Московский Государственный Технический Университет имени Н. Э. Баумана

Пугачев Е.К.

Исследование среды управления и методов защиты.

Методические указания по выполнению лабораторной работы
по дисциплине "Операционные системы".

Москва 2013

Часть 1. Исследование среды управления.

В Windows 2000 был кардинально изменен интерфейс управления системами Windows NT. В соответствии с новой концепцией Microsoft из системы Windows NT были удалены все автономные и несовместимые друг с другом административные утилиты и разработана единая среда управления, получившая название *консоль управления Microsoft* (Microsoft Management Console, MMC). Существуют инструменты оснастки, которые позволяют управлять пользователями и группами.

Цель работы - исследование структуры и основных функций (оснасток) консоли управления MMC.

Продолжительность работы - 4 часа.

Краткие теоретические сведения

MMC - это общая консоль управления, которая разработана для запуска всех программных модулей администрирования, конфигурирования или мониторинга локальных компьютеров и сети в целом. Такие законченные модули называются оснастками (snap-in). Консоль управления сама по себе не выполняет никаких функций администрирования, но служит в качестве хост среды (рабочей среды) для запуска оснасток. Оснастки создаются как компанией Microsoft, так и независимыми поставщиками программного обеспечения (independent software vendor, ISV). Среда MMC обеспечивает тесную интеграцию всех оснасток независимо от их производителя.

Появление MMC связано с желанием создать единую среду управления для администрирования операционных систем Windows. Оснастки представляют собой управляющие компоненты, которые объединены в среде MMC. Из нескольких оснасток можно создать индивидуальный управляющий инструмент.

Консоль MMC включает в себя интерфейсы прикладного программирования (API), оболочку пользовательского интерфейса (консоли) и набор инструкций.

Microsoft Management Console позволяет создавать более совершенные административные инструменты, которые могут предоставлять различные уровни функциональных возможностей. Эти инструменты можно легко интегрировать в операционную систему, а также изменять и настраивать по своему усмотрению. В данном случае инструмент представляет собой не просто одиночное приложение. Инструмент может состоять из одной или нескольких оснасток, и каждая оснастка в свою очередь может содержать дополнительные оснастки-расширения. Такая модульная структура позволяет системному администратору существенно снизить стоимость управления системой благодаря созданию индивидуальных инструментов на основе выбранных

оснасток, которые предоставляют только необходимые возможности и средства просмотра. Администратор может затем сохранять каждый индивидуальный инструмент в отдельном файле (файле консоли MMC с расширением .msc) и отправлять его другим пользователям или администраторам, которым делегированы права на выполнение данных административных задач.

MMC и модель администрирования Windows 2000 представляют следующий шаг в развитии технологии администрирования. Консоль управления имеет ряд преимуществ, которые заключаются в упрощении интерфейса, предоставлении больших возможностей по настройке разработанных решений для определенных административных проблем и в обеспечении различных уровней функциональности. В большинстве случаев достаточно сложно разработать инструмент, который будет являться неотъемлемой частью операционной системы. С помощью MMC эта задача существенно упрощается.

В операционные системы Windows 2000 и следующие версии продуктов семейства оснастки MMC включены в качестве стандартных административных программ.

Microsoft Management Console представляет собой приложение с многооконным интерфейсом, которое активно использует технологии Internet. Компания Microsoft и независимые поставщики программного обеспечения могут разрабатывать оснастки MMC для выполнения задач управления локальным компьютером и сетью в целом. MMC не подменяет собой имеющиеся инструменты управления предприятиями, такие как HP OpenView или IBM Tivoli Management Environment. Консоль управления расширяет возможности данных инструментов, предоставляя им возможность взаимодействия друг с другом или объединяя эти инструменты в оснастки, доступ к которым может осуществляться из MMC.

Например, приложение управления предприятием может обнаружить событие и отправить извещение в оснастку (рис. 1а). Системный администратор затем обнаружит событие в сеансе MMC и предпримет необходимые меры. Интерфейсы программирования MMC позволяют интегрировать оснастки с консолью (рис. 1б).

