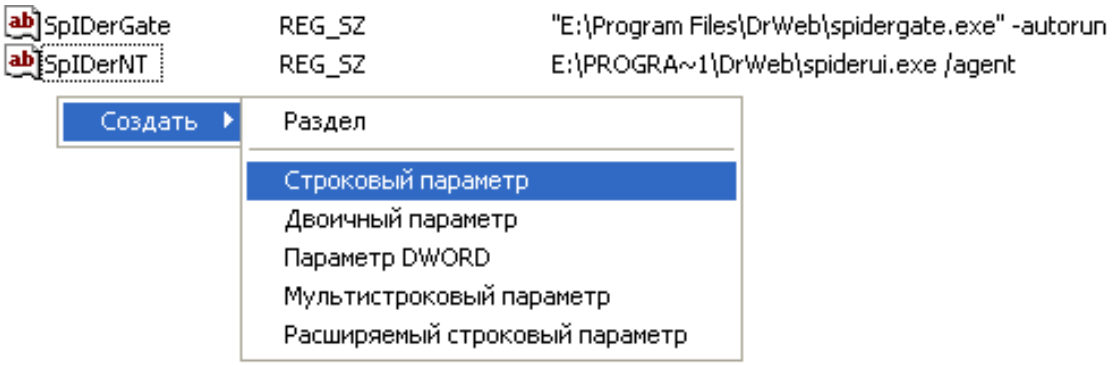

Рисунок 18 – Создание строкового параметра

Изменение ключей, разделов в реестре

1. В меню Правка выберите команду Изменить. В поле Значение введите новое значение параметра и нажмите кнопку OK.

2. Или же щелкаем правой кнопкой мышки по изменяемому параметру и выбираем пункт меню "Изменить". Вводим новые данные и нажимаем кнопку OK (рис. 19).

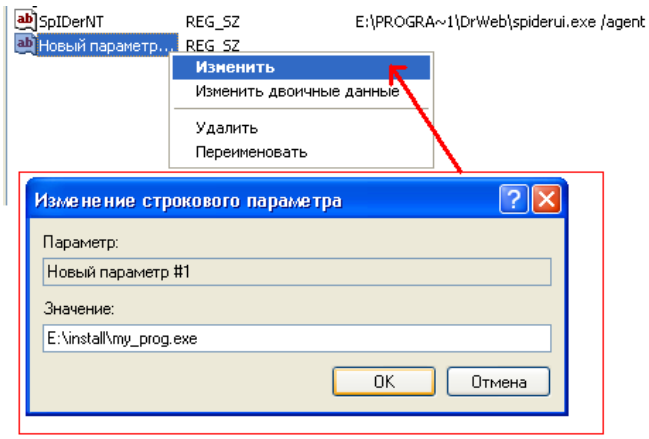

Рисунок 19 – Изменение ключей

Удаление разделов, ключей в реестре

1. Выберите удаляемый раздел или параметр. В меню Правка выберите команду Удалить.

2. Или же щелкаем правой кнопкой мышки по удаляемому параметру и выбираем пункт меню "Удалить" и подтверждаем удаление (рис. 20).

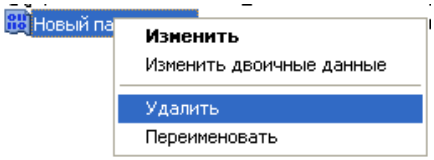

Рисунок 20 – Удаление раздела

Задание 1 Создание резервной копии

Порядок выполнения задания:

1. Сохранить файлы реестра в папке repear (http://regedit.readthedocs.org/work-whith-regedit.html).

Прежде чем вносить изменения в реестр, рекомендуется выполнить экспорт и создать его резервную копию. Можно сделать резервную копию как всего реестра в целом, так и отдельных разделов. Позже эту резервную копию можно импортировать, чтобы отменить внесенные изменения.

1 вариант. Чтобы создать полную копию реестра необходимо: запустить редактор реестра, как это описано выше, оставаясь в корне реестра открыть меню «Файл» и выбрать пункт «Экспорт…». Выбрать место, где будет сохранена резервная копия, и указать *«Имя файла» и* нажать на кнопку *«Сохранить»*.

2 вариант. Или, чтобы сохранить весь реестр на сменный носитель: копируется папка C:/Windows/System32/Config, а также файл Ntuser.dat, который находится по адресу: C:/Documents and Settings/<имя пользователя> (рис. 21).

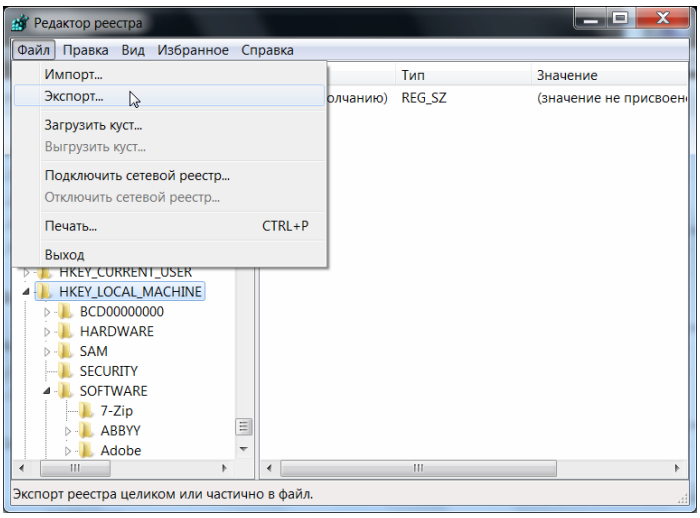

Рисунок 21 – Экспорт веток реестра. Создание резервной копии.

2. Сохраните раздел реестра.

Чтобы сэкономить место, можно сделать резервную копию отдельного раздела или подраздела. Для этого необходимо нажать правой кнопкой мыши на раздел (подраздел) и в выпадающем меню выбрать пункт *«Экспорт…»*. Также можно просто перейти в нужный раздел (подраздел) и выбрать меню *«Файл → Экспорт…»*. Экспортированный файл будет иметь расширение .reg.

Выполните. Выбрать ветку реестра HKEY_LOCAL_MACHINE\SOFTWARE\Adobe\Acrobat Reader\5.0\InstallPath1. В контекстном меню выбрать экспорт. В открывшемся окне указать путь (Мои документы) и имя файла (copy1.reg). При сохранении части реестра мы экспортировали данные в reg-файл.

3. Восстановите сохраненный раздел реестра.

В главном меню редактора выбрать Файл/Импорт с указанием пути к импортируемому файлу или просто запустить reg-файл, подтвердив импорт в реестр.

4. Чтобы создать раздел реестра необходимо выделить нужный раздел и в контекстном меню выбрать создать→раздел и ввести имя.

Выполните. К примеру, создать раздел в HKLM/SOFTWARE с именем «Программы».

Задание 2 Работа с реестром

1. Создать в реестре параметр, благодаря которому при каждой загрузке ОС будет автоматически запускаться блокнот.

Зайти в HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run, нажать правой кнопкой на папку «Run», выбрать «Создать»→«Строковый параметр». Указать любое имя. Нажать правой кнопкой по новому параметру и выбрать «Изменить», в поле значение ввести путь к блокноту: C:/Windows/notepad.exe.

2. Скрыть все значки с рабочего стола.

Зайти в HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer, создать параметр DWORD с именем NoDesktop и значением 1 (0 – все значки видны).

3. Изменить имя значка «Корзина» на «Мусор».

Зайти в HKEY_CLASSES_ROOT\CLSID\{645FF040-5081-101B-9F08-00AA002F954E}. Открыть параметр LocalizedString и в самом конце значения вместо «Корзина» написать «Мусор».

4. Создание своего окна при входе в систему

Это полезно тогда, когда требуется о чем-то предупредить пользователя. Раздел HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Winlogon

Параметры:

LegalNoticeCaption = например, "Внимание!" текст заголовка окна LegalNoticeText ="С 25-го по 30-е число каждого месяца необходимо сменить пароль" текст в окне.

5. Убрать слова «Ярлык для» из названия ярлыка.

Зайти в HKEY_CURRENT_USERS\Software\Microsoft\Windows\ CurrentVersion\Explorer и изменить параметр link на 00 00 00 00.

6. Зайти в HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\ CurrentVersion\Policies\Explorer и добавить следующие параметры:

NoLogOff=hex:01 00 00 00 (не dword, а hex) нет "Завершение сеанса <Имя>"

NoFind=dword:00000001 – нет пункта "Найти"

NoFavoritesMenu=dword:00000001 – нет "Избранное"

NoRecentDocsMenu=dword:00000001 – нет "Документы"

NoSetFolders=dword:00000001 нет "– Панели управления" в подменю "Настройка"

NoSetTaskbar=dword:00000001 – нет "Панель задач" в подменю "Настройка"

NoNetHood=dword:00000001 – нет "Сетевое окружение"

NoInternetIcon=dword:00000001 – нет значка "Интернет" на Рабочем столе Windows

NoTrayContextMenu=hex:01,00,00,00 – отключить меню, вызываемое правой кнопкой мыши на панели задач

NoViewContextMenu=hex:01,00,00,00 – отключить меню, вызываемое правой кнопкой мыши на Рабочем столе: Чтобы включить обратно, надо 01 заменить на 00.

ClearRecentDocsOnExit=hex:01,00,00,00 – не сохранять список последних открываемых документов по выходу из системы.

7. Сделать запрет запуска функции «Экран».

Чтобы запретить запуск функции Экран в Панели управления, надо создать в разделе HKEY_CURRENT_USER\Software\Microsoft\Windows\ CurrentVersion\Policies раздел System, а в нем параметр DWORD с именем NoDispCPL и установить его значение равным 1.

8. Восстановите реестр из папки repear.

Контрольные вопросы

Что такое системный реестр? Что такое куст? Какие основные разделы реестра вы знаете? Какая программа предназначена для редактирования реестра? Назовите типы данных для параметров реестра. Для чего нужен журнал транзакций? Какие способы есть для сохранения реестра?

