**Задание на 11.02.2022**

МДК 03.02 Программно-аппаратные средства защиты информации

1. **Сделать конспект лекции**
2. **Сделать практическую работу №21 Виртуальные частные сети**
3. **Сделать практическую работу №22 Средства и методы обеспечения целостности данных в СУБД**
4. **Отчет отправить на почту** [**svebalch@mail.ru**](mailto:svebalch@mail.ru)

**СРС:** [**https://stepik.org/course/94299/promo**](https://stepik.org/course/94299/promo) **пройти онлайн курсы на платформе Степик. Тема: Кибербезопасность**

**Лекция 23**

**Защита сетевого файлового ресурса**. **Защита файловой системы WINDOWS 2000. Шифрование.**

**1. Разрешения для файлов и папок**

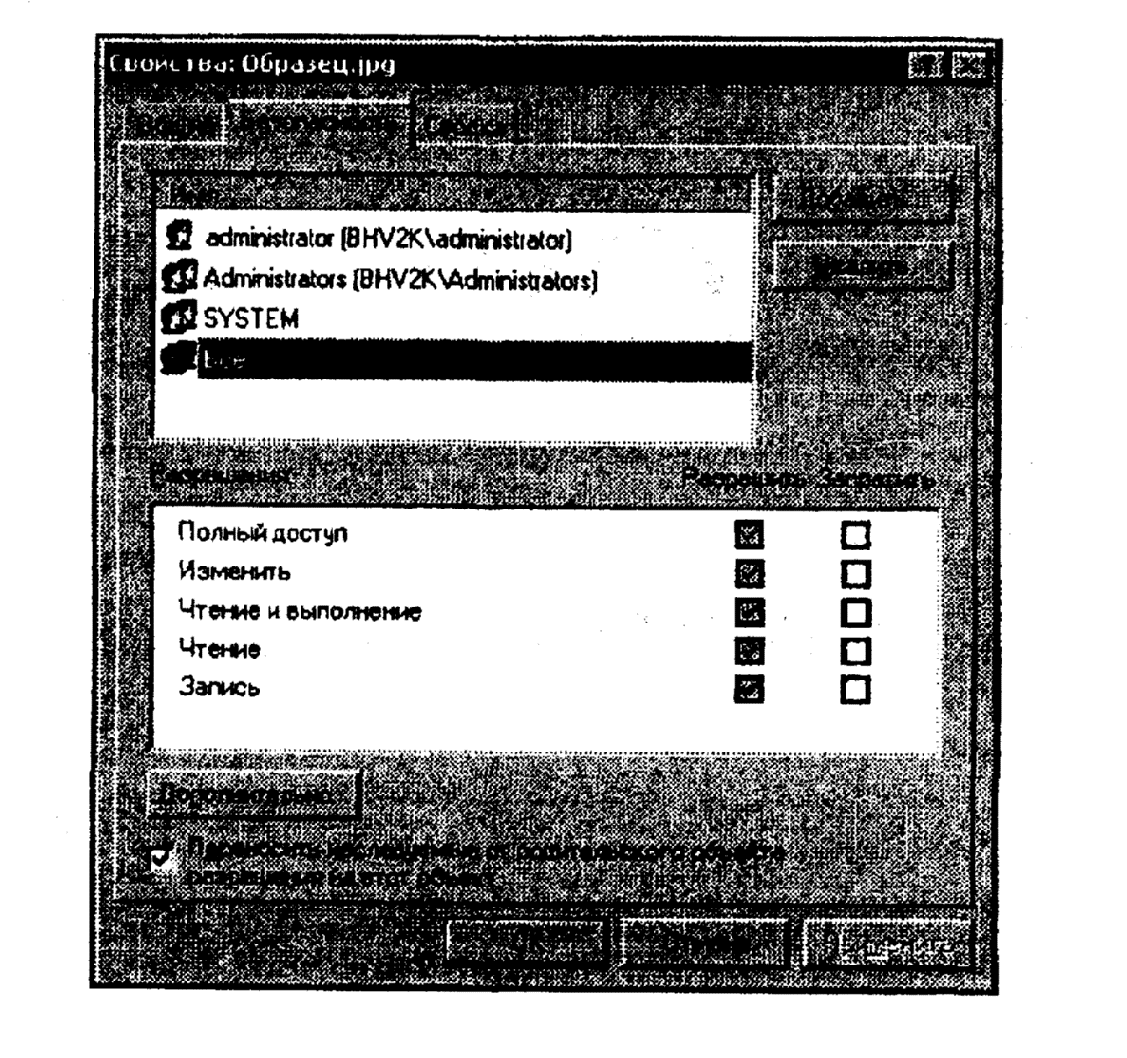
Устанавливая пользователям определенные *разрешения* для файлов и каталогов (папок), администраторы системы могут защищать конфиденциальную информацию от несанкционированного доступа. Каждый пользователь должен иметь определенный набор разрешений на доступ к конкретному объекту файловой системы. Кроме того, он может быть *владельцем* файла или папки, если сам их создает. Администратор может назначить себя владельцем любого объекта файловой системы (файла или папки). Следует, однако, помнить, что обратная передача владения от администратора к пользователю невозможна. Администратор должен зарегистрироваться в системе под именем того пользователя, которому он хочет передать владение файлом или папкой, а затем стать владельцем нужного объекта.