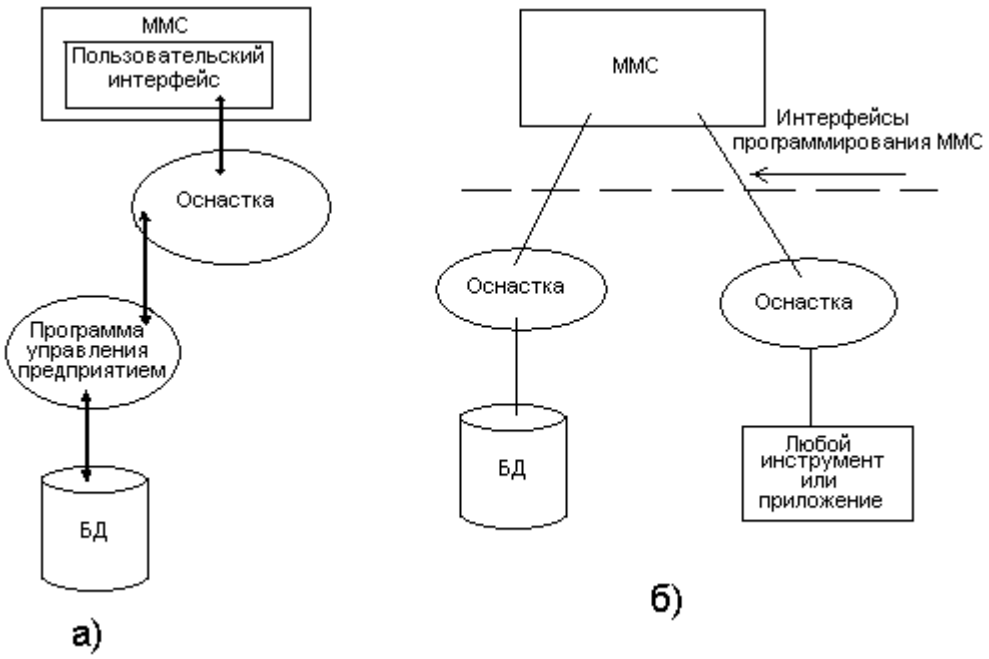

Рис. 1.

Данные интерфейсы предоставляют только расширения пользовательского интерфейса, поскольку каждая оснастка самостоятельно определяет механизм выполнения своих задач. Интерфейсы ММС позволяют оснасткам совместно использовать общую хост-среду и обеспечивают интеграцию между приложениями. Консоль ММС не выполняет никаких функций управления.

Компания Microsoft и независимые поставщики программного обеспечения могут разрабатывать инструменты управления для запуска в среде ММС и приложения, которыми будут управлять инструменты ММС.

Инструменты, которые не предназначены для работы в среде ММС, могут быть интегрированы с ММС посредством оснасток или запушены независимо. Системный администратор может одновременно запускать не ММС инструменты управления и экземпляры ММС на одном компьютере.

Моно выделить следующие преимущества:

- Возможность индивидуальной настройки и передача полномочий. Помимо обеспечения интеграции и обшей среды для административных инструментов, консоль ММС предоставляет возможность полностью индивидуальной настройки консоли, так что администраторы могут создавать такие консоли управления, которые будут включать только необходимые им инструменты. Такая настройка позволяет ориентировать администрирование на выполнение конкретных задач, администратор может выделить только необходимые объекты и элементы. Настройка консоли также позволяет администраторам

передавать определенную часть полномочий менее опытным сотрудникам. С помощью ММС можно создать консоль, которая будет содержать объекты, необходимые для выполнения только определенных функций.

- Интеграция и унификация. ММС обеспечивает общую среду, в которой могут запускаться оснастки, и администраторы могут управлять различными сетевыми продуктами, используя единый интегрированный интерфейс. Благодаря этому все оснастки имеют одинаковый интерфейс, что упрощает изучение работы с различными инструментами.
- Гибкость в выборе инструментов и продуктов. В среде ММС можно использовать различные инструменты и оснастки. Для использования в среде ММС оснастка должна поддерживать распределенную модель объектной компоновки (Distributed Component Object Model, DCOM) или модель объектной компоновки (Component Object Model, COM). Это позволяет выбирать наиболее оптимальный продукт среди оснасток, причем гарантируется его полная совместимость со средой ММС.

Пользовательский интерфейс ММС.

Консоль управления ММС имеет пользовательский интерфейс, позволяющий открывать множество документов (multiple *document* interface, MDI). Родительское окно ММС имеет главное меню и панель инструментов. Главное меню предоставляет возможность выполнения функций 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 обзора (scope pane) и панель результатов (result pane). Панель управления содержит панель меню и панель инструментов. Панель обзора отображает пространство имен инструментов в виде дерева консоли. Это дерево содержит все видимые узлы, каждый из которых является управляемым объектом, задачей или средством просмотра. Панель результатов в дочернем окне отображает результат выбора узла в панели обзора. В некоторых случаях в панели результатов отображается список элементов выбранного узла или элемент управления в ActiveX или Web-страница. Возможности ММС также позволяют отображать окно в упрощенном виде, доступном для менее опытных администраторов. В наиболее простой форме окно может содержать набор значков, которые обеспечивают доступ к определенным задачам.

Архитектура ММС.

На рис. 2 представлена архитектура ММС. Диспетчер оснасток (Snap- In Manager) дает системному администратору или разработчику оснасток возможность добавлять, удалять или изменять оснастки. Кроме того, Диспетчер оснасток позволяет системному администратору определить, является ли некоторая оснастка автономной или зависит от других оснасток. Диспетчер оснасток сохраняет произведенные установки в *инструменте* или *документе* (файл с расширением .msc). Пользователь определенного инструмента взаимодействует с элементами, которые находятся в левой части рисунка (файл .msc и элементы пользовательского интерфейса). Разработчики и администраторы работают с элементами, показанными в правой части рис.2 (Диспетчер оснасток и оснастки Event Viewer и Router Manager).