Практическое задание № 2 Reg-файлы

Теоретические сведения

Reg-файл – это текстовый файл определенной структуры с расширением reg. При помощи reg-файла можно удалять, создавать ключи реестра и параметры с определенными значениями. Reg-файлы очень удобны для переноса настроек программ между компьютерами, создания резервных копий настроек программ с последующим их восстановлением за пару щелчков мыши.

Приложения часто содержат reg-файлы в своей группе файлов установки и используют их для регистрации информации конфигурирования. Любой пользователь может написать reg-файл (этап написания не представляет сложностей; опасной частью может оказаться результат пересылки этого файла в реестр).

Вы можете использовать reg-файлы, чтобы поручать администрирование реестров вашей системы. Ознакомившись с тем, как они действуют и что они делают, вы можете использовать их для управления пользователями, настройками программного обеспечения, настройками компьютеров или другими элементами, хранящимися в реестре.

Структура reg-файла

Файлы регистрации – это текстовые файлы с расширением reg, использующие следующий формат:

```
Имя инструментального средства
пустая строка
[Путь в реестре 1]
"Имя элемента данных 1"=Тип данных 1: значение 1
"Имя элемента данных 2"=Тип данных 2: значение 2
```

"Имя элемента данных 3"=Тип данных 3: значение 3
пустая строка
[Путь в реестре 2]
"Имя элемента данных 1"=Тип данных 1: значение 1
"Имя элемента данных 2"=Тип данных 2: значение 2
пустая строка

Имя инструментального средства. Первая строка идентифицирует средство, которое используется для выполнения этой процедуры:

для Windows Server 2003/2000/XP: Windows Registry Editor Version 5.00.

для всех версий Windows 9x/NT: REGEDIT4.

После этой строки следует пустая строка.

Путь в реестре. Путь в реестре к разделу, содержащему значения, которые вы импортируете, заключается в прямоугольные скобки, причем каждый уровень в иерархии отделяется обратным слешем, например, [HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\System]. У вас может быть несколько путей в файле регистрации.

Если нижний уровень иерархии, которую вы вводите в reg-файле, отсутствует в текущем реестре, то вы создаете новый подраздел. Содержимое файлов регистрации пересылается в реестр в порядке его ввода: если вы создаете новый раздел и подраздел в этом разделе, вводите строки в соответствующем порядке.

Последняя строка в файле должна быть пустой. Пустая строка обозначает начало нового пути реестра. Каждый раздел или подраздел является новым путем реестра. При наличии нескольких разделов в reg-файлы пустые строки облегчают анализ содержимого и устранение неполадок.

Данные. Данные, которые вы пересылаете в реестр, вводятся в следующем виде:

"Имя элемента данных"=Тип элемента данных: Значение элемента данных
Имя элемента данных заключается в кавычки.
После элемента данных непосредственно следует знак равенства (=).

Тип элемента данных непосредственно следует после знака равенства и заканчивается символом двоеточие (:).

Значение элемента данных должно вводиться в подходящем формате (строчный, шестнадцатеричный, десятичный или двоичный).

Вы можете ввести несколько строк элементов данных для одного пути в реестре, например,

"GroupPolicyRefreshTime"=dword:00000014
"GroupPolicyRefreshTimeOffset"=dword:0000000f

В этих двух строках представлены шестнадцатеричные значения для значений данных: 00000014 – это шестнадцатеричный эквивалент 20, и 0000000f – это шестнадцатеричный эквивалент 15. В реестре нет булева типа данных. Но вы можете пересылать булевы данные в реестр, используя в reg-файле элементы типа DWORD (4 байта) или STRING (2 байта), и при этом не обязательно вводить полную строку. Просто введите 1, и вы увидите в реестре значение 0x00000001(1).

В реестре существуют параметры "По умолчанию" ("Default"). Чтобы присвоить им какое-то значение через reg-файл, надо добавить такую строку:

@=Тип данных: значение

Здесь значок @ показывает, что у нас присваивается значение параметра "По умолчанию". Обратите внимание на то, что он не заключается в кавычки.

Удаление элементов реестра с помощью reg-файла

Вы можете также использовать reg-файл для удаления подразделов и элементов данных:

чтобы удалить подраздел, введите знак "минус" в начале имени этого подраздела

[-HKEY_LOCAL_MACHINE\Software\QuickSoft\QuickStart]

чтобы удалить отдельный элемент данных, введите знак "минус" вместо его значения ("Имя_элемента_данных=-")

```
[HKEY_CURRENT_USER\Software]
"xxx"= -
```

Слияние файла регистрации с реестром

Файлы регистрации используются путем слияния reg-файла с реестром, осуществляемого в Regedit. Имеется три способа пересылки содержимого этого файла в реестр:

- дважды щелкнуть на этом файле (действие по умолчанию для reg-файла – это слияние);
- ввести Regedit <имя_файла>.reg в командной строке;
- выбрать File\Import в линейке меню Regedit.

Если вы хотите запускать reg-файлы из командной строки в несопровождаемом режиме или хотите формировать пакетные файлы, с помощью которых происходит слияние reg-файлов с реестром без вмешательства пользователя, используйте команду Regedit со следующим синтаксисом: Regedit /s <имя_файла>.reg.

Во время пересылки .reg-файла в реестр происходят следующие действия:

- ✓ если путь в этом файле не указан, он добавляется;

✓ если какой-либо элемент данных еще не существует, он добавляется (вместе с его значением);

✓ если какой-либо элемент данных уже существует, его значение заменяется значением из reg-файла.

Файлы регистрации действуют, даже если вы применили групповую политику, отключающую средства редактирования реестра (в противном случае программы и система не могли бы вносить изменения в реестр, если это требуется).

Задание 1 Создание reg-файл в командной строке

Порядок выполнение задания

Написать в командной строке reg-файл, который создает или изменяет параметр в реестре. Задание выполняется в соответствии с вариантом.

Пример выполнения задания

В качестве примера напишем reg-файл отключающий меню недавних документов. Далее напишем bat-файл, который будет запускать на выполнение reg-файл:

1. В меню Пуск выбрать пункт Выполнить и ввести cmd.
2. Создать файл с расширением reg и записать в него необходимые значения:

```
Copy con 10.reg
Windows Registry Editor Version 5.00


[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Poli-
cies\ Explorer]
“NoRecentDocsMenu”=hex:01,00,00,00


^Z (нажать клавишу F6)
```

3. Создать файл с расширением bat, который будет запускать reg-файл (рис. 22).

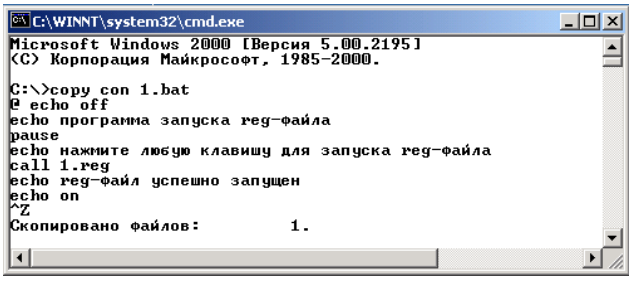

Рисунок 22 – Пример bat-файла

4. Запустите bat-файл на выполнение и положительно ответьте на вопрос (рис. 23).

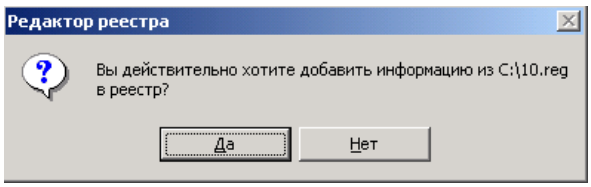

Рисунок 23 – Сообщение на добавление информации в реестр

После согласия появится сообщение, что данные успешно добавлены (рис. 24). Если же выйдет сообщение что данные не добавлены, значит, reg-файл написан неправильно. Следует еще раз прочитать теоретические сведения.

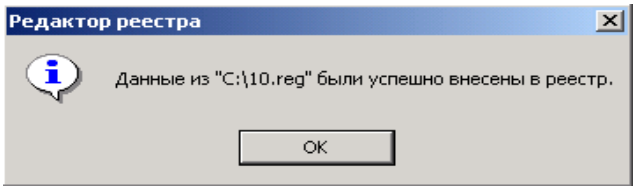

Рисунок 24 – Сообщение об успешном добавлении данных

Контрольные вопросы

Что такое файл регистрации? Для чего нужны файлы регистрации? Какова архитектура файла регистрации? Что происходит во время пересылки reg-файла в реестр? Какие ошибки могут возникнуть при написании файла регистрации? Что означает символ @? Как можно удалить элемент реестра при помощи файла регистрации?

Практическое задание № 3 Консоль управления Microsoft

Теоретические сведения

В операционных системах семейства Windows 2000/XP был кардинально изменен интерфейс управления операционной системой. В соответствии с новой концепцией Microsoft разработана единая среда управления *(*программа mmc.exe), получившая название *консоль управления Microsoft (Microsoft Management Console, MMC*).

Эта общая консоль управления разработана для запуска всех программных модулей администрирования, конфигурирования или мониторинга локальных компьютеров и сети в целом. Такие законченные программные модули называются оснастками (snap-in). Консоль управления сама по себе не выполняет никаких функций администрирования или конфигурирования, но служит в качестве рабочей среды для запуска оснасток. Фактически, Консоль Управления Microsoft представляет собой оболочку для работы оснасток, не выполняя при этом никаких функций управления.

Каждая оснастка представляет собой минимальную единицу управления. Ряд оснасток могут быть объединены администратором в инструмент (также называется документом), который сохраняется в файле с расширением .msc (Management Saved Console).

На практике термины инструмент и оснастка иногда могут использоваться как

взаимозаменяемые, поскольку некоторые инструменты MMC (и стандартные, и вновь созданные) содержат только одну оснастку. Поэтому чаще можно встретить фразу типа "данная функция реализуется при помощи оснастки...".