Разрешения пользователя на доступ к объектам файловой системы работают по принципу *дополнения* (аддитивности). Это значит, что *действующие* разрешения, то есть те разрешения, которые пользователь реально имеет в отношении конкретного каталога или файла, образуются из всех *прямых и косвенных* разрешений, назначенных пользователю для данного объекта с помощью логической функции ИЛИ. Например, если пользователь имеет прямо назначенное разрешение для каталога на чтение, а косвенно, через членство в группе, ему дано разрешение на запись, то в результате пользователь сможет читать информацию в файлах каталога и записывать в них данные. Следует все же заметить, что правило сложения разрешений с помощью логического ИЛИ не выполняется, когда пользователь имеет определенное разрешение, а группе, в которую он входит, отказано в этом разрешении (или наоборот). В этом случае отказ в разрешении имеет более высокий приоритет над предоставлением разрешения, т.е. в результате пользователь не будет иметь данного разрешения. Появление возможности отказа пользователю или группе в разрешении для файлов и каталогов сделало ненужным разрешение No Ассеss, применявшееся в Windows NT 4.0. Теперь для отказа пользователю в разрешении на доступ к какому-либо файлу или папке следует включить пользователя в группу, которой отказано в разрешении «Полный доступ (Full Control) для данного объекта файловой системы».

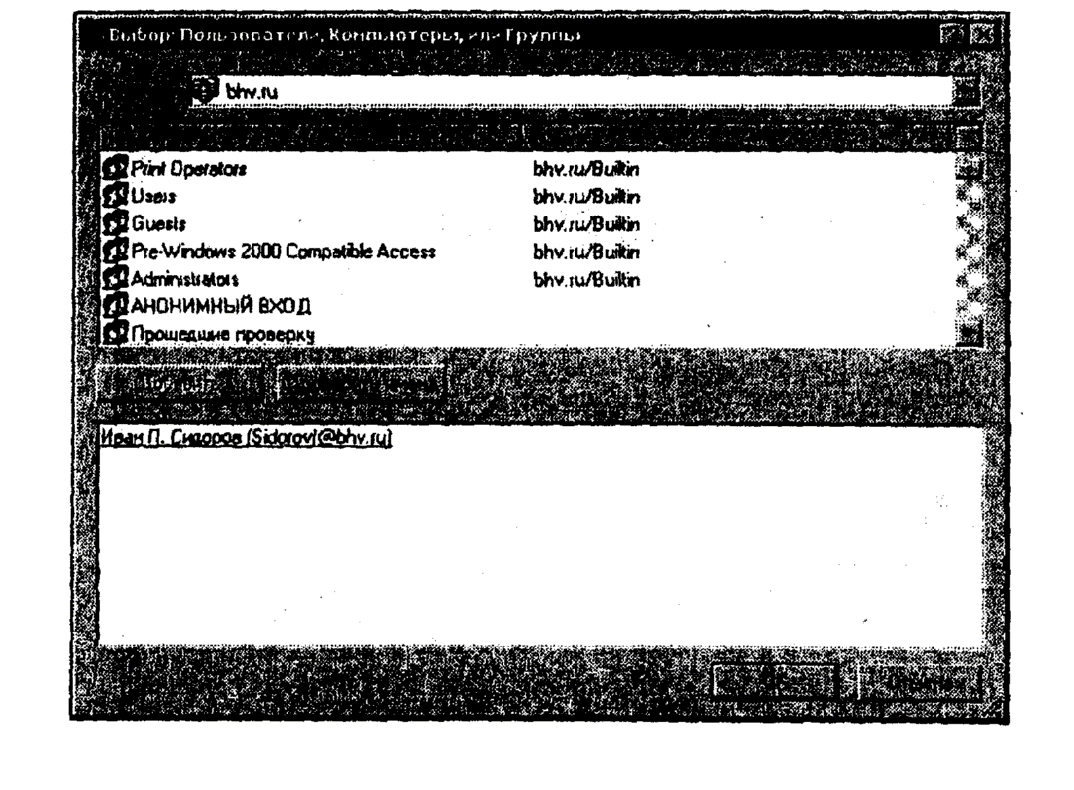
*ПРИМЕР. Назначения разрешений для файлов*

Для назначения пользователю или группе разрешения на доступ к определенному файлу:

1. Укажите файл мышью и нажмите правую кнопку. Выберите команду **Свойства** (Properties) контекстного меню. В появившемся окне свойств файла перейдите на вкладку **Безопасность**



1. В группе **Имя** (Name) показан список пользователей и групп, которым уже предоставлены разрешения для данного файла. для того чтобы добавить или удалить пользователей или группы, нажмите кнопку **Добавить** (Аdd) или **Удалить** (Remove). При добавлении пользователей появится окно диалога **Выбор: Пользователи, Компьютеры или Группы** (Select Users, Computers, or Groups).
2. Выделите нужный объект в группе **Имя** и нажмите кнопки **Добавить** (Аdd) и ОК, чтобы вернуться на вкладку **Безопасность**.
3. **В** группе **Разрешения** (Permissions) можно назначить или запретить *стандартные разрешения* для файлов — **Полный доступ** (Full Control), **Изменить** (Modify), **Чтение и выполнение** (Read *s* Execute), **Чтение** (Read) и **Запись** (Write). Для получения разрешения или отказа в разрешении служат флажки **Разрешить** (Allow) и **Запретить** (Deny). На вкладке также присутствуют кнопка **Дополнительно** (Advanced) и флажок **Переносить наследуемые от родительского объекта разрешения на этот объект** (Allow inheritable permission from parent to propagate to this object).



Каждое из перечисленных выше стандартных разрешений состоит из набора более специальных (особых) разрешений, за- дающих возможность выполнения того или иного конкретного действия с файлами или каталогами.

**2. Шифрующая файловая система (EFS)**

**Encrypting File System**

Шифрующая файловая система — это тесно интегрированная с NTFS служба, располагающаяся в ядре Windows 2000. Ее назначение: защита данных, хранящихся на диске, от несанкционированного доступа путем их шифрования. Появление этой службы не случайно и ожидалось давно. дело в том, что существующие на сегодняшний день файловые системы не обеспечивают необходимую защиту данных от несанкционированного доступа.