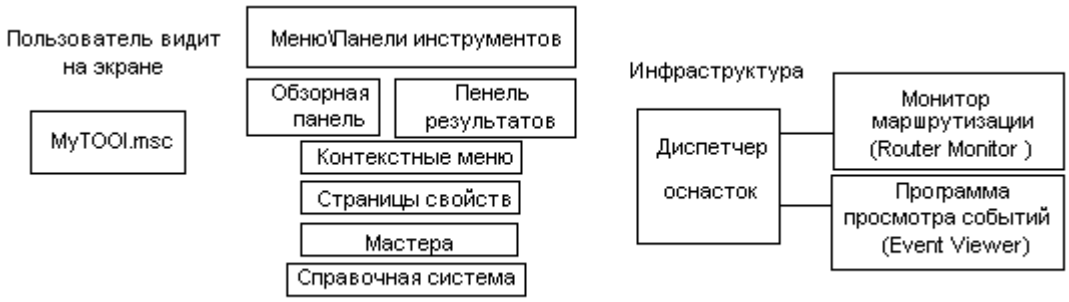

Рис. 2. Архитектура ММС.

Данные оснастки объединены для создания пространства имен - набора узлов, которые отображаются в виде дерева в панели обзора. Пространство имен является главным деревом, которое показывает возможности инструмента. Пространство имен может включать все управляемые объекты сети компьютеры, пользователей, группы и т. д. Пространство имен содержит объекты, средства просмотра и задачи. Дочерние окна ММС представляют собой средства просмотра главного пространства имен.

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 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. В случае отсутствия необходимых оснасток на компьютере этого администратора MMC с помощью службы Active Directory Service автоматически загрузит необходимые оснастки при запуске файла .msc.

Благодаря возможности индивидуальной настройки MMC 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.

Оснастки могут работать в следующих внутренних режимах:

- *Автономная оснастка* (standalone snap-in) обеспечивает выполнение своих функций даже при отсутствии других оснасток, например, Computer Management.
- *Оснастка-расширение* (extension snap-in) может работать только после активизации родительской оснастки. Функция оснастки-расширения заключается в увеличении числа типов узлов, поддерживаемых родительской оснасткой. Оснастка-расширение является подчиненным элементом узлов определенных типов и при каждом запуске узлов данных типов консоль автоматически запускает все связанные с ней расширения. В качестве примера можно привести оснастку Device Manager.
- *Оснастки-расширения* могут предоставлять различные функциональные возможности. Например, такие оснастки могут расширять пространство имен консоли, увеличивать число пунктов в меню или добавлять определенные мастера.
- *Комбинированные оснастки* поддерживают оба режима работы, располагая возможностями автономной работы и предоставляя некоторые дополнительные функциональные возможности другим оснасткам, например, Event Viewer.

Создание новой консоли.

Для того чтобы получить представление о гибкости MMC, полезно рассмотреть процесс создания файла консоли - инструмента (документа) MMC с самого начала. Для

примера опишем процедуру создания новой консоли и добавления к ней оснасток Computer Management и Monitoring Control..

1. В меню **Start** (Пуск) выберите пункт **Run** (Выполнить), введите **mmc** и нажмите кнопку **OK.** Откроется окно Microsoft Management Console с пустой *консолью* (или *административным инструментом).* По умолчанию консоль ММС открывается в *авторском режиме,* в котором можно создавать новые консоли и изменять созданные ранее административные инструменты. Пустая консоль не имеет никаких функциональных возможностей до тех пор, пока в нее не добавить оснастки. Команды меню ММС на панели меню в верхней части окна применимы ко всей консоли.

2. В меню **Console** (Консоль) выберите пункт **Add/Remove Snap-In** (Добавить/Удалить оснастку). Откроется окно **Add/Remove Snap-In.** В этом окне перечислены автономные оснастки (вкладка Standalone) и оснастки расширения (вкладка Extensions), которые будут добавлены в консоль (или уже включены в нее). Оснастки можно добавлять к корневому узлу дерева консоли управления или к уже имеющимся автономным оснасткам (другим узлам дерева); это указывается в списке Snap-In added to (Подключено к оснастке). В нашем случае оставим значение по умолчанию - Console Root.

3. Нажмите кнопку Add (Добавить). На экране появится окно Add Standalone Snap-in (Добавить автономную оснастку) со списком автономных оснасток, имеющихся в системе.

Следует различать *имена оснасток* (т.е. их названия, которые были даны разработчиками и которые зафиксированы в поставляемых пакетах оснасток и *названия* элементов меню, инструментов ММС и узлов в дереве оснасток консоли ММС. Например, оснастка Active Directory Manager вызывается командой Start│Administrative Tools│Directory Management. Еще пример: автономная оснастка System Service Management входит в оснастку Computer Management под именем Services. Этот нюанс может поначалу сбивать с толку новых пользователей системы.

4. Выполните двойной щелчок на оснастке Computer Management. Появится окно с конфигурационными опциями для данной оснастки.

5. Оставьте переключатель в положении Local computer (Локальный компьютер). Затем нажмите кнопку Finish (Готово).

6. В окне оснасток выберите пункт Monitoring Control и нажмите кнопку Add (Добавить). Данная оснастка является элементом ActiveX и заменяет программу Performance Monitor, которая использовалась в Windows NT 4.0. Запустится мастер установки компонента ActiveX. Нажмите кнопку Next (Далее).

7. В следующем окне по умолчанию выбрано положение Only monitor controls (Только элементы управления для мониторинга) переключателя, нажмите кнопку Next. Если установить переключатель в положение AII controls (Все элементы управления), то в окне будут выведены названия всех элементов управления.