Благодаря возможности индивидуальной настройки MMC, администратор может создать требуемый ему инструмент на основе доступных оснасток. Инструмент может состоять из одной или нескольких оснасток, и каждая оснастка в свою очередь может содержать дополнительные оснастки расширения. Такая модульная структура позволяет администратору сохранять индивидуальные инструменты управления в отдельном файле (.msc) и в дальнейшем ими манипулировать.

Пользовательский интерфейс MMC

Консоль управления MMC имеет пользовательский интерфейс, позволяющий открывать множество документов (Multiple Document Interface, MDI). Интерфейс консоли MMC на примере оснастки Управление компьютером (Computer Management) показан на рисунке 25.

Для запуска оснастки Управление компьютером помимо стандартного способа (в Панели управления открыть группу Администрирование) можно использовать следующий:

щелкните правой кнопкой мыши на значке Мой компьютер (My Computer) на рабочем столе и выберите в контекстном меню пункт Управление (Manage).

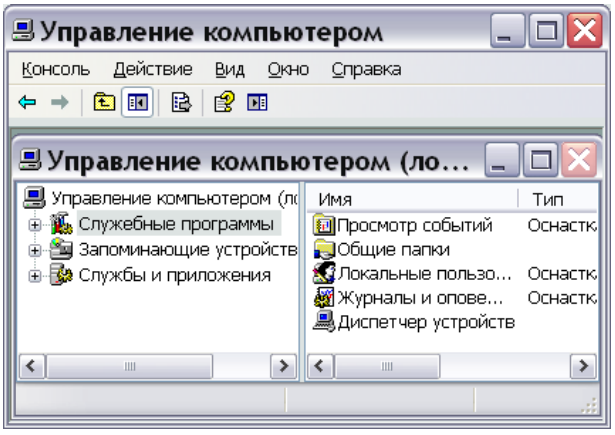

Рисунок 25 – Интерфейс консоли MMC на примере оснастки Управление компьютером

Рассмотрим консоль (и одноимённую оснастку) *Управление компьютером*, которая является основным средством администратора для конфигурирования компьютера.

В оснастке имеются три узла: Служебные программы (System Tools), Запоминающие устройства (Storage) и Службы и приложения (Services and applications). Данные узлы являются контейнерами и содержат ряд оснасток:

Служебные программы – содержит инструменты, предназначенные для администрирования компьютеров Windows. В данный узел входят оснастки:

– просмотр событий (Event Viewer);

– сведения о системе (System Information);

– оповещения и журналы производительности (Performance Logs and Alerts);

– общие папки (Shared Folders);

– диспетчер устройств (Device Manager);

– локальные пользователи и группы (Local Users and Groups).

Запоминающие устройства – узел содержит оснастки, служащие для управления дисками:

– управление дисками (Disk Management);

– дефрагментация диска (Disk Defragmenter);

– логические диски (Logical Drives);

– съемные ЗУ (Removable Storage).

Службы и приложения – узел содержит следующие оснастки:

– службы (Services);

– другие оснастки (например, DNS, DHCP, IIS) – в зависимости от того, какие дополнительные службы установлены в системе.

Задание 1 Создание консоли, состоящей из оснасток Управление компьютером и Сертификаты

Создание новой консоли, состоящей из оснасток *Управление компьютером и Сертификаты:*

Открыть окно *Консоль* с пустой консолью

1. В меню Пуск/Выполнить введите mmc. Откроется окно Microsoft Management Console с пустой консолью. По умолчанию консоль MMC открывается в авторском режиме, в котором можно создавать новые консоли и изменять созданные ранее административные инструменты (рис. 26);

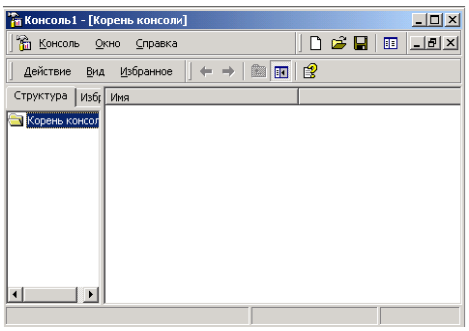

Рисунок 26 – Пустая консоль

Добавить оснастку *Управление компьютером*

2. В меню Консоль выберите пункт Добавить/Удалить оснастку. В открывшемся окне имеются две вкладки – автономные оснастки (изолированные) и оснастки расширения, которые будут добавлены в консоль (или уже включены в нее);

3. Нажмите кнопку Добавить. На экране появится окно Добавить автономные оснастки со списком автономных оснасток, имеющихся в системе. Выполните двойной щелчок на оснастке Управление компьютером и оставьте переключатель в положении Локальный компьютер. Затем нажмите кнопку Готово (рис. 27).

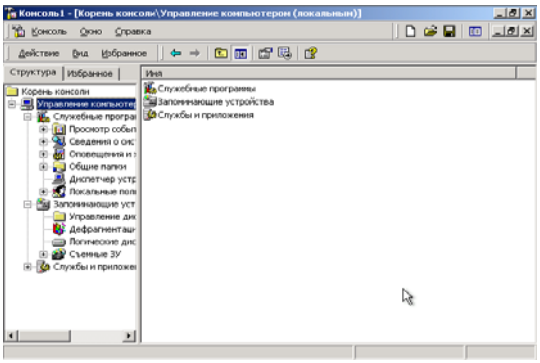

Рисунок 27 – Добавление оснастки

4. В окне оснасток выберите пункт Сертификаты и нажмите кнопку Добавить. Оснастка будет добавлена в список.

5. В окне Добавить/Удалить оснастку перейдите на вкладку Расширения. На этой вкладке выберите оснастку Управление компьютером из списка оснасток, которые могут быть расширены. Если вы не собираетесь подключать все оснастки-расширения, сбросьте флажок Добавить все расширения (который ставится по умолчанию) и снимите флажки с лишних оснасток. По окончании процедуры нажмите кнопку ОК.

6. Закройте окно выбора оснасток. Теперь окно консоли содержит две оснастки – Управление компьютером и Сертификаты.

7. Для того чтобы сохранить созданный инструмент, в меню Консоль выберите пункт Сохранить как и укажите имя файла и папку, в которой будет сохранен файл консоли (рис. 28).

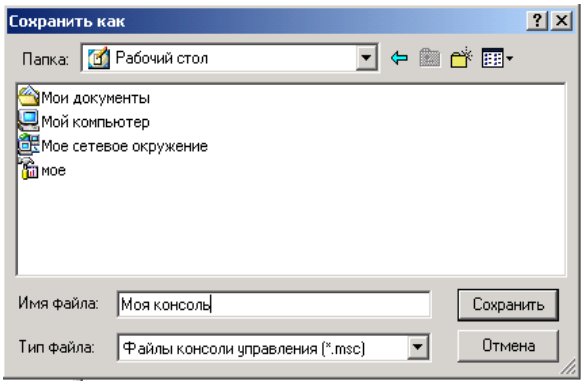

Рисунок 28 – Сохранение консоли

Индивидуальная настройка окон оснасток

После добавления оснасток можно развернуть окна оснасток, чтобы облегчить работу с ними. Для этого выполните следующие действия:

1. В левом подокне (в окне структуры) только что созданной консоли щелкните правой кнопкой мыши на узле Управление компьютером и выберите в контекстном меню Новое окно отсюда (New Window from Here). Будет открыто окно Управление компьютером, представляющее одноименную оснастку.

2. Аналогичные действия выполните для узла Сертификаты. В новом окне нажмите кнопку Скрытие или отображение дерева консоли или избранного (Show/Hide Console tree) на панели инструментов для того, чтобы скрыть панель структуры. Закройте окно, содержащее корень консоли.

3. В меню Окно (Windows) выберите команду Сверху вниз (Tile Horizontally). Консоль будет выглядеть, как показано на рисунке 29.

Примечание. Дочерние окна в окне консоли имеют панель инструментов с кнопками и раскрывающимися меню. Кнопки и команды этих меню применяются только к содержанию соответствующего окна.

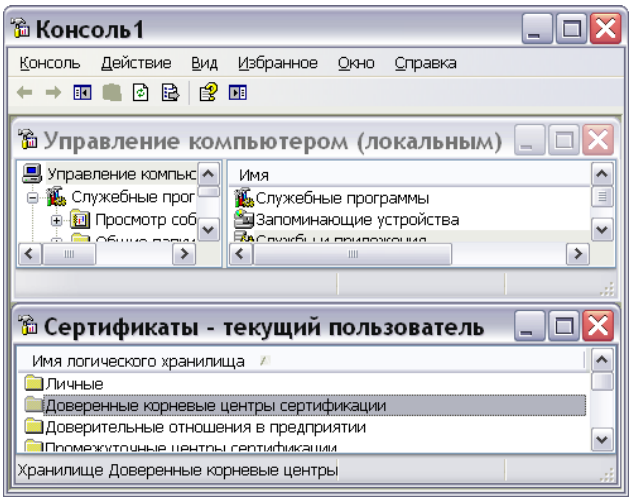

Рисунок 29 – Окно консоли с индивидуальной настройкой

Установка опций консоли

В меню *Консоль* выберите пункт *Параметры,* установите пункт *Пользовательский режим - полный доступ,* сохраните файл.

Для того чтобы сохранить созданный инструмент, в меню Консоль выберите пункт Сохранить как (Save As) и укажите имя файла и папку, в которой будет сохранен файл консоли (рис. 30).

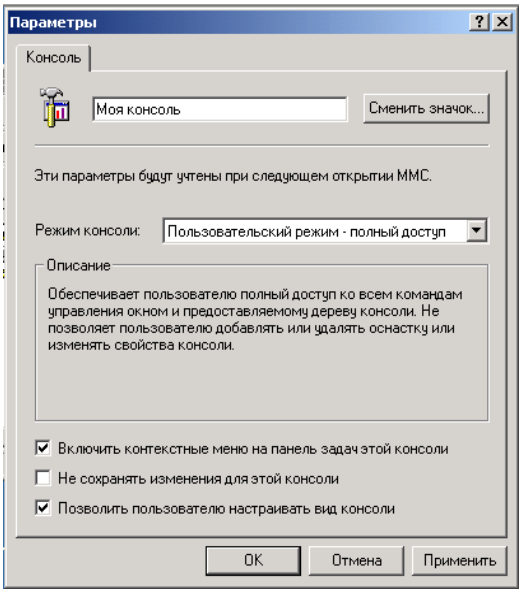

Рисунок 30 – Установка опций консоли

Задание 2 Самостоятельное по вариантам

Создайте консоль, добавив в нее необходимые оснастки по вариантам, выполните индивидуальную настройку окон оснасток и установите запрет на добавление новых оснасток. Варианты заданий и номера оснасток представлены в приложении 3.