Хотя и NTFS обеспечивает разграничение доступа и защиту данных от несанкционированного доступа, но как быть в том случае, когда доступ к разделу NTFS осуществляется не с помощью средств операционной системы Windows NТ, а напрямую, на физическом уровне? Ведь это сравнительно легко реализовать, например, загрузившись с дискеты и запустив специальную программу: например, весьма распространенную Конечно, можно предусмотреть такую возможность, и задать пароль на запуск системы, однако практика показывает, что такая защита малоэффективна, особенно в том случае, когда за одним компьютером работают сразу несколько пользователей. А если злоумышленник может извлечь жесткий диск из компьютера, то здесь уже не помогут никакие пароли. Подключив диск к другому компьютеру, его содержимое можно будет прочитать без особых проблем. Таким образом, злоумышленник свободно может овладеть конфиденциальной информацией, которая хранится на жестком диске. Единственный способ защиты от физического чтения данных — это шифрование файлов. Простейший случай такого шифрования — архивирование файла с паролем. Однако здесь есть ряд серьезных недостатков. Во-первых, пользователю требуется каждый раз вручную шифровать и дешифровать (то есть, в нашем случае архивировать и разархивировать) данные перед началом и после окончания работы, что уже само по себе уменьшает защищенность данных. Пользователь может забыть зашифровать (заархивировать) файл после окончания работы или (еще более банально) просто оставить на диске копию файла. Во-вторых, пароли, придуманные пользователем, как правило, легко угадываются. В любом случае, существует достаточное количество утилит, позволяющих распаковывать архивы, защищенные паролем. Как правило, такие утилиты осуществляют подбор пароля путем перебора слов, записанных в словаре. Система EFS была разработана с целью преодоления этих недостатков.

**2.1. Технология шифрования**

ЕР$ использует архитектуру Windows CryptoAPI. В ее осноне лежит технология шифрования с открытым ключом, для шифрования каждого файла случайным образом генерируется ключ шифрования файла. При этом для шифрования файла может применяться любой симметричный алгоритм шифрования. В настоящее же время в EFS используется один алгоритм — это DESX, являющийся специальной модификацией широко распространенного стандарта DES. Ключи шифрования EFS хранятся в резидентном пуле памяти (сама EFS расположена в ядре Windows 2000), что исключает несанкционированный доступ к ним через файл подкачки.

По умолчанию EFS сконфигурирована таким образом, что пользователь может сразу начать использовать шифрование файлов. Операция шифрования и обратная поддерживаются для файлов и каталогов. В том случае, если шифруется каталог, автоматически шифруются все файлы и подкаталоги этого каталога. Необходимо отметить, что если зашифрованный файл перемещается или переименовывается из зашифрованного каталога в незашифрованный, то он все равно остается зашифрованным. Операции шифрования/дешифрования можно выполнить двумя различными способами — используя Windows Explorer или консольную утилиту Cipher. для того чтобы зашифровать каталог из Windows Explorer, пользователю нужно просто выбрать один или несколько каталогов и установить флажок шифрования в окне расширенных свойств каталога. Все создаваемые позже файлы и подкаталоги в этом каталоге будут также зашифрованы. Таким образом, зашифровать файл можно, просто скопировав (или перенеся) его в «зашифрованный» каталог. Зашифрованные файлы хранятся на диске в зашифрованном виде. При чтении файла данные автоматически расшифровываются, а при записи автоматически шифруются. Пользователь может работать с зашифрованными файлами так же, как и с обычными файлами, то есть открывать и редактировать в текстовом редакторе Microsoft Word документы, редактировать рисунки в Adobe Photoshop или графическом редакторе Paint, и так далее.

Необходимо отметить, что ни в коем случае нельзя шифровать файлы, которые используются при запуске системы в это время личный ключ пользователя, при помощи которого производится дешифровка, еще недоступен. Это может привести к невозможности запуска системы! B EFS предусмотрена простая защита от таких ситуаций: файлы с атрибутом «системный» не шифруются. Однако будьте внимательны: это может создать «дыру» в системе безопасности! Проверяйте, не установлен ли атрибут файла <системный» для того, чтобы убедиться, что файл действительно будет зашифрован.

Важно также помнить о том, что зашифрованные файлы не могут быть сжаты средствами Windows 2000 и наоборот. Иными словами, если каталог сжат, его содержимое не может быть зашифровано, а если содержимое каталога зашифровано, то он не может быть сжат.

В том случае, если потребуется дешифровка данных, необходимо просто снять флажки шифрования у выбранных каталогов в Windows Explorer, и файлы и подкаталоги автоматически будут дешифрованы. Следует отметить, что эта операция обычно не требуется, так как EFS обеспечивает «прозрачную» работу с зашифрованными данными для пользователя.