8. В последнем окне укажите название устанавливаемого компонента. Оставьте значение по умолчанию - System Monitor Control и нажмите кнопку Finish.

9. Закройте окно выбора оснасток, нажав кнопку Close (Закрыть).

10. Наконец, в окне Add/Remove Snap-In (где отображен список подключаемых оснасток) нажмите кнопку OK. Теперь окно консоли содержит две оснастки - Console Management и System Monitor Control.

11. Для того чтобы сохранить созданный инструмент, в меню Console выберите пункт Save As и укажите имя файла и папку, в которой будет сохранен файл консоли (по умолчанию файл консоли будет сохранен в папке My Administrative Tools). Данная папка находится в профиле текущего пользователя, поэтому созданный инструмент можно запустить, открыв меню Start | Programs | My Administrative Tools.

Дополнительным преимуществом такого подхода является то, что при наличии у пользователя блуждающих профилей, все созданные им инструменты будут перемещаться вместе с ним.

Оснастки, имеющиеся в Windows 2000.

Ниже в таблице перечислены оснастки, устанавливаемые по умолчанию в системах Windows 2000 Professional и Windows 2000 Server. (Для оснасток, включенных в пользовательский интерфейс, указаны названия соответствующих пунктов меню, для остальных оснасток даны их собственные имена.) Оснастки, работающие только на контроллере домена под управлением Windows 2000 Server, отмечены буквой D. Оснастки, которые можно вызывать непосредственно из меню Start (т. е. оснастки, включенные в пользовательский интерфейс при инсталляции системы), отмечены звездочкой.

1. *Certificate Manager - служит для управления сертификатами.
2. *Computer Management - предоставляет функции администрирования системы; управляет подключением к серверу и открытыми файлами; содержит в своем составе ряд автономных оснасток и оснасток расширений.
3. Device Manager - содержит список всех устройств, подключенных к компьютеру, и позволяет их конфигурировать.
4. Directory Management (D)- служит для управления объектами Active Directory; служит для управления учетными записями пользователей и групп; позволяет передавать административные функции другому пользователю; позволяет устанавливать разрешения, аудит и владельцев для объектов каталога; служит для управления политикой безопасности домена; служит для управления политикой безопасности для всех компьютеров домена; служит для публикации общих ресурсов в качестве томов в Active Directory.

5. Disk Defragmenter - служит для анализа и дефрагментации дисковых томов.
6. Disk Management - служит для управления дисками и защитой данных, разбиения дисков на логические тома, форматирования, управления совместным доступом, квотами и т. д.
7. *Domain Tree Management (D) – служит для управления доверительными отношениями доменов.
8. *Event Viewer - служит для просмотра и управления системным журналом, журналом безопасности и приложений.
9. Fax Services Manager - служит для управления службой и устройствами факсимильной связи.
10. File Service Management - служит для установки разрешений, аудита и владельцев общих ресурсов; служит для управления подключениями, общими томами, папками и файлами.
11. Folder – служит для добавления новой папки в дерево.
12. General Control – служит для подключения к консоли элементов ActiveX.
13. Group Policy Editor - служит для назначения сценариев регистрации; служит для назначения групповых политик для компьютера и пользователей не которого компьютера сети.
14. *Index Server Management – служит для индексирования различных документов с целью ускорения их поиска при работе с IIS(Internet Information Server).
15. IP Security Policy Management – служит для настройки политик безопасности для межсетевых коммуникаций.
16. Link to Web Address – служит для подключения Web-страницы (html,asp,stml).
17. Local User Manager – служит для управления локальными учетными записями пользователей и групп.
18. Microsoft System Information – служит для просмотра информации о компьютере (конфигурация, запущенные приложения и сервисы и т. п.).
19. Monitoring Control – служит для мониторинга производительности системы.
20. Remote Storage Service – служит для управления удаленными хранилищами.
21. *Removable Storage Management – служит для управления сменными носителями информации.
22. Security Configuration Editor – служит для управления безопасностью системы.
23. System Monitor Log Manager – служит для создания журналов на основе данных производительности системы.

24. System Service Management - служит для управления системными сервисами и службами.

Управление пользователями и группами.

Создание локальных учетных записей пользователей и групп.

Создание учетных записей и групп занимает важное место в обеспечении безопасности Windows 2000, поскольку назначая им права доступа, администратор получает возможность ограничить пользователей в доступе к конфиденциальной информации компьютерной сети, разрешить или запретить им выполнить в сети определенное действие, 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 Local User Manager.

Local User Manager - это инструмент, оснастка 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 2000 (как отдельно стоящих, так и являющихся *членами* домена, member server). На контроллерах домена Windows 2000 Local User Manager недоступен, поскольку все управление учетными записями и группами домена выполняется с помощью оснастки Active Directory Manager. Запускать Local User Manager может любой пользователь. Выполнять администрирование учетных записей могут только администраторы и члены группы Power Users.

Папка Users.

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

- **Administrator** - 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 **Administrators,** ее можно только переименовать.
- **Guest** - эта учетная запись применяется для регистрации в компьютере без использования специально созданной учетной записи. Учетная запись Guest не требует ввода пароля и по умолчанию блокирована (обычно пользователь, чья учетная запись блокирована, но не уничтожена, при регистрации получает предупреждение и входить в систему не может). Она является

членом группы Guests. Ей можно предоставить права доступа к ресурсам системы точно так же, как любой другой учетной записи.