Контрольные вопросы

Что такое консоль управления? Что такое оснастка? Какие оснастки вы знаете? Какие типы оснасток вы знаете? Какими преимуществами обладает консоль управления? С каким расширением сохраняется файл с рядом оснасток?

Практическое задание № 4 Работа с *оснасткой Локальные пользователи и группы*

Теоретические сведения

Учетные записи

Создание учетных записей и групп занимает важное место в обеспечении безопасности Windows NT, поскольку, назначая им права доступа, администратор получает возможность ограничить пользователей в доступе к конфиденциальной информации компьютерной сети, разрешить или запретить им выполнение в сети определенного действия, например, архивацию данных или завершение работы компьютера. Обычно право доступа ассоциируется с объектом – файлом или папкой. Оно определяет возможность данного пользователя получить доступ к объекту.

Оснастка Локальные пользователи и группы – это инструмент MMC, с помощью которого выполняется управление локальными учетными записями пользователей и групп – как на локальном, так и на удаленном компьютерах. С ним можно работать на рабочих станциях и автономных серверах Windows, как на изолированных,

так и рядовых членах домена.

На контроллерах домена Windows инструмент *Локальные пользователи и группы* недоступен, поскольку все управление учетными записями и группами в домене выполняется с помощью оснастки *Active Directory — пользователи и компьютеры*.

Запускать оснастку *Локальные пользователи и группы* может любой пользователь. Выполнять администрирование учетных записей могут только администраторы и члены группы *Опытные пользователи*. Окно изолированной оснастки Локальные пользователи и группы выглядит аналогично показанному на рис 31.

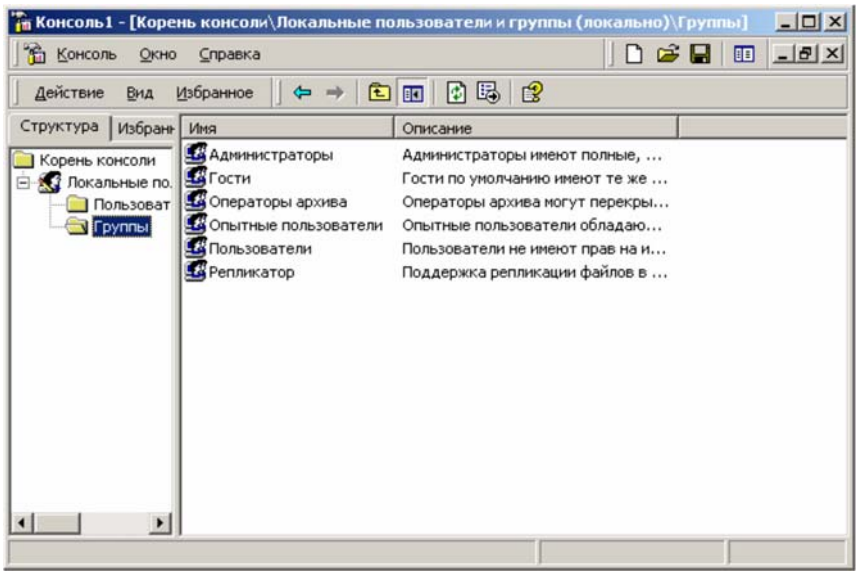

Рисунок 31 – Окно оснастки Локальные пользователи и группы

Папка Пользователи

Сразу после установки Windows папка Пользователи содержит две встроенные учетные записи – *Администратор* и *Гость*. Они создаются автоматически при установке. *Администратор* – эту учетную запись используют при установке и настройке рабочей станции или сервера, являющегося членом домена. Она не может быть уничтожена, блокирована или удалена из группы *Администраторы*, ее можно только переименовать.

Гость – эта учетная запись применяется для регистрации в компьютере без использования специально созданной учетной записи. Учетная запись *Гость* не требует ввода пароля и по умолчанию блокирована. Обычно пользователь, учетная запись которого блокирована, но не удалена, при регистрации получает предупреждение и входить в систему не может. Она является членом группы *Гости*. Ей можно предоставить права доступа к ресурсам системы точно так же, как любой другой учетной записи.

Папка Группы

Папка *Группы* содержит шесть встроенных групп.

Администраторы – ее члены обладают полным доступом ко всем ресурсам системы. Это единственная встроенная группа, автоматически предоставляющая своим членам весь набор встроенных прав.

Операторы архива – члены этой группы могут архивировать и восстанавливать файлы в системе независимо от того, какими правами эти файлы защищены. Кроме того, операторы архива могут входить в систему и завершать ее работу, но они не имеют права изменять настройки безопасности.

Гости – эта группа позволяет выполнить регистрацию пользователя с помощью учетной записи *Гость* и получить ограниченные права на доступ к ресурсам системы. Члены этой группы могут завершать работу системы.

Опытные пользователи – члены этой группы могут создавать учетные записи пользователей, но они имеют право модифицировать настройки безопасности только для созданных ими учетных записей. Кроме того, они могут создавать локальные группы и модифицировать состав членов созданных ими групп. То же самое они могут делать с группами *Пользователи*, *Гости* и *Опытные пользователи*. Члены группы *Опытные пользователи* не могут модифицировать членство в группах *Администраторы* и *Операторы архива*. Они не могут быть владельцами файлов, архивировать или восстанавливать каталоги, загружать и выгружать драйверы устройств и модифицировать настройки безопасности и журнал событий.

Репликатор – членом группы *Репликатор* должна быть только учетная запись, с помощью которой можно зарегистрироваться в службе репликации контроллера домена. Ее членами не следует делать рабочие учетные записи.

Пользователи – члены этой группы могут выполнять большинство пользовательских функций, например, запускать приложения, пользоваться локальным или сетевым принтером, завершать работу системы или блокировать рабочую станцию. Они также могут создавать локальные группы и регулировать состав их членов. Они не могут получить доступ к общему каталогу или создать локальный принтер.

Управление учетными записями

В качестве примера использования оснастки *Локальные пользователи и группы* для работы с учетными записями рассмотрим процедуру создания пользовательской учетной записи.

4.1 Создание учетной записи

Для создания учетной записи:

1. В оснастке *Локальные пользователи и группы* установите указатель мыши на папку *Пользователи* и нажмите правую кнопку. В появившемся контекстном меню выберите команду *Новый пользователь*.

2. Появится окно диалога *Новый пользователь*. В поле *Пользователь* введите имя создаваемого пользователя. В поле *Полное имя* введите полное имя создаваемого пользователя. В поле *Описание* введите описание создаваемого пользователя или его учетной записи. В поле *Пароль* введите пароль пользователя и в поле *Подтверждение* подтвердите его правильность вторичным вводом. Длина пароля не может превышать 14 символов.

3. Установите или снимите флажки *Потребовать смену пароля при следующем входе в систему*, *Запретить смену пароля пользователем*, *Срок действия пароля не ограничен* и *Отключить учетную запись*.

4. Чтобы создать еще одного пользователя, нажмите кнопку *Создать* и повторите шаги с 1 по 3. Для завершения работы нажмите кнопку *Создать* и затем *Закрыть*.

Имя пользователя должно быть уникальным для компьютера. Оно может содержать до 20 символов верхнего и нижнего регистра. Ниже приведены символы, применение которых в имени пользователя недопустимо:

" / \ I]:; I =, + *?<>

Изменение и удаление учетных записей

Изменять, переименовывать и удалять учетные записи можно с помощью контекстного меню, вызываемого щелчком правой кнопки мыши на имени пользователя, либо — меню *Действие* на панели меню оснастки *Локальные пользователи и группы* (при этом в правом подокне оснастки должна быть выбрана модифицируемая или удаляемая учетная запись пользователя).

Поскольку переименованная учетная запись сохраняет идентификатор безопасности (Security Identifier, SID), она сохраняет и все свои свойства, например, описание, полное имя пароля, членство в группах и т. д.

Создание учетной записи с помощью командной строки

Для создания учетной записи в командной строке используется команда net user.

Команда net user предназначена для создания и изменения учетных записей пользователей на компьютерах. При выполнении команды без параметров командной строки отображается список учетных записей пользователей, присутствующих на компьютере.

С командой net user используются следующие параметры:

– имя_пользователя

Имя пользователя учетной записи, которую необходимо добавить, удалить, изменить или просмотреть. Если вы хотите создать пользователя, в имени которого есть

пробелы, то вы должны заключить такое имя в кавычки.

– пароль

Определяет или изменяет пароль учетной записи пользователя. Длина пароля должна быть не меньше минимального допустимого значения, определяемого параметром /minpwlen команды net accounts, и не больше 14 символов.

– /domain

Операция выполняется для основного контроллера домена текущего домена. Этот параметр применим только для компьютеров под управлением Windows NT Workstation, которые являются членами домена Windows NT Server. Компьютеры под управлением Windows NT Server выполняют операции для основного котроллера домена по умолчанию.

– /add

Добавляет учетную запись пользователя в базу данных учетных записей пользователей.

– /delete

Удаляет учетную запись пользователя из базы данных учетных записей пользователей.

Чтобы просмотреть информацию об учетной записи нужно после команды net user указать имя пользователя.

Дополнительные параметры команды net user:

– /active:{yes | no}

Активирует или деактивирует учетную запись. Если учетная запись не активирована, пользователь не может получить доступ к серверу. По умолчанию учетная запись активирована.

– /fullname:"имя"

Полное имя пользователя (в отличии от имени учетной записи пользователя). Имя указывается в кавычках.