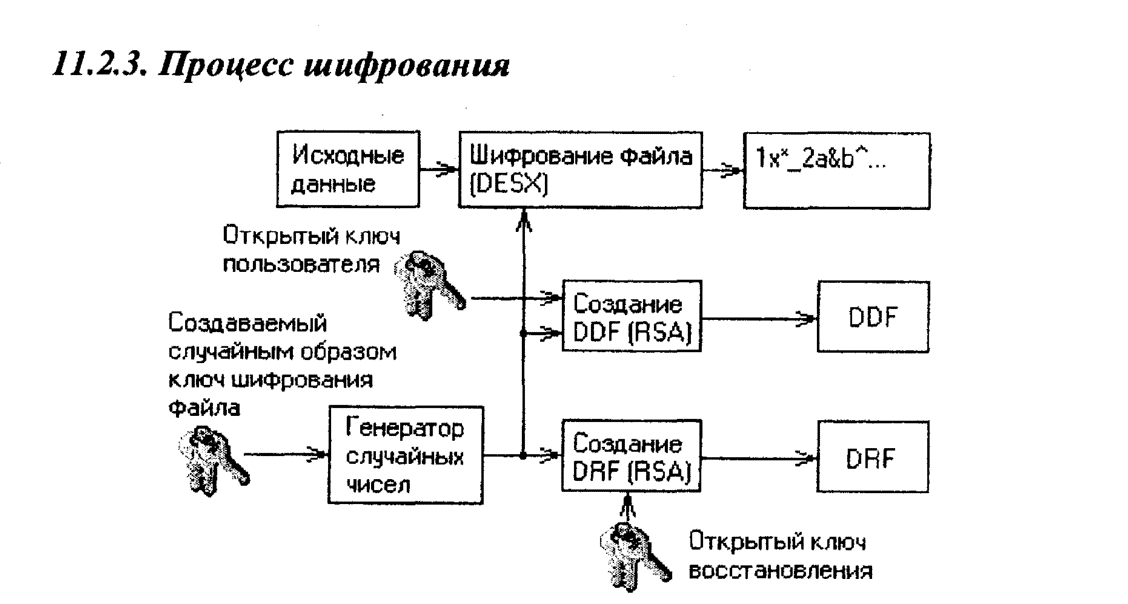
**2.2. Восстановление данных**

ЕFS обеспечивает встроенную поддержку восстановления данных на тот случай, если потребуется их расшифровать, но, по каким-либо причинам это не может быть выполнено обычным способом. По умолчанию, EFS автоматически сгенерирует ключ восстановления, установит сертификат доступа в учетной записи администратора и сохранит его при первом входе в систему. Таким образом, администратор становится так называемым агентом восстановления и сможет расшифровать любой файл в системе. Разумеется, политику восстановления данных можно изменить, и назначить в качестве агента восстановления специального человека, ответственного за безопасность данных, или даже несколько таких лиц.

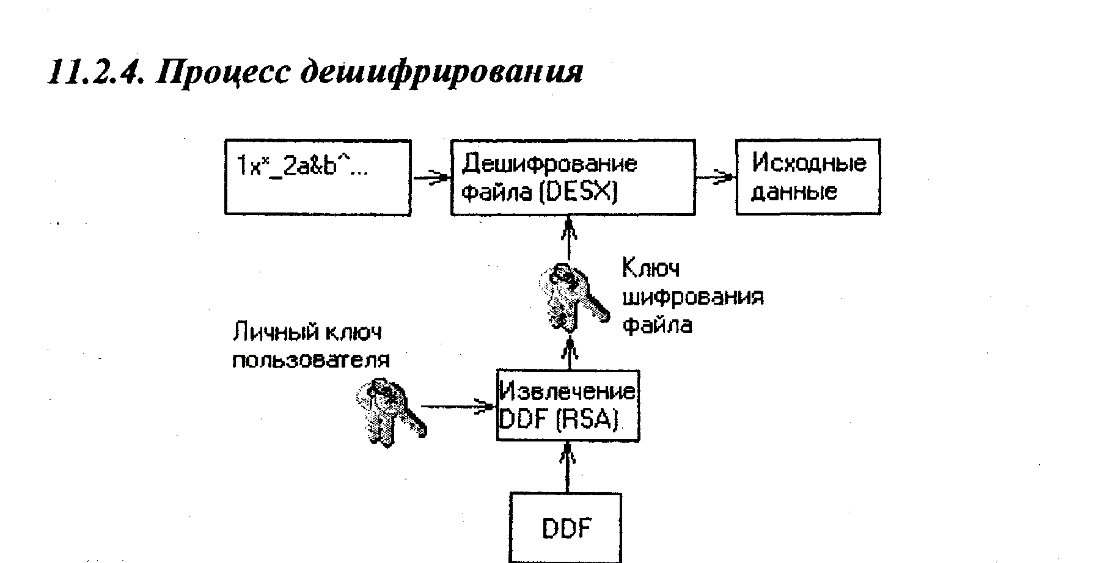
EFS осуществляет шифрование данных, используя схему с общим ключом. Данные шифруются быстрым симметричным алгоритмом при помощи ключа шифрования файла FEK (File Encryption Key). FЕК — это случайным образом сгенерированный ключ определенной длины. Длина ключа в североамериканской версии EFS 128 бит, в международной версии EFS используется уменьшенная длина ключа 40 или 56 бит.

FEK шифруется одним или несколькими общими ключами шифрования, в результате чего получается список зашифрованных ключей FEK. Список зашифрованных ключей FЕК хранится в специальном атрибуте ЕFS, который называется DDF (Data decryption field- поле дешифрования данных). Информация, при помощи которой производится шифрование данных, жестко связана с этим файлом. Общие ключи выделяются из пар пользовательских ключей сертификата Х509 с дополнительной возможностью использования «File Encryption». Личные ключи из этих пар используются при дешифровке данных и FЕК. Личная часть ключей хранится либо на смарт-картах, либо в другом надежном месте (например, в памяти, безопасность которой обеспечивается при помощи CryptoAPI). FЕК также шифруется при помощи одного или нескольких ключей восстановления (полученных из сертификатов Х509, записанных в политике восстановления зашифрованных данных для данного компьютера, с дополнительной возможностью «File recovery»).