Папка Groups.

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

- Administrators - ее члены обладают полным доступом ко всем ресурсам системы. Это единственная встроенная группа, автоматически предоставляющая своим членам весь набор встроенных прав.
- Backup Operators - члены этой группы могут архивировать и восстанавливать файлы в системе независимо от того, какими правами эти файлы защищены. Кроме того, операторы архивации могут входить в систему и завершать ее работу, но они не имеют права изменять настройки безопасности.
- Power Users - члены этой группы могут создавать учетные записи пользователей, но они имеют право модифицировать настройки безопасности только созданных ими пользователей. Кроме того, они могут создавать локальные группы и модифицировать состав членов созданных ими групп. То же самое они могут делать с группами Users, Guests и Power Users. Члены группы Power Users не могут модифицировать членство в группах Administrator и Backup Operators. Они не могут быть владельцами файлов, архивировать или восстанавливать каталоги, загружать и выгружать драйверы устройств и модифицировать настройки безопасности и журнал событий.
- 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.
- Guests - эта группа позволяет выполнить регистрацию пользователя с помощью учетной записи Guest и получить ограниченные права на доступ к ресурсам системы. Члены этой группы могут завершать работу системы.
- Replicator - членом группы Replicator должна быть только учетная запись, с помощью которой можно зарегистрироваться в службе репликации контроллера домена. Ее членами не следует делать рабочие учетные записи.

Управление учетными записями.

Создание учетной записи.

Для создания учетной записи необходимо выполнить следующие действия:

1. В оснастке Local User Manager установите указатель мыши на папку Users и нажмите правую кнопку. В появившемся контекстном меню выберите команду Create User (Создать пользователя).

2. Появится окно диалога Create User. В поле User name (Имя пользователя) введите имя создаваемого пользователя. В поле Full name (Полное имя) введите полное имя создаваемого пользователя. В поле Description (Описание) введите описание создаваемого пользователя или его учетной записи. В поле Password (Пароль) введите пароль пользователя и в поле Confirm (Подтверждение) подтвердите его правильность вторичным вводом. Длина пароля не может превышать 14 символов.

3. Установите или снимите флажки User must change password ayt next logon (Пользователь должен изменить пароль при следующей регистрации), User cannot change password (Пользователь не может изменять пароль), Password never expires (Время действия пароля не истекает) и Account disabled (Учетная запись блокирована).

4. Если вы хотите создать еще одного пользователя, нажмите кнопку Create (Создать) и повторите шаги с 1 по 3. Если вы хотите завершить работу, нажмите кнопку Create и затем Close (Закрыть).

Имя пользователя должно быть уникальным в пределах компьютера. Оно может содержать до 20 символов верхнего и нижнего регистра. Ниже приведены символы, применение которых в имени пользователя недопустимо: " / \ [ ] : ; | = , + * ? < >

Имя пользователя не может состоять целиком из точек и пробелов.

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 Action (Действие) на панели меню оснастки Local User Manager.

Поскольку переименованная учетная запись сохраняет SID (Security Identifiers – идентификатор безопасности), она сохраняет и все свои свойства, например, описание, полное имя пароль, членство в группе и т. д.

Управление локальными группами.

Создание локальной группы.

Для создания локальной группы необходимо выполнить:

1. В окне оснастки Local User Manager установите указатель мыши на папке Groups (Группы) и нажмите правую кнопку. В появившемся контекстном меню выберите команду Create Group (Создать группу).

2. В поле Name (Имя) введите имя новой группы.

3. В поле Description (Описание) введите описание новой группы.

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 ‘\’.

Изменение членства локальной группы.

Чтобы добавить или удалить учетную запись пользователя из группы:

1. В окне оснастки Local User Manager щелкните на папке Groups.

2. В правом подокне установите указатель мыши на модифицируемую группу и нажмите правую кнопку. В появившемся контекстном Меню выберите команду Properties.

3. Для того чтобы добавить новые учетные записи в группу, нажмите кнопку Add. Далее следуйте указаниям окна диалога Select Users.

4. Для того чтобы уничтожить членов локальной группы, в поле Members (Члены) выберите одну или несколько учетных записей и нажмите кнопку Remove (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 уничтожены. Уничтоженные группы не могут быть восстановлены. При уничтожении групп их члены сохраняются.

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 окружения, параметров реестра и набора доступных приложений.

Для управления средой пользователя предназначены следующие средства Windows 2000:

- *Профили пользователей.* В профиле пользователя хранятся все настройки рабочей среды компьютера, на котором работает Windows 2000, определенные самим пользователем. Это могут быть, например, настройки экрана и соединения с сетью. Все настройки, выполняемые самим пользователем, автоматически хранятся в файле, путь к которому выглядит

следующим образом: *Имя устройства \корневой каталог \ Profiles.* Как правило, корневым является каталог winnt.