– /homedir:путь

Путь к домашней папке пользователя (существующий).

– /passwordchg:{yes | no}

Указывает, может ли пользователь изменять свой пароль (по умолчанию может).

– /passwordreq:{yes | no}

Указывает, должна ли учетная запись пользователя иметь пароль (по умолчанию должна).

4.2 Создание локальной группы

Для создания локальной группы:

1. В окне оснастки *Локальные пользователи и группы* установите указатель мыши на папке *Группы* и нажмите правую кнопку. В появившемся контекстном меню выберите команду *Новая группа*.

2. В поле *Имя группы* введите имя новой группы.

3. В поле *Описание* введите описание новой группы.

4. В поле *Члены группы* можно сразу же добавить пользователей и группы, которые войдут в данную группу: для этого нужно нажать кнопку *Добавит* и выбрать их в списке.

5. Для завершения нажмите кнопку *Создать* и затем *Закрыть*.

Имя локальной группы должно быть уникальным в пределах компьютера. Оно может содержать до 256 символов в верхнем и нижнем регистрах. В имени группы запрещено применение символа обратного слэша (\).

Изменение членства в локальной группе

Чтобы добавить или удалить учетную запись пользователя из группы:

1. В окне оснастки *Локальные пользователи и группы* щелкните на папке *Группы*.

2. В правом подокне установите указатель мыши на модифицируемую группу и нажмите правую кнопку. В появившемся контекстном меню выберите команду *Добавить в группу* или *Свойства*.

3. Для того чтобы добавить новые учетные записи в группу, нажмите кнопку *Добавить*. Далее следуйте указаниям окна диалога *Выбор: Пользователи или Группы*.

4. Для того чтобы удалить из группы некоторых пользователей, в поле *Члены группы* окна свойств группы выберите одну или несколько учетных записей и нажмите кнопку *Удалить*.

В локальную группу можно добавлять, как локальных пользователей, созданных на компьютере, так и пользователей и глобальные группы, созданные в домене, к которому принадлежит компьютер; или в доверяемых доменах.

Встроенные группы не могут быть удалены. Удаленные группы не могут быть восстановлены. Удаление группы не отражается на входящих в нее пользователей.

Создание локальной группы с помощью командной строки

Для создания локальной группы используется команда *net localgroup*. С командой net localgroup используются такие же параметры, как и с командной net user.

имя_группы

Задает имя локальной группы, которую необходимо добавить, расширить или удалить. Если указать только имя группы, то будет выведен список пользователей или глобальных групп в локальной группе.

Чтобы добавить учетную запись в локальную группу, то после имени группы надо написать имя пользователя.

Пример

Создадим учетную запись Masha с паролем 1234 и добавим ее только в группу Администраторы. Срок действия пароля неограничен (рис. 32).

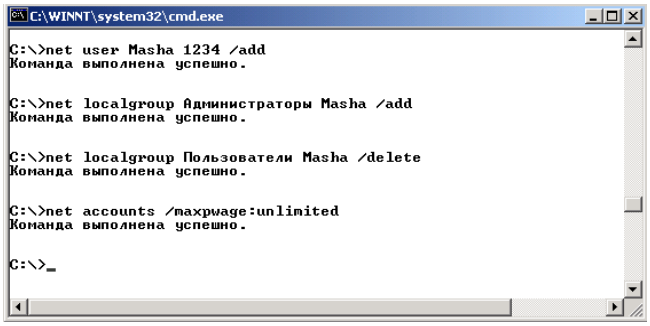

Рисунок 32 – Пример создания учетной записи

Давайте рассмотрим команды по порядку:

net user Masha 1234 /add – создает пользователя Masha с паролем 1234
net localgroup Администраторы Masha /add – добавляет пользователя Masha в группу Администраторы
net localgroup Пользователи Masha /delete – удаляет пользователя Masha из группы Пользователи (пользователь автоматически добавляется в нее при создании)
net accounts /maxpwage:unlimited – позволяет избежать истечения срока действия пароля (14 дней)

Теперь проверим что учетная запись создана и добавлена в нужную группу (рис. 33).

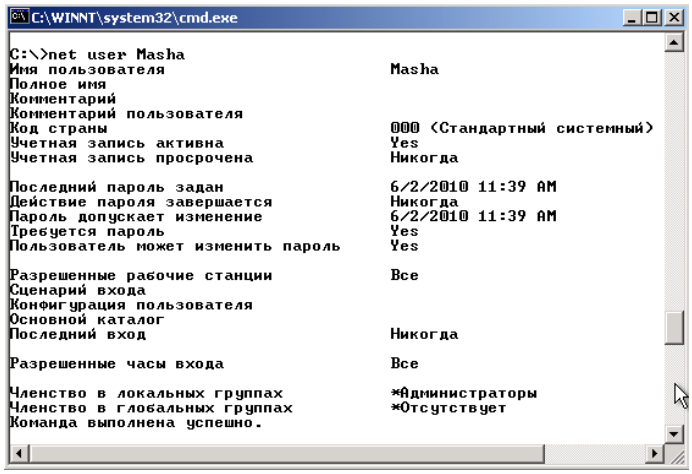

Рисунок 33 – Просмотр учетной записи Masha

4.3 Управление рабочей средой пользователя

Рабочая среда пользователя состоит из настроек рабочего стола, например, цвета экрана, настроек мыши, размера и расположения окон, из настроек процесса обмена информацией по сети и с устройством печати, переменных среды, параметров реестра и набора доступных приложений. Для управления средой пользователя предназначены следующие средства Windows:

1. *Сценарий входа в сеть* (сценарий регистрации) представляет собой командный файл, имеющий расширение .bat, или исполняемый файл с расширением .exe, который выполняется при каждой регистрации пользователя в сети. Сценарий может содержать команды операционной системы, предназначенные, например, для создания соединения с сетью или для запуска приложения. В состав системы Windows NT входит сервер сценариев Windows (Windows Script Host, WSH), используемый для создания сценариев.

2. *Профили пользователей*. В профиле пользователя хранятся все настройки рабочей среды компьютера, определенные самим пользователем. Это могут быть, например, настройки экрана и соединения с сетью.

3. *Сервер сценариев Windows*. Сервер сценариев независим от языка и предназначен для работы на платформах Windows. Он включает в себя как ядро сценариев Visual Basic Scripting Edition (VBScript), так и JScript. Сервер сценариев Windows предназначен для выполнения сценариев прямо на рабочем столе Windows или на консоли команд. При этом сценарии не надо встраивать в документ HTML.

Назначение сценариев входа учетным записям пользователей и групп

Для того чтобы назначить сценарий входа учетным записям пользователей и групп, с помощью оснастки *Локальные пользователи и группы* указывается путь к сценарию. Если при регистрации пользователя с помощью определенной учетной записи среди ее параметров указан путь к сценарию входа, соответствующий файл сценария открывается и выполняется.

На вкладке *Профиль* окна свойств учетной записи вы можете назначить сценарий входа, введя в поле *Сценарий входа* имя файла (и, возможно, относительный путь к нему). При регистрации сервер, аутентифицирующий пользователя, находит файл сценария (если таковой существует) с помощью указанного в учетной записи имени и пути. Если перед именем файла указан относительный путь, сервер ищет сценарий входа в подкаталоге основного локального пути сценариев.

Данные поля *Сценарий входа* определяют только имя файла и относительный путь, но не содержат сам сценарий входа. После создания файл сценария с определенным именем помещается в соответствующий реплицируемый (если компьютеры объединены в домен) каталог.

Сценарий входа можно поместить в локальный каталог компьютера пользователя. Но подобный подход, как правило, применяется только при администрировании учетных записей, существующих на одиночном компьютере, а не в домене. В этом случае вы должны поместить файл сценария в соответствии с локальным путем к сценариям входа в компьютер.

Помимо оснастки *Локальные пользователи и группы*, сценарии входа могут быть назначены пользователям или компьютерам и с помощью оснастки *Групповая политика*.

Профили пользователей

На изолированном компьютере с Windows NT локальные профили пользователей создаются автоматически. Информация локальных профилей необходима для поддержки настроек рабочего стола локального компьютера, характерных для конкретного пользователя. Профиль создается для каждого пользователя в процессе его первой регистрации в компьютере.

Профиль пользователя обладает следующими преимуществами:

1. При регистрации пользователя в системе рабочий стол получает те же настройки, какие существовали в момент предыдущего выхода пользователя из системы.
2. Несколько пользователей могут работать на одном и том же компьютере в индивидуальных средах.
3. Профили пользователей могут быть сохранены на сервере. В этом случае пользователь получает возможность работать со своим профилем при регистрации на любом компьютере сети. Такие профили называются перемещаемыми.

Структура профиля пользователя

Профиль пользователя создается на основе профиля, назначаемого по умолчанию. Он хранится на каждом компьютере, где работает Windows. Файл NTuser.dat, находящийся в папке Default User, содержит настройки конфигурации, хранящиеся в реестре Windows. Кроме того, каждый профиль пользователя использует общие программные группы, находящиеся в папке All Users.

Папки профиля пользователя

Как уже говорилось, при создании профиля пользователя используется профиль, назначаемый по умолчанию, находящийся в папке Default User. Папка Default User, папки профилей индивидуальных пользователей, а также папка All Users, находятся в папке Documents and Settings корневого каталога. В папке Default User находятся файл NTuser.dat и список ссылок на объекты рабочего стола. На рисунке показана структура папок локального профиля пользователя. В этих папках, в частности, хранятся ссылки на различные объекты рабочего стола (рис. 34).