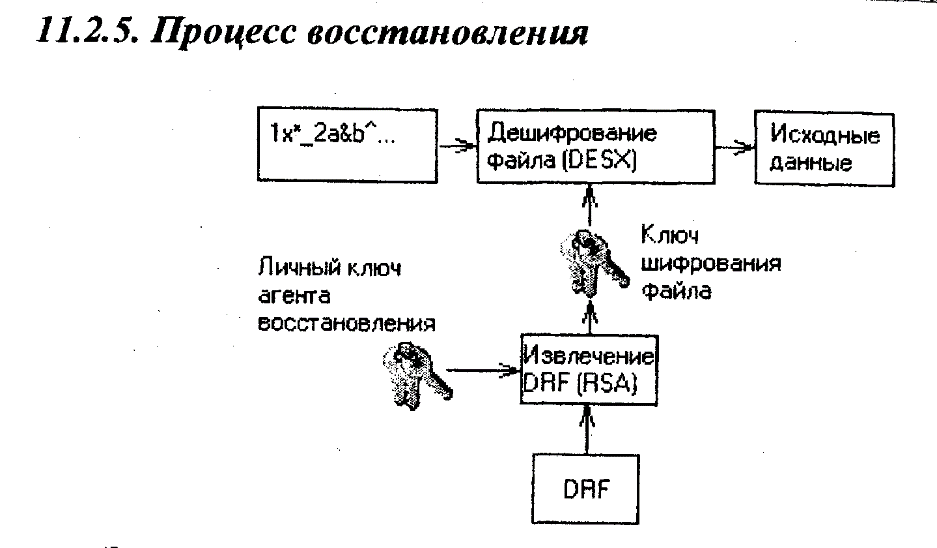
Как и в предыдущем случае, общая часть ключа используется для шифрования списка FЕК. Список зашифрованных ключей FЕК также хранится вместе с файлом в специальной области ЕFS, которая называется DRF (data recovery field — поле восстановления данных). Для шифрования списка FЕК в DRF используется только общая часть каждой пары ключей. Для нормального осуществления файловых операций необходимы только общие ключи восстановления. Агенты восстановления могут хранить свои личные ключи в безопасном месте вне системы (например, на смарт-картах). На рисунке приведены схемы процессов шифрования, дешифрования и восстановления данных.



Незашифрованный файл пользователя шифруется при помощи случайно сгенерированного ключа FЕК. Этот ключ записывается вместе с файлом, файл дешифруется при помощи общего ключа пользователя (записанного в DDF), а также при помощи общего ключа агента восстановления (записанного в DRF).

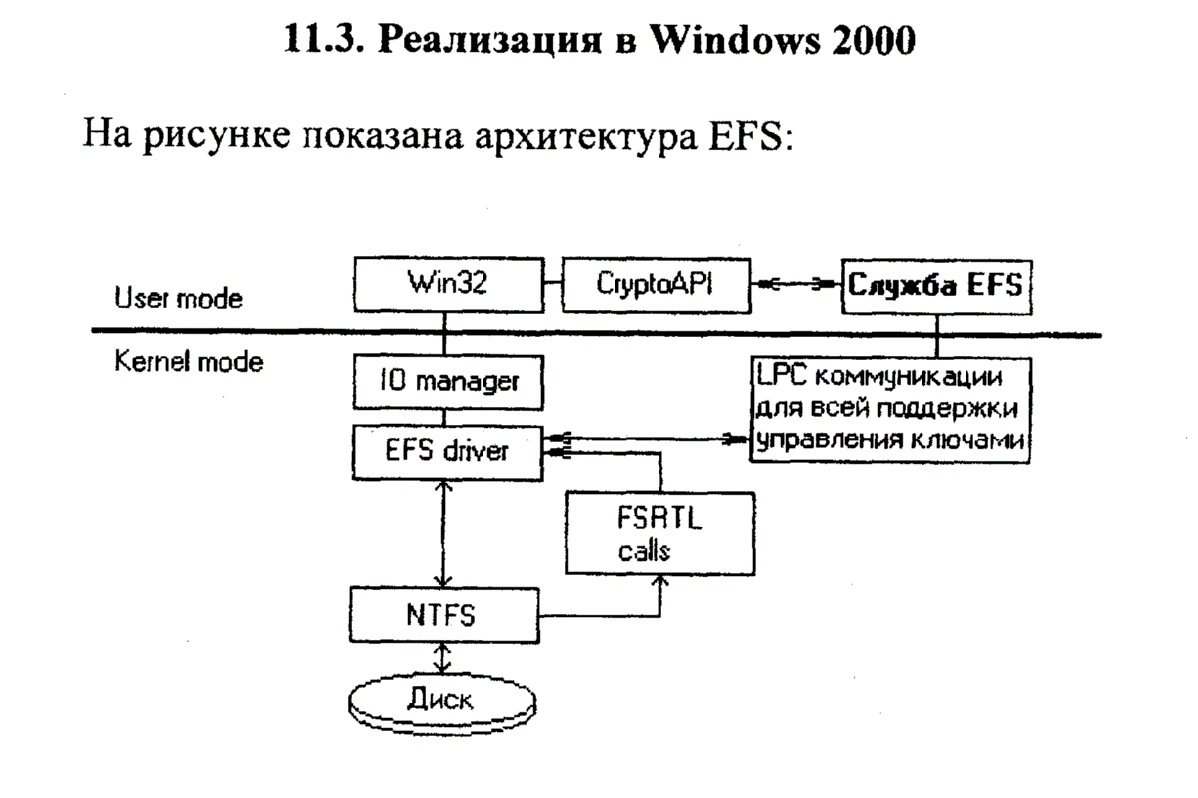


Сначала используется личный ключ пользователя для дешифрации FЕК — для этого используется зашифрованная версия FЕК, которая хранится в DDF. Расшифрованный FЕК используется для поблочного дешифрования файла. Если в большом файле блоки считываются не последовательно, то дешифруются только считываемые блоки. Файл при этом остается зашифрованным.



Этот процесс аналогичен дешифрованию с той разницей, что для дешифрования FЕК используется личный ключ агента восстановления, а зашифрованная версия FЕК берется из DRF.