- *Сценарии*. Сценарий входа в сеть 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 запуска приложения. Кроме того, с помощью сценария можно устанавливать значения переменных окружения, указывающих пути поиска компьютера, каталог для временных файлов и другую подобную информацию.
- *Сервер сценариев Windows (WSH-Windows Scripting Host).* Сервер сценариев независим от языка и предназначен для работы на 32-разрядных 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 HTML- документ.

Профили пользователей.

На компьютере, где работает Windows 2000 Professional или автономный Windows 2000 Server, локальные профили пользователей создаются автоматически. Затем они используются 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:

- При регистрации пользователя в системе рабочий стол получает те же настройки, какие существовали в момент предыдущего выхода пользователя из системы.
- Несколько пользователей могут использовать один и тот же компьютер, и каждый из них будет работать в индивидуальной среде.
- 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 *блуждающими* (roaming).

Выступая в качестве инструментов администрирования, профили пользователя позволяют выполнять следующие функции:

- Можно создать несколько типов профилей и назначить их определенным группам пользователей. Это позволит получить несколько типов рабочих сред, соответствующих различным задачам, решаемым пользователями.
- Можно назначать общие групповые настройки всем пользователям.
- Можно назначать обязательные профили, не позволяющие пользователям изменять какие-либо настройки.

Настройки, хранящиеся в профиле пользователя.

Профиль пользователя хранит настройки конфигурации и параметры, индивидуально назначаемые каждому пользователю и полностью определяющие его рабочую среду (см. табл.).

Объект	Параметры
Windows NT Explorer	Все настройки, определяемые самим пользователем, касающиеся программы Windows NT Explorer.
Панель задач	Все персональные группы программ и их свойства, все программные объекты и их свойства, все настройки Панели задач.
Настройка принтера	Сетевые соединения принтера.
Панель управления	Все настройки, определенные самим пользователем, касающиеся Панели управления.
Стандартные	Настройки всех стандартных приложений, запускаемых для конкретного пользователя.
Приложения работающие в ОС Windows NT	Любое приложение, специально созданное для работы в среде Windows 2000, может обладать средствами отслеживания своих настроек относительно каждого пользователя. Если такая информация существует, она' хранится в профиле пользователя.
Электронная подсказка	Любые закладки, установленные в системе Windows 2000 Help.
Консоль управления Microsoft	Индивидуальный файл конфигурации и текущего состояния консоли управления.

Структура профиля пользователя.

При создании профиля пользователя используется профиль, назначаемый по умолчанию. Он хранится на каждом компьютере, где работает Windows 2000 Professional или Windows 2000 Server. Файл NTuser.dat, находящийся в папке Default User, содержит настройки конфигурации, хранящиеся в реестре Windows 2000. 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 Profiles корневого каталога (как правило, winnt). В папке Default User находятся файл NTuser.dat и список ссылок на объекты рабочего стола.

В папках профилей пользователей хранятся ссылки на различные объекты рабочего стола. Ниже перечислены папки, находящиеся внутри папки профиля пользователя.

- Application Data - Данные, относящиеся к конкретному приложению. Например, индивидуальный словарь. Разработчики приложений сами принимают решение, какие данные должны быть сохранены в папке профиля пользователя.
- Cookies – Данные о вспомогательных программах, предназначенных для работы с Internet.
- Desktop – Объекты рабочего стола, включая файлы и ярлыки.
- Favorites – Ярлыки часто используемых программ и папок.
- Local Settings – Данные о локальных настройках, влияющих на работу программного обеспечения компьютера.
- My Documents – Данные о документах и графических файлах, используемых пользователем.
- NetHood – Ярлыки объектов сетевого окружения.
- PrintHood – Ярлыки объектов папки принтера.
- Recent – Ярлыки недавно используемых объектов.
- SendTo – Ярлыки объектов, куда могут посылаться документы.
- Start Menu – Ярлыки программ.
- Templates – Ярлыки шаблонов.
- Temporary Internet Files - Временные файлы, создаваемые при работе в Internet.

Папки NetHood, PrintHood, Recent и Templates скрыты, поэтому по умолчанию они не видны в Windows NT Explorer. Для их просмотра выберите команду Folder Options (Пара метры папки) меню View (Вид), затем установите флажок Show All Files (Показать все файлы) на вкладке View.

Папка All Users.

Хотя настройки, находящиеся в папке All Users, не копируются в папки профиля пользователя, они 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 Profiles. Папка All Users также содержит настройки для рабочего стола и меню Start. Группы этого типа на компьютерах, где работает Windows 2000 Server и Windows 2000 Professional , могут создавать только члены группы Admil1istrators.

Создание локального профиля пользователя.

Локальный профиль пользователя хранится на компьютере в папке, имя которой совпадает с именем данного пользователя, находящейся в папке Profiles. Если для данного пользователя не существует сконфигурированного блуждающего (находящегося на сервере) профиля, 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 используется для обеспечения отказоустойчивости, позволяя Windows NT восстанавливать профиль пользователя в случае сбоя при модификации содержимого файла NTuser.dat.

Блуждающие профили пользователя.