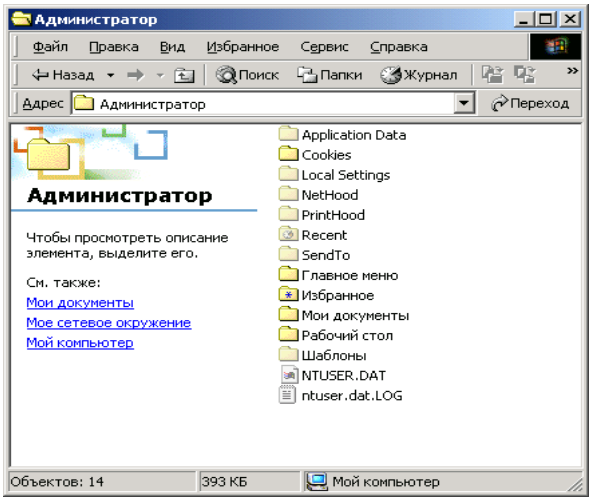

Рисунок 34 – Структура подпапок профиля пользователя

Папка All Users

Настройки, находящиеся в папке All Users, не копируются в папки профиля пользователя, но используются для его создания. Платформы Windows NT поддерживают два типа программных групп:

Общие программные группы. Они всегда доступны на компьютере, независимо от того, кто зарегистрирован на нем в данный момент. Только администратор может добавлять объекты к этим группам, удалять или модифицировать их.

Персональные программные группы. Они доступны только создавшему их пользователю.

Общие программные группы хранятся в папке All Users, находящейся в папке Documents and Settings. Папка All Users также содержит настройки для рабочего стола и меню *Пуск*.

Создание локального профиля пользователя

Локальный профиль пользователя хранится на компьютере в папке, имя которой совпадает с именем данного пользователя, находящейся в папке Documents and Settings. Если для данного пользователя не существует сконфигурированный перемещаемый (находящийся на сервере) профиль, то при первой регистрации пользователя в компьютере для него создается индивидуальный профиль. Содержимое папки Default User копируется в папку нового профиля пользователя. Информация профиля, вместе с содержимым папки All Users используется при конфигурации рабочей среды пользователя.

При завершении пользователем работы на компьютере все сделанные им изменения настроек рабочей среды, выбираемых по умолчанию, записываются в его профиль. Содержимое папки Default User остается неизменным.

Если пользователь имеет отдельную учетную запись на локальном компьютере

и в домене, для каждой из них создается свой профиль пользователя, поскольку регистрация на компьютере происходит с помощью различных учетных записей. При завершении работы все сделанные изменения также записываются в соответствующий данной учетной записи профиль.

Папка профиля пользователя на локальном компьютере содержит файл NTuser.dat и файл журнала транзакций с именем NTuser.dat.LOG. Он нужен для обеспечения отказоустойчивости.

Сохранение и перемещение профиля пользователя

Чтобы сохранить профиль пользователя требуется выполнить следующие действия:

1. Создайте папку на сетевом диске, в котором собираетесь хранить сетевые профили. Или же можно создать папку на локальном компьютере и в меню *Свойства* выбрать *Открыть общий доступ к этой папке*.

2. В *Панели управления*, дважды щелкните *Система*, затем щелкните на вкладке *Профили пользователей*. Под списком *Профили, хранящиеся на этом компьютере*, щелкните на профиле, который хотите скопировать, и нажмите *Копировать* (рис. 35).

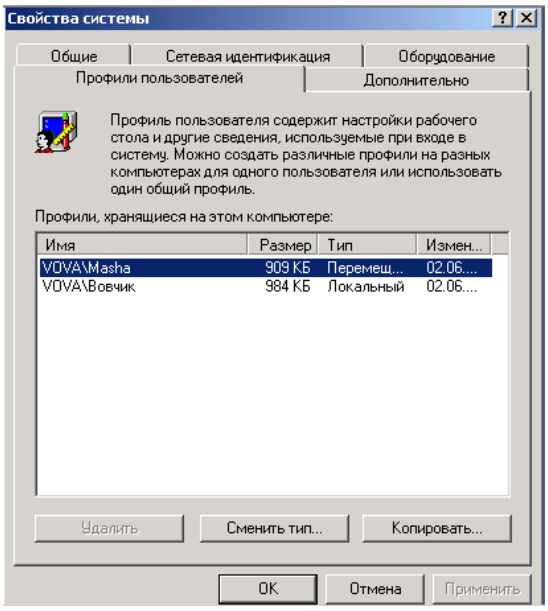

Рисунок 35 – Диалоговое окно выбора профиля пользователя

3. В диалоговом окне *Копировать профиль на*, введите путь к папке. Под *Разрешить использование*, нажмите *Изменить*.

4. Добавьте соответствующего пользователя, и нажмите *ОК*.

5. В папке, которую создали на сетевом диске или локальном диске, переименуйте файл Ntuser.dat в Ntuser.man, если это – принудительный профиль пользователя.

6. В списке пользователей дважды щелкните по учетной записи пользователя, затем в диалоговом окне *Свойства пользователя*, нажмите *Профиль*.

7. В строке *Путь к профилю*, напечатайте путь к папке профиля.

4.5 Задание для самостоятельной работы

1. Используя консоль, созданную в предыдущей лабораторной работе создать два профиля пользователя. Первый профиль – профиль администратора с вашим именем, пароль – ваша фамилия, входит в группу администраторы. Второй профиль – профиль пользователя, без пароля, имя – студент, входит в группу пользователи. В настройках обоих профилей запретить смену пароля. Настроить рабочий стол и панель задач для профилей. Создайте новую группу «Институт» и добавьте в нее учетную запись «студент».

2. Сохраните профиль администратора на сетевой диск. Перенесите этот профиль на учетную запись «студент».

3. Напишите bat-файл, которые будет создавать 3 учетных записи. Имена и пароли взять произвольные. Первую учетную запись добавить в группу администраторы, вторую – пользователи, третью – операторы архива. Создайте группу «работники» и добавьте туда третью учетную запись. Вывести список всех учетных записей и всех групп. Bat-файл должен содержать комментарии о том, какое действия выполняется или будет выполняться.

Контрольные вопросы

Что такое профиль пользователя? Для чего используется учетная запись Администратор? Перечислите стандартные группы пользователей и дайте их характеристику. Что такое профиль пользователя? Какими преимуществами обладает профиль пользователя? Где хранится профиль пользователя? Как создается индивидуальный профиль пользователя?

Практическое задание № 5 Работа с оснасткой Групповая политика

Теоретические сведения

Эффективное функционирование ни одной многопользовательской операционной системы невозможно без четкого разграничения доступа к ресурсам. Одним из средств, позволяющих настраивать параметры безопасной работы пользователей в сети в операционных системах Windows, являются *политики безопасности*. Политики безопасности в Windows реализуются с помощью средств групповых политик.

Групповые политики расширяют и используют преимущества Active Directory. Их настройки находятся в объектах групповых политик (Group Policy Object, GPO), которые в свою очередь ассоциируются с такими контейнерами Active Directory, как

сайты, домены и подразделения (организационные единицы). Политики безопасности Windows хранятся в двух типах объектов GPO: локальном объекте групповой политики и объекте групповой политики домена.

Схема именования GPO и его структура

При запуске оснастка *Групповая политика* загружает корневой узел, представляющий собой GPO, присоединенный к определенному контейнеру. Имя этого GPO и имя контейнера, к которому он присоединен, отображаются в окне структуры. Затем пространство имен подразделяется на два узла более низкого уровня — Конфигурация компьютера и Конфигурация пользователя. Используя их, можно создавать и настраивать групповые политики для компьютера и пользователей.

Узел Конфигурация компьютера

Узел содержит параметры всех политик, определяющих работу компьютера. Они регулируют функционирование операционной системы, вид рабочего стола, задают параметры выполняемых приложений, определяют работу средств обеспечения безопасности и т. д. Групповая политика применяется к компьютеру на этапе загрузки системы и в дальнейшем при выполнении циклов обновления.

Узел Конфигурация пользователя

Узел содержит параметры всех политик, определяющих работу пользователя на компьютере. Они регулируют вид рабочего стола, как и в предыдущем случае, задают параметры выполняющихся приложений, определяют работу средств обеспечения безопасности и пользовательских сценариев входа и выхода. Групповая политика применяется к пользователю при его регистрации на компьютере и в дальнейшем при выполнении циклов обновления.

Расширения оснастки Групповая политика

Ниже родительских узлов *Конфигурация компьютера* и *Конфигурация пользователя* находятся дочерние узлы, каждый из которых является полноценным расширением оснастки *Групповая политика*. Они могут находиться в обоих родительских узлах, хотя и с различными параметрами, или индивидуально расширять узлы *Конфигурация компьютера* или *Конфигурация пользователя*.

Оснастка *Групповая политика* имеет следующие расширения (рис. 36):

Административные шаблоны. Здесь находится групповая политика, определяющая параметры реестра, задающие работу и внешний вид рабочего стола, компонент операционной системы и приложений.

Параметры безопасности. Служит для настройки параметров системы безопасности компьютеров, на которые воздействует данный объект групповой политики. С помощью групповых политик можно настроить безопасность локального

компьютера, домена и целой сети.

Установка программ. Служит для централизованного управления программным обеспечением организации. С его помощью можно задавать различные режимы установки новых программ на компьютеры пользователей.

Сценарии. Сценарии используются для автоматического выполнения набора команд при загрузке операционной системы и в процессе завершения ее работы, а также при регистрации и отключении пользователя от сети. Для выполнения сценариев, написанных на Microsoft JScript и Microsoft Visual Basic Scripting Edition, можно применять сервер сценариев (Windows Scripting Host).

Перенаправление папок. Позволяет перенаправлять обращение к специальным папкам в сеть.

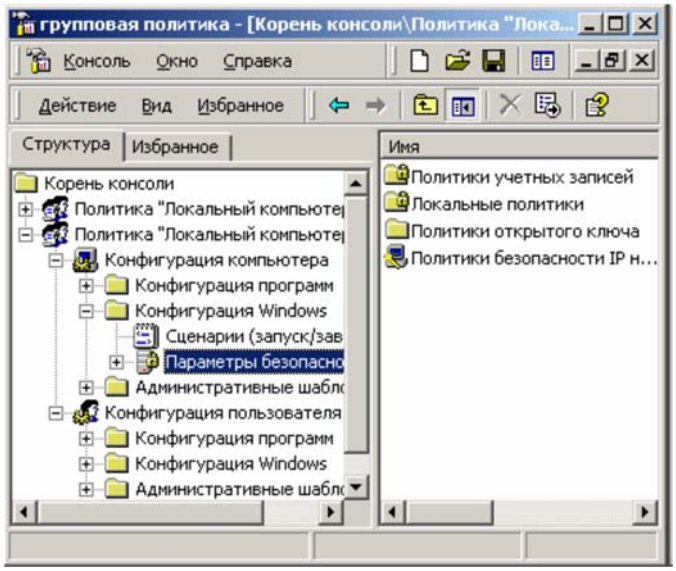

Рисунок 36 – Развернутое дерево оснастки *Групповая политика*, в котором можно видеть ее расширения (в виде узлов дерева)

Административные шаблоны

С помощью расширения *Административные шаблоны* администратор системы может настроить целый набор параметров реестра, задающих режим функционирования компонентов операционной системы и приложений.