**Лекция 24**



ЕFS состоит из следующих компонентов:

**1. драйвер ЕFS**

Этот компонент расположен логически на вершине NTFS. Он взаимодействует с сервисом EFS, получает ключи шифрования файлов, поля DDF, DRF и другие данные управления ключами. драйвер передает эту информацию в FSRTL (file system runtime library, библиотека времени выполнения файловой системы) для прозрачного выполнения различных файловых системных операций (например, открытие файла, чтение, запись, добавление данных в конец файла).

**2. Библиотека времени выполнения EFS (FSRTL)**

FSRTL — это модуль внутри драйвера EFS, который осуществляет внешние вызовы NTFS для выполнения различных операций файловой системы, таких, как чтение, запись, открытие зашифрованных файлов и каталогов, а также операций шифрования, дешифрования, восстановления данных при записи на диск и чтении с диска. Несмотря на то, что драйвера EFS и FSRTL реализованы в виде одного компонента, они никогда не взаимодействуют напрямую. Для обмена сообщениями между собой они используют механизм вызовов NTFS. Это гарантирует участие NTFS во всех файловых операциях. Операции, реализованные с использованием механизмов управления файлами, включают запись данных в файловые атрибуты EFS (DDF и DRF) и передачу вычисленных в EFS ключей FEK в библиотеку FSRTL так как эти ключи должны устанавливаться в контексте открытия файла. Такой контекст открытия файла позволяет затем осуществлять незаметное шифрование и дешифрование файлов при записи и считывании файлов с диска.

**З. Служба EFS**

Служба EFS является частью подсистемы безопасности. Она использует существующий порт связи LPS между LSA (Local security authority, локальные средства защиты) и работающим в kernel- mode монитором безопасности для связи с драйвером EFS. В режиме пользователя служба EFS взаимодействует с программным интерфейсом CryptoAPI, предоставляя ключи шифрования файлов и обеспечивая генерацию DDF и DRF. Кроме этого, служба EFS осуществляет поддержку интерфейса WIN32 API.

**4. WIN32** **API**

Обеспечивает интерфейс программирования для шифрования открытых файлов, дешифрования и восстановления закрытых файлов, приема и передачи закрытых файлов без их предварительной расшифровки. Реализован в виде стандартной системной библиотеки advapi32.dll.

*ПРИМЕР*

Для того чтобы зашифровать файл или каталог, выполните следующие операции:

1. Запустите Windows Explorer, нажмите правую кнопку мыши на каталоге, выберите пункт Properties (Свойства).

2. На закладке General (Общие) нажмите кнопку Advanced.

З. Отметьте галочкой пункт ‘Encrypt contents to secure data‘. Нажмите ОК, затем нажмите Аррlу (применить) в диалоге Properties. Если Вы выбрали шифрование отдельного файла, то дополнительно появится диалоговое окно следующего вида

Система предлагает зашифровать также и каталог, в котором находится выбранный файл, так как в противном случае шифрование будет автоматически отменено при первой модификации такого файла. Всегда имейте это в виду, когда шифруете отдельные файлы! На этом процесс шифрования данных можно считать законченным.

Чтобы расшифровать каталоги, просто снимите выделение в пункте ‘Encrypt contents to secure data’. При этом каталоги, а также все содержащиеся в них подкаталоги и файлы будут расшифрованы.

• Система EFS в Windows 2000 предоставляет пользователям возможность зашифровывать каталоги NTFS, используя устойчивую, основанную на общих ключах криптографическую схему, при этом все файлы в закрытых каталогах будут зашифрованы. Шифрование отдельных файлов поддерживается, но не рекомендуется из-за непредсказуемого поведения приложений.

• Система EFS также поддерживает шифрование удаленных файлов, доступ к которым осуществляется как к совместно используемым ресурсам. Если имеют место пользовательские профили для подключения, используются ключи и сертификаты удаленных профилей. В других случаях генерируются локальные профили и используются локальные ключи.

• Система EFS предоставляет возможность установить политику восстановления данных таким образом, что зашифрованные данные могут быть восстановлены при помощи EFS, если это потребуется.

• Политика восстановления данных встроена в общую политику безопасности Windows 2000. Контроль за соблюдением политики восстановления может быть делегирован уполномоченным на это лицам. для каждого подразделения организации может быть сконфигурирована своя политика восстановления данных.

• Восстановление данных в EFS — закрытая операция. В процессе восстановления расшифровываются данные, но не ключ пользователя, при помощи которого эти данные были зашифрованы.

• Работа с зашифрованными файлами в EFS не требует от пользователя каких-либо специальных действий по шифрованию и дешифрованию данных. Дешифрование и шифрование происходят незаметно для пользователя в процессе считывания и записи данных на диск.

• Система EFS поддерживает резервное копирование и восстановление зашифрованных файлов без их расшифровки. Программа NtBackup поддерживает резервное копирование зашифрованных файлов.

• Система EFS встроена в операционную систему таким образом, что утечка информации через файлы подкачки невозможна, при этом гарантируется, что все создаваемые копии будут зашифрованы.

• Предусмотрены многочисленные меры предосторожности для обеспечения безопасности восстановления данных, а также защита от утечки и потери данных в случае фатальных сбоев системы.