Блуждающие профили пользователя могут быть созданы тремя способами:

- Каждой учетной записи назначается путь к профилю пользователя. В этом случае на сервере происходит автоматическое создание пустой папки профиля пользователя. Затем пользователь может сам создать свой профиль.
- Каждой учетной записи назначается путь к профилю пользователя. Затем в папку, указанную в пути, копируется приготовленный заранее профиль пользователя.
- Каждой учетной записи назначается путь к профилю пользователя. Затем, в папку, указанную в пути, копируется приготовленный заранее профиль пользователя. После этого файл NTuser.dat, путь к которому указан в каждой учетной записи, переименовывается в NTuser.man. В этом случае создается обязательный профиль пользователя.

Имя сервера, на котором будут находиться блуждающие профили пользователей, указывается с помощью Local User Manager и вкладки Profile окна свойств пользователя. В результате при завершении работы пользователя на компьютере его профиль сохраняется как на локальном компьютере, так и в папке на сервере, в соответствии с путем профиля. При следующей регистрации пользователя в сети, дата копии профиля, находящейся на сервере, сравнивается с копией, расположенной локально на компьютере. Если они отличаются, информация берется из более свежей копии. Блуждающий профиль находится в централизованном хранилище профилей в масштабах домена. Он может быть доступен только при условии работоспособности хранящего его сервера. В обратном случае используется локальная кэшированная копия профиля пользователя. Если пользователь первый раз зарегистрировался в компьютере, создается новый профиль. В любом случае, если хранящийся централизованно профиль пользователя недоступен, он не обновляется при завершении работы. При следующей регистрации в компьютере пользователю придется напрямую указать, какая копия профиля должна быть использована - более новая локальная или старая копия, находящаяся на сервере.

Для настройки блуждающих профилей пользователей, являющихся членами домена Windows 2000, используется оснастка Active Directory Manager, поскольку вся информация о пользователях домена хранится в каталоге. В остальном логика управления профилями остается неизменной (блуждающий профиль хранится в указанной папке на некотором общем сетевом ресурсе, а в случае его недоступности используется кэшированная копия с локального компьютера).

С помощью Local User Manager можно указать имя сервера, где будет храниться заранее созданный блуждающий профиль пользователя. Затем в окне System (Система), вызываемом из Control Panel, перейдите на вкладку User Profiles (Профили пользователя), нажмите кнопку Copy To (Скопировать на) и скопируйте профиль заранее созданного

профиля на сервер. При первой регистрации вместо профиля, установленного по умолчанию, пользователь получает копию заранее сконфигурированного профиля с сервера. В дальнейшем этот профиль функционирует так же, как любой стандартный профиль пользователя. Каждый раз, когда пользователь завершает работу, его профиль сохраняется локально и одновременно копируется на сервер.

Обязательный профиль представляет собой сконфигурированный заранее блуждающий профиль, который недоступен пользователю для модификации. Пользователь может изменять настройки рабочего стола, но при завершении работы на компьютере изменения не заносятся в профиль. При следующей регистрации на компьютере загружается обязательный профиль пользователя, в котором не произошло никаких изменений. Профиль пользователя становится обязательным, когда вы переименовываете файл NTuser.dat в Ntuser.man. В этом случае файл становится доступен только для чтения. Один обязательный профиль может быть использован большим количеством пользователей.

Когда для обеспечения безопасности или приведения рабочей среды пользователя в соответствие с его уровнем подготовки для работы на компьютере необходимо контролировать набор доступных функций, лучше использовать системные политики. С их помощью вы можете выбрать подмножество настроек, а также осуществлять контроль, как параметров среды пользователя, так и настроек компьютера.

Добавление пути профиля пользователя к учетной записи.

Добавить путь расположения профиля пользователя к учетной записи можно с помощью вкладки Profile окна свойств пользователя, открытого для определенной учетной записи в окне оснастки Local User Manager (или Active Directory Manager). Перейдите на вкладку Profile и добавьте путь профиля пользователя.

В учетной записи следует указать полный путь к профилю пользователя:

\\сервер\имя общего ресурса \имя профиля

В качестве общего ресурса может выступать папка **Profiles.** Затем следует организовать к ней общий доступ для группы Everyone. В качестве имени профиля следует указать имя папки профиля данного пользователя (это может быть любая папка на общем ресурсе, в которой будет храниться профиль). Путь профиля пользователя может указывать на любой сервер. Это не обязательно должен быть контроллер домена. Когда пользователь регистрируется в сети, Windows 2000 Server проверяет, указан ли в его учетной записи путь профиля. Если путь указан, система находит соответствующий профиль.

Копирование профиля пользователя на сервер.

Для того чтобы сделать определенный профиль доступным нескольким пользователям, скопируйте его на сервер с помощью вкладки **User Profiles** окна **System,** вызываемого из Панели управления. Место, куда скопирован профиль, должно совпадать с путем профиля, указанным в учетных записях пользователей.

В окне диалога **System Properties** (Свойства системы) перейдите на вкладку **User Profiles.** Все профили пользователей, созданные на компьютере, появятся в списке окна диалога **Profiles Stored On This Computer** (Профили, сохраненные на этом компьютере).

Для копирования определенного профиля пользователя перейдите на вкладку **Copy To** и введите имя целевой папки. В качестве альтернативы можно выбрать целевую папку с помощью службы просмотра.

Добавление пользователей и групп к списку разрешений блуждающего профиля пользователя.