Задать конкретные параметры, доступные для модификации с помощью интерфейса пользователя оснастки *Групповая политика*, можно с помощью специальных административных шаблонов. Модифицируемые значения параметров реестра, относящиеся к зарегистрированному в компьютере пользователю, записываются в раздел реестра HKEY_CURRENT_USER. Значения шаблонов, относящиеся к компьютеру, записываются в раздел HKEY_LOCAL_MACHINE.

Административный шаблон представляет собой текстовый файл в кодировке Unicode с расширением adm, информация которого определяет, как доступные для

модификации параметры реестра должны отображаться в окне интерфейса пользователя оснастки *Групповая политика*.

Кроме того, административные шаблоны задают разделы реестра, куда должны быть записаны модифицированные значения параметров, с их помощью проверяется допустимость вводимых значений параметров. В некоторых случаях с помощью шаблонов могут быть заданы значения параметров реестра, выбираемые по умолчанию.

Например, операционная система Windows 2000 содержит два файла административных шаблонов – system.adm и inetres.adm, где описаны все параметры реестра, доступные для изменения и отображаемые в расширении *Административные шаблоны* по умолчанию. Узел *Административные шаблоны* может быть расширен. Для этого администратор должен присоединить индивидуальный административный шаблон:

1. Установите указатель мыши на узел *Административные шаблоны* и нажмите правую кнопку.

2. В появившемся контекстном меню выберите команду *Добавление и удаление шаблонов*.

3. В окне *Добавление и удаление шаблонов* нажмите кнопку *Добавить*. Если вы на данном этапе хотите удалить ненужный шаблон, нажмите кнопку *Удалить*.

4. Если была нажата кнопка *Добавить*, появится окно диалога *Шаблоны политики*, в котором следует выбрать добавляемый шаблон и нажать кнопку *Открыть* (рис. 37).

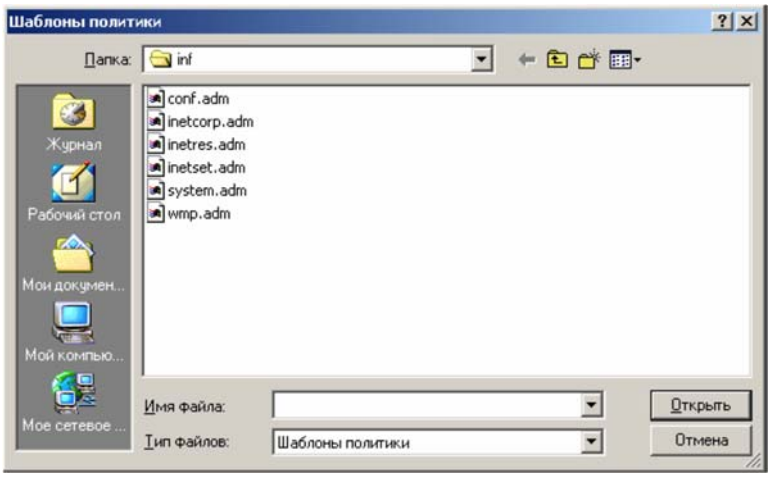

Рисунок 37 – Присоединение административного шаблона

5. В окне *Добавление и удаление шаблонов* нажмите кнопку *Закрыть*.

Внутри узла *Административные шаблоны* появятся дополнительные папки, соответствующие добавленному шаблону. С их помощью администратор может редактировать параметры дополнительного набора разделов реестра.

Параметры безопасности (Secyrity Settings)

С помощью расширения *Параметры безопасности* в GPO можно определить параметры политики безопасности, определяющие различные аспекты работы системы безопасности Windows.

Расширение *Параметры безопасности* позволяет настраивать следующие аспекты системы безопасности компьютера:

Политики учетных записей. Можно настраивать политики безопасности как учетных записей в масштабах домена, так и локальных учетных записей. Здесь определяются политика паролей, политика блокировки паролей.

Локальные политики. Можно настраивать политику аудита, назначать права пользователей и различные параметры безопасности, доступные для настройки в системе Windows.

Журнал событий. Можно настраивать политики безопасности, определяющие работу журналов событий приложений, системы и безопасности.

Группы с ограниченным доступом. Можно регулировать членство пользователей в специфических группах. Сюда обычно включают встроенные группы, такие как *Администраторы*, *Операторы архива* и другие, имеющие по умолчанию права администратора. В эту категорию могут быть включены и другие группы, безопасность которых требует особого внимания и членство, в которых должно регулироваться на уровне политики.

Системные службы. Можно настраивать безопасность и параметры загрузки для работающих на компьютере служб. В этом разделе могут быть использованы расширения, с помощью которых можно осуществлять настройку безопасности, специфическую для данной службы.

Реестр. Можно настраивать безопасность различных разделов реестра.

Файловая система. Можно настраивать безопасность определенных файлов.

Политики открытого ключа. Можно настраивать политики безопасности в отношении шифрования информации с помощью EFS, авторизации корневого сертификата в масштабах домена, авторизации доверенного сертификата и т. д.

Политики безопасности IP (IPSEC). Позволяет настраивать политику безопасности IP для компьютеров, находящихся в определенной области действия.

Для модификации настроек безопасности щелкните на папке *Параметры безопасности*, затем щелчками на соответствующих узлах откройте весь путь, ведущий к интересующим настройкам. В правом подокне окна оснастки *Групповая политика* двойным щелчком выберите настраиваемую политику и в открывшемся окне настройте ее.

Задание 1 Настройка параметров безопасности

Порядок выполнения заданий

С помощью оснастки *Групповая политика* выполнить следующие задания:

1. В политике учетных записей установить минимальную длину пароля 5 символов. Пароли должны отвечать требованиям сложности.

2. В конфигурации пользователя выбрать административные шаблоны и выполнить следующие действия:

- удалить все шаблоны;
- добавить шаблон system.adm;
- изучить появившиеся каталоги;
- удалить команду *Выполнить* из меня *Пуск*;
- запретить изменение параметров панели задач и меню *Пуск*;
- отключить хранение журнала недавно открытых документов;
- отключить панель управления;
- запретить установку и удаление программ;
- запретить изменение фонового рисунка рабочего стола;
- запретить доступ к свойствам подключений по локальной сети;
- запретить изменение пароля;
- запретить использование командной строки;
- сделать недоступными средства редактирования реестра;
- отключить оснастку управление дисками;
- Вернуть все измененные параметры на прежнее место;

Практическое задание № 6 Работа с консолью Управление компьютером

Теоретические сведения

Инструмент (и одноименная оснастка) Управление компьютером является одним из основных средств системного администратора и служит для:

- администрирования – оснастки Просмотр событий (Event Viewer), сведения о системе (System Information), Оповещения и журналы производительности (Performance Logs and Alerts), Общие папки (Shared Folders), Локальные пользователи и группы (Local Users and Groups);
- управления конфигурацией аппаратных средств – оснастка Менеджер устройств (Device Manager);
- управления конфигурацией дискового пространства (узел Запоминающие устройства).

Для запуска оснастки Управление компьютером можно воспользоваться папкой Администрирование в Панели управления или папкой Мой компьютер (My Computer*)* на Рабочем столе.

Оснастка Управление дисками

Оснастка Управление дисками в Windows позволяет работать наряду с базовым режимом хранения информации с новым типом устройств – устройствами с динамическим режимом хранения данных (dynamic storage). Диск, инициализированный для динамического хранения, называется динамическим диском. На нем могут находиться простые, составные, чередующиеся, зеркальные тома и тома RAID-5. Используя динамическое хранение, вы можете управлять дисками и томами без перезагрузки операционной системы. Пример окна оснастки Управление дисками приведен на рисунке 38.

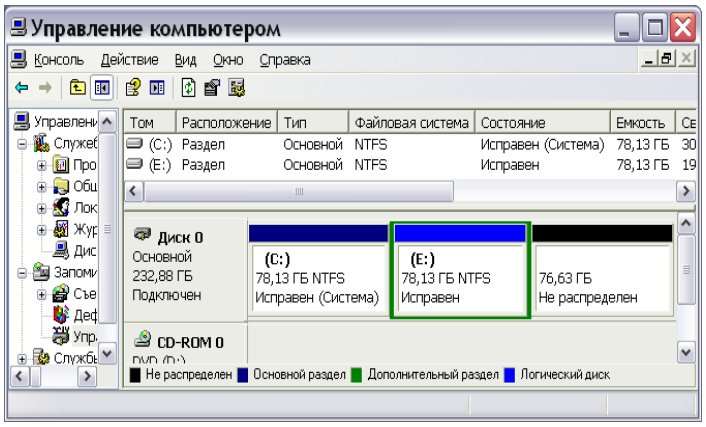

Рисунок 38 – Оснастка Управление дисками

Базовый режим хранения информации

Разделом является часть базового диска, функционирующая как физически автономная единица. Основной раздел (primary partition) зарезервирован для использования операционной системой. Каждый физический диск может иметь до четырех основных разделов (или до трех, если создан дополнительный раздел).

Дополнительный раздел (extended partition) создается с использованием оставшегося свободного пространства диска и может быть также разделен на логические устройства. На каждом физическом диске может быть только один дополнительный раздел.