**11.4. Взаимодействие файловой системы защиты NTFS и защиты ресурса общего доступа (Sharing)**

Как правило, когда вы ограничиваете доступ к общему ресурсу или к файлу/папке, права доступа конкретного пользователя суммируются. Т.е., например, если некий пользователь является членом двух групп, одной из которых предоставлен доступ только для чтения, а другой — доступ с правом изменения содержимого ресурса, то этот пользователь будет обладать правами и на чтение, и на изменение ресурса. Исключением из данного правила является только случай, когда одна из групп, к которым принадлежит пользователь, вообще не имеет доступа к данному ресурсу. В этом случае любой пользователь из такой группы не будет иметь доступа к данному ресурсу, независимо от членства в каких-либо других группах. В случае, когда ограничение доступа устанавливается на файл/папку, который еще является и общим ресурсом, действовать будет наиболее строгое ограничение. Например, если в свойствах файловой системы пользователю предоставлен полный доступ к данному файлу, а свойства ресурса общего доступа, в который входит этот файл, разрешают доступ только для чтения, то пользователь будет обладать доступом только для чтения. То же самое будет верно и в обратном случае, т.е. если в свойствах файла\папки будет разрешен доступ только для чтения, а к общему ресурсу будет открыт полный доступ, пользователь все равно будет ограничен доступом только для чтения.

Следует иметь в виду, что ограничения доступа, устанавливаемые на ресурсы общего доступа, используются только в тех случаях, когда доступ к ресурсу осуществляется при помощи сетевого соединения. Если пользователь обращается к общему ресурсу локально, то ограничения доступа, установленные на этот ресурс, игнорируются.

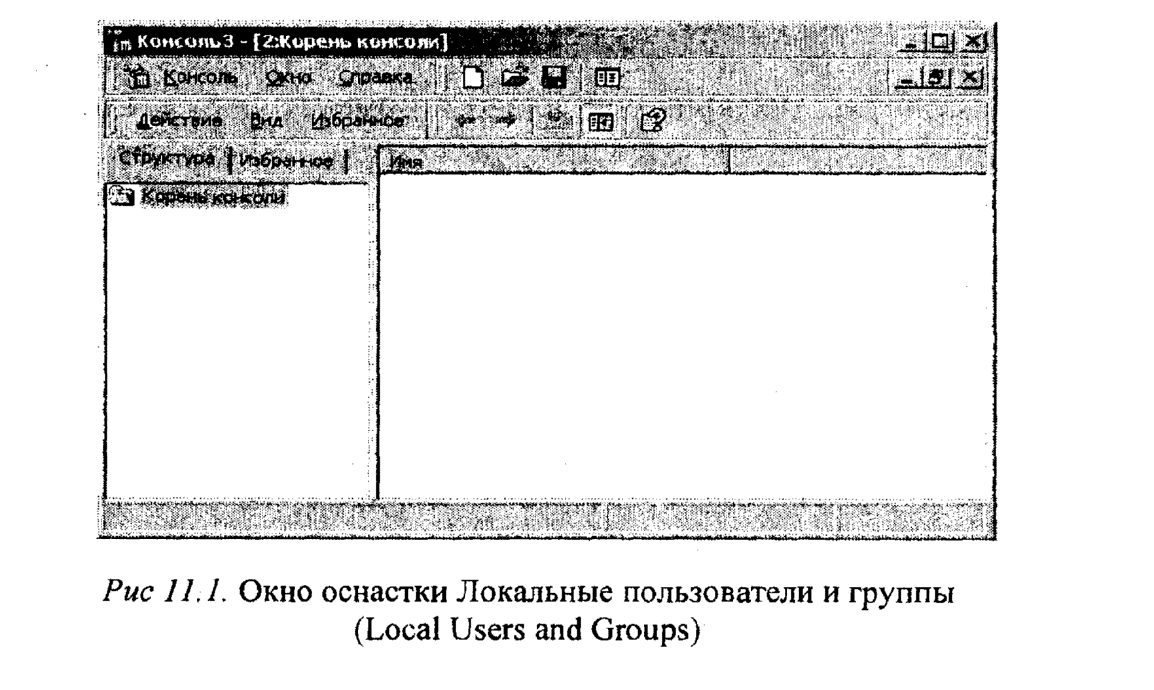
**11.5. Типовые задачи администрирования**

Создание учетных записей и групп занимает важное место в обеспечении безопасности Политика восстановления данных встроена в общую политику безопасности Windows 2000. Контроль за соблюдением политики восстановления может быть делегирован уполномоченным на это лицам. для каждого подразделения организации может быть сконфигурирована своя политика восстановления данных, 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.

1. **Оснастка *Локальные пользователи и группы***

**(Local Users and Groups)**

Оснастка **Локальные пользователи и группы** — это инструмент ММС (Консоль управления Microsoft), 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 2000, как на изолированных, так и рядовых членах домена (member server). На контроллерах домена Windows 2000 инструмент **Локальные пользователи и группы** недоступен, поскольку все управление учетными записями и группами в домене выполняется с помощью оснастки **Active Directory — пользователи и компьютеры** (ActiveDirectoryUses and Computers). Запускать оснастку **Локальные пользователи и группы** может любой пользователь. Выполнять администрирование учетных записей могут только администраторы и члены группы Опытные пользователи (Power Usesr). Окно изолированной оснастки **Локальные пользователи и группы** выглядит аналогично показанному на рис. 11.1.



• **Папка Пользователи (Users)**

Сразу после установки системы Windows 2000 (рабочей станции или сервера, являющегося членом домена) папка **Пользователи** содержит две встроенные учетные записи — Администратор (Administrator) и Гость (Guest). Они создаются автоматически при установке Windows 2000. Ниже даны описания свойств обеих встроенных учетных записей:

**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 (Administrator), 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 (Guests). Ей можно предоставить права доступа к ресурсам системы точно так же, как любой другой учетной записи.