Динамический режим хранения информации

Динамический диск подразделяется на тома, а не на разделы. Том состоит из одного или нескольких физических дисков в одной из следующих конфигураций: простой том, составной том, зеркальный том, чередующийся том и том RAID-5.

Том – это единица хранения, состоящая из свободного пространства на одном или нескольких дисках.

Простой том использует пространство одного диска. Это может быть один участок на диске или несколько участков, соединенных друг с другом. Составной том состоит из связанного вместе пространства нескольких дисков.

Зеркальный том – это средство обеспечения отказоустойчивости, где данные дублируются на двух физических дисках. Все данные одного диска копируются на дополнительный диск, что обеспечивает возможность получения избыточности данных.

Данные на чередующемся томе разбиваются при записи и помещаются на несколько физических дисков, причем информация равномерно распределяется среди всех дисков, входящих в состав такого тома.

Том RAID-5 является средством обеспечения отказоустойчивости дисковой системы, поскольку данные тома расщепляются при записи на три или большее количество дисков. Том RAID-5 обеспечивает избыточность информации, подсчитывая контрольную сумму информации, расположенной на каждом диске.

Свободное пространство – это неиспользованная и неформатированная часть жесткого диска, которая может быть использована при создании томов.

Системный том содержит файлы, жестко привязанные к оборудованию (Ntldr, Osloadenexe, Boot.hii, Ntdetect.com), необходимые для загрузки Windows.

Загрузочный том содержит файлы операционной системы Windows, расположенные в папках %SystemRoot% и %SystemRoot% \System32.

Задание 1 Инициализация диска

После присоединения к компьютеру диск необходимо инициализировать. Только после этого на нем можно создавать тома и разделы. Для инициализации диска:

1. Запустите оснастку Управление дисками.

2. В меню Действие (Action) выберите команду Повторить сканирование дисков (Rescan Disks).

Новые диски, присоединенные к компьютеру, подключаются как базовые. Впоследствии базовые диски могут быть превращены в динамические:

3. Укажите нужный базовый диск и нажмите правую кнопку мыши. В появившемся контекстном меню выберите команду Обновление до динамического диска (Upgrade to Dinamic Disk).

Возможно и обратное превращение. Однако динамические тома нельзя непосредственно конвертировать в разделы: предварительно все тома на диске придется удалить. Чтобы превратить динамический диск в базовый:

укажите динамический диск и нажмите правую кнопку мыши. В появившемся

контекстном меню выберите команду Возвратить к базовому диску (Revert to Basic Disk).

Создание базовых разделов

Создать новый раздел можно только в том случае, если на жестком диске компьютера осталось свободное пространство. Для создания базового раздела:

1. В окне оснастки Управление дисками щелкните на свободном пространстве диска (помеченном на экране как Свободно (Unallocated)).

2. В меню Действие выберите команду Создать (New). В появившемся меню выберите команду Раздел (Partition).

3. Откроется начальное окно Мастера создания раздела (Create Partition Wizard). Прочтите выведенный в нем текст и нажмите кнопку Далее.

4. В следующих окнах мастера вы можете сообщить тип раздела (основной (Primary) или дополнительный (Extended)), имя устройства и параметры форматирования.

5. Сообщив всю необходимую информацию, нажмите кнопку Готово (Finish) в последнем окне мастера. В результате будет создан новый раздел.

При создании раздел необходимо отформатировать. При форматировании может быть задано несколько файловых систем: FAT 16, FAT32 или NTFS 5.0.

Приложение Стандартные консоли и оснастки

На основе наиболее часто используемых оснасток были созданы стандартные консоли, список которых представлен ниже. Стандартные консоли (файл с расширением .msc) хранятся в каталоге C:\MINDOWS\system32:

certmgr.msc - *Сертификаты*. Позволяет управлять пользовательскими сертификатами, сертификатами удаленной или локальной службы, а также сертификатами локального или удаленного компьютера.

compmgmt.msc* - *Управление Компьютером*. Данная консоль является набором из нескольких оснасток, которые по отдельности входят в консоли fsmgmt.msc, devmgmt.msc, perfmon.msc, services.msc, taskschd.msc, WmiMgmt.msc. Консоль можно вызывать непосредственно из меню Пуск\Панель Управления\Администрирование, т. е. она включена в пользовательский интерфейс при инсталляции системы;

devmgmt.msc - *Диспетчер устройств*. Позволяет просмотреть информацию об установленном на вашем компьютере оборудовании, просмотреть список используемых ими драйверов (или откатить недавно переустанавливаемые драйверы);

diskmgmt.msc - *Управление дисками*. Она предназначена для работы с разделами жесткого диска компьютера.

eventvwr.msc - *Просмотр событий*. Позволяет получить доступ к системным журналам, содержащим сведения о работе компонентов операционной системы и сторонних программ.

gpedit.msc - *Редактор объектов групповой политики*. Предназначена для редактирования групповых политик.

gpmc.msc - *Управление групповой политикой*. Данная консоль может использоваться только на компьютерах, подключенных к домену Active Directory. Она представляет собой более функциональную оснастку, чем стандартная оснастка Редактор объектов групповой политики.

lusrmgr.msc - *Локальные пользователи и группы*. Позволяет добавлять, удалять или редактировать содержимое групп и отдельных учетных записей пользователей.

perfmon.msc - *Монитор надежности и производительности*. Она предназначена для наблюдения за производительностью и стабильностью работы компьютера

rsop.msc - *Результирующая политика*. Упрощает процесс определения групповых политик, которые действуют на конкретного пользователя.

secpol.msc - *Локальная политика безопасности*. Позволяет просматривать и изменять политику безопасности, политику аудита и права пользователей.

services.msc - C*лужбы*. Запускает, останавливает и конфигурирует службы (сервисы) Windows.

wmiMgmt.msc - *Элемент управления wmi*. Позволяет настроить параметры работы инструментария управления Windows на локальном или удаленном компьютере.

Рассмотрим консоль (и одноимённую оснастку) *Управление компьютером*, которая является основным средством администратора для конфигурирования компьютера.

В оснастке имеются три узла: Служебные программы (System Tools), Запоминающие устройства (Storage) и Службы и приложения (Services and applications). Данные узлы являются контейнерами и содержат ряд оснасток: Служебные программы - содержит инструменты, предназначенные для администрирования компьютеров Windows 2000. В данный узел входят:

Просмотр событий (Event Viewer)

Сведения о системе (System Information)

Оповещения и журналы производительности (Performance Logs and Alerts)

Общие папки (Shared Folders)

Диспетчер устройств (Device Manager)

Локальные пользователи и группы (Local Users and Groups)

Запоминающие устройства - узел содержит оснастки, служащие для управления дисками:

Управление дисками (Disk Management)

Дефрагментация диска (Disk Defragmenter)

Логические диски (Logical Drives)

Съемные ЗУ (Removable Storage)

Службы и приложения - узел содержит следующие оснастки:

Управляющий элемент WMI (WMI Control)

Службы (Services)

Служба индексирования (Indexing Service)

Телефония (Telephony) (на Windows 2000 Server)

Другие оснастки (например, DNS, DHCP, IIS) - в зависимости от того, какие дополнительные службы установлены в системе.

Таблица – Оснастки, имеющиеся в Windows

Оснастка	Назначение
1. Анализ и настройка безопасности	Служит для управления безопасностью системы с помощью шаблонов безопасности

2. Групповая политика	Служит для назначения сценариев регистрации, групповых политик для компьютера и пользователей некоторого компьютера сети; позволяет просматривать и изменять политику безопасности, политику аудита и права пользователей
3. Дефрагментация диска	Служит для анализа и дефрагментации дисковых томов
4. Диспетчер устройств	Содержит список всех устройств, подключенных к компьютеру, и позволяет их конфигурировать
5. Локальные пользователи и группы	Служит для управления локальными учетными записями пользователей и групп
6. Общие папки	Отображает совместно используемые папки, текущие сеансы и открытые файлы
7. Оповещения и журналы производительности	Конфигурирует журналы данных о работе системы и службу оповещений
8. Папка	Служит для добавления новой папки в дерево
9. Просмотр событий	Служит для просмотра и управления системным журналом, журналами безопасности
10. Сведения о системе	Отображает информацию о системе
11. Сертификаты	Служит для управления сертификатами
12. Системный монитор	Используется для сбора и просмотра в реальном времени данных, характеризующих работу памяти, дисков, процессора и других компонентов системы
13. Служба индексирования	Служит для индексирования документов различных типов с целью ускорения их поиска
14. Служба компонентов	Конфигурирует и управляет службами компонентов COM+
15. Службы	Запускает, останавливает и конфигурирует службы (сервисы) Windows
16. Ссылка на ресурс веб	Служит для подключения веб-страниц (html, asp, stml)
17. Управление дисками	Служит для управления дисками и защитой данных, для разбиения дисков на логические тома, форматирования, управления совместным доступом, квотами и т. д.
18. Управление компьютером	Предоставляет функции администрирования системы. Содержит в своем составе ряд изолированных оснасток и оснасток расширения
19. Управление политикой безопасности IP	Служит для управления политиками IPSec для безопасного соединения с другими компьютерами
20. Управление службой факсов	Служит для управления службой и устройствами факсимильной связи
21. Управление съемными носителями	Служит для управления сменными носителями информации
22. Управляющий элемент	Служит для конфигурирования средств Windows Management
23. Шаблоны безопасности	Обеспечивает возможность редактирования файлов-шаблонов безопасности
24. Элемент ActiveX	Подключение к дереву консоли различных элементов управления ActiveX
Active Directory - домены и доверие	Служит для управления доменами и доверительными отношениями
Active Directory - пользователи и компьютеры	Управляет пользователями, группами, организационными единицами и другими объектами AD
Active Directory - сайты и службы	Определяет топологию и расписание репликации AD. Обеспечивает изменение служб корпоративного уровня Windows 2000
Маршрутизация и удаленный доступ	Служит для управления маршрутизацией и удаленным доступом
Политика безопасности домена	Служит для управления политиками для всего домена. Фактически, представляет собой оснастку Групповая политика, настроенную на работу с конкретным доменом