**• Папка Группы (Groups)**

После установки системы Windows 2000 (рабочей станции или сервера, являющегося членом домена) папка **Группы** (Groups) содержит шесть встроенных групп. Они создаются автоматически при установке Windows 2000. Ниже описаны свойства всех встроенных групп:

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

**Операторы архива** (Backup Operators) — 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** (**Guests**) — 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

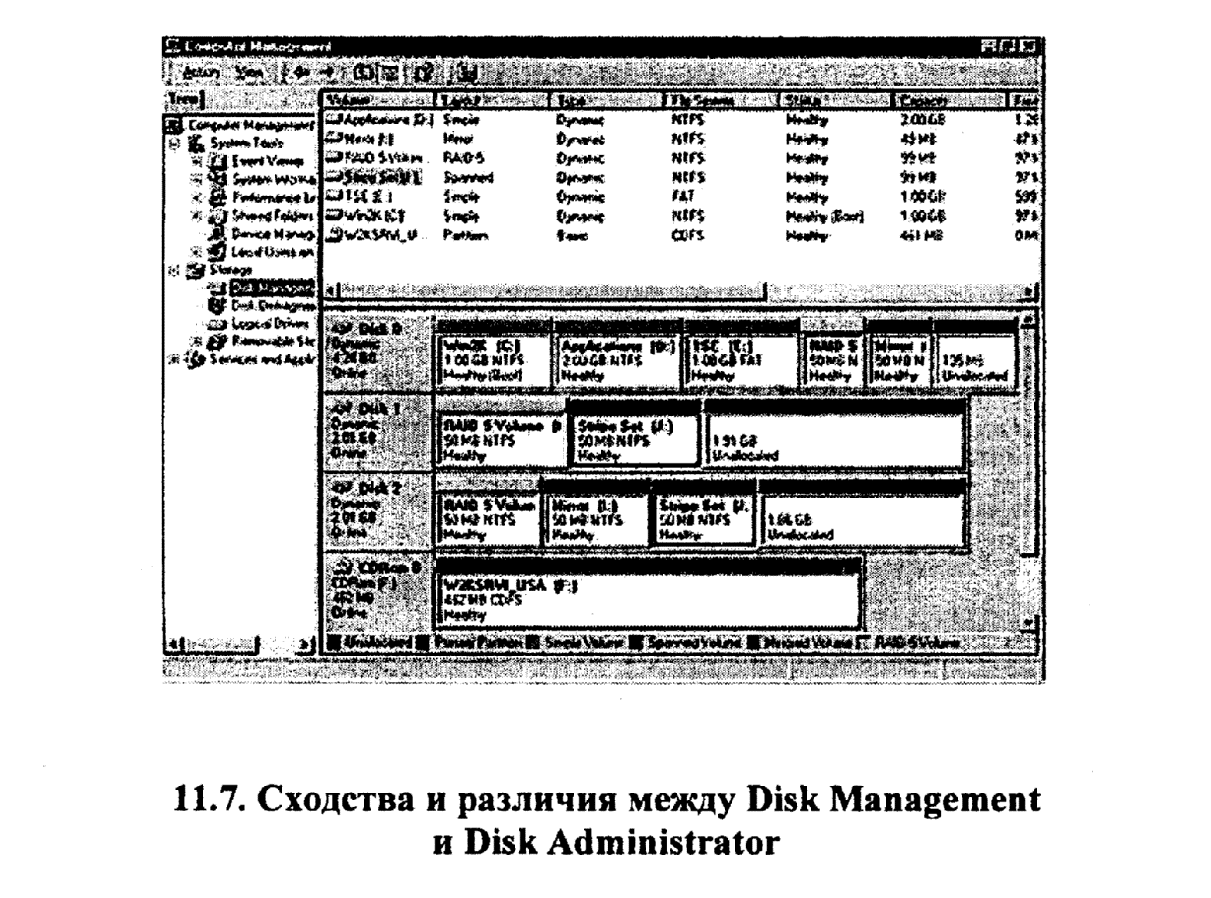
**Опытные пользователи** **(Power Users)** — члены этой группы могут создавать учетные записи пользователей, но они имеют право модифицировать настройки безопасности только для созданных ими учетных записей. Кроме того, они могут создавать локальные группы и модифицировать состав членов созданных ими 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** (Replicator) — 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** (Users) — 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.

**11.6. Администрирование дисков в Windows** **2000**

Опытные пользователи Windows NТ успели привыкнуть к утилите Disk Administrator, с помощью которой можно было создавать, удалять и менять параметры томов на физических дисках. В Windows 2000 ее заменил модуль административной консоли Disk Management. Для работы с ним следует открыть папку Administrative Tools и выбрать Computer Management- Storage- Disk Management (см. ниже Рис).



Disk Management и Disk Administrator во многом похожи. Новое средство по-прежнему поддерживает RAID уровней 0, 1 и 5, дисковые массивы с чередованием и с четностью (состоят из трех и более дисков, на один из которых записывается информация, позволяющая восстановить данные при выходе из строя любого диска), дисковые массивы с чередованием без четности (данные размещаются с чередованием на нескольких физических дисках), а также дублирование (для каждого тома создается зеркальная копия на другом физическом диске). Кроме того, поддерживаются многодисковые тома, объединяющие свободное пространство на нескольких дисках в один логический том. Хотя интерфейс может показаться знакомым, следует иметь в виду некоторые особенности Disk Management. Необходимость полной реализации новых свойств несколько изменила и расширила терминологию. Кроме того, преодолено прежнее ограничение в 26 логических дисков. Новый интерфейс упрощает создание и изменение томов, причем после этих операций перезагрузка больше не требуется. Теперь вся информация обо всех дисках (нормально ли они работают, их объем, тип файловой системы, а также количество свободного места и процент использования) собрана в одном месте, причем тут же отображаются сведения о структуре тома. Это позволяет отказаться от постоянного переключения режимов просмотра, что приходилось делать в Disk Administrator. Наконец, теперь можно восстановить данные без перезагрузки, поскольку при замене вышедшего из строя диска в томах с дублированием и RAID 5-го уровня эти тома становятся работоспособны сразу же после автоматической регенерации данных.