С помощью окна System вместе с профилем пользователя копируются и соответствующие разрешения. Поэтому пользователь автоматически получает доступ к своему профилю. Однако если вы хотите, чтобы к профилю получили доступ другие пользователи и группы, необходимо добавить их в список объектов, которым разрешено использовать данный профиль. Для этого в списке **Profiles stored on this computer** выберите интересующий вас профиль и нажмите кнопку **Copy To**. Появится окно диалога **Copy To**. В группе **Permitted to use** показывается, кто имеет разрешение на использование данного профиля. Для того чтобы добавить нового пользователя или группу к списку разрешений профиля пользователя, нажмите кнопку **Change**.

Изменение типа профиля пользователя для работы по низкоскоростному соединению.

Пользователи, присоединяющиеся к сети по низкоскоростному соединению, например при использовании службы удаленного доступа, могут работать со своим локальным профилем, а не загружать его с сервера по сети, что значительно ускоряет процесс регистрации. В таком случае при регистрации появляется окно, позволяющее пользователю указать, какой профиль должен быть загружен.

Если вы уже зарегистрировались в сети, с помощью кнопки **Change Type** (Изменить тип) на вкладке **Profile** окна свойств системы можно изменить тип профиля пользователя с блуждающего на локальный и наоборот. Новые настройки останутся неизменными до их следующей модификации. При изменении профиля пользователя с блуждающего на локальный кэшированная локально копия вашего профиля будет загружаться при каждой

регистрации в компьютере. При каждом завершении работы все изменения также будут записываться в локальную копию профиля.

В случае если вы работаете по низкоскоростному соединению, можно установить флажок **Use cached profile оп low connections** (Использовать кэшированный профиль на низкоскоростных каналах).

Подготовка заранее настроенных блуждающих и обязательных профилей пользователя.

Хотя для создания заранее сконфигурированного блуждающего или обязательного профиля можно использовать любую учетную запись, часто удобнее иной подход. Например, если вы хотите создать три различных заранее настроенных блуждающих или обязательных профиля для трех отделов предприятия, следует создать три различные ссылочные учетные записи. Затем необходимо зарегистрироваться с помощью каждой из созданных учетных записей и тем самым создать три профиля пользователя для трех отделов. После этого опять зарегистрируйтесь с помощью учетной записи администратора и, используя Local User Manager, назначьте созданные профили индивидуальным пользователям или группам. Затем с помощью вкладки **User Profile** окна **System** Панели управления скопируйте созданные профили на соответствующий сервер.

Работа пользователей с различными конфигурациями оборудования.

Поскольку профили пользователей могут быть использованы на различных типах рабочих станций, следует помнить, что компьютеры могут отличаться по конфигурации оборудования. Особенно это относится к типам мониторов и видеокартам. Профиль пользователя может определять положение и размер окон, поэтому тип оборудования экрана в значительной степени влияет на качество работы профиля. Например, параметры окна, выводимого на экране типа Super VGA, могут быть неверны при выводе того же изображения на экране с типом VGA. Для предотвращения появления подобных проблем:

- При создании и редактировании профиля пользователя для определенного пользователя используйте компьютер, тип экрана которого совпадает с типом экрана компьютера пользователя.
- При создании обязательного профиля для нескольких пользователей создавайте один профиль для группы пользователей только в случае, если все члены группы работают на компьютерах с одинаковым типом экранов.

Уничтожение профиля пользователя.

Если вы больше не хотите использовать блуждающий или обязательный профиль, назначенный пользователям, с помощью Loca1 User Manager уничтожьте путь к нему в учетных записях соответствующих пользователей. Сам профиль пользователя, находящийся на сервере, можно уничтожить с помощью кнопки **Delete** (Удалить) на вкладке **User Profile** окна **System** Панели управления.

Использование сценариев входа для настройки рабочей среды пользователя.

Сценарии входа выполняются автоматически в процессе каждой регистрации пользователя на компьютере, работающем с программным обеспечением Windows 2000. Хотя чаще всего сценарий входа представляет собой командный файл с расширением .bat, в качестве сценария может быть использован и исполняемый файл (.exe).

Сценарии входа не являются обязательными. Они могут применяться для настройки рабочей среды пользователя, создания сетевого соединения или запуска приложений. Сценарии входа очень удобны, когда вам необходимо изменить некоторые параметры рабочей среды пользователя без выполнения ее полной настройки.

Профили пользователя могут в процессе регистрации восстанавливать существовавшие ранее соединения с сетью, но они не могут быть использованы для создания новых соединений.

Создание сценариев входа.

Для создания сценариев входа может быть использован обыкновенный текстовый редактор. Затем с помощью оснастки Local User Manager сценарии входа назначаются соответствующим пользователям. Кроме того, один сценарий может быть назначен нескольким пользователям. В таблице приведены параметры, значения которых можно устанавливать с помощью сценария входа, и их описания.

Параметр	Описание
%HOMEDRIVE%	Имя устройства локального компьютера, связанного с домашним каталогом пользователя
%HOMEPATH%	Полный путь к домашнему каталогу пользователя
%HOMESHARE%	Имя общего ресурса, где находится домашний каталог пользователя
%OS%	Операционная система компьютера пользователя
%PROCESSOR_ARCHITECTURE%	Тип процессора (например, Pentium) компьютера пользователя
%PROCESSOR_LEVEL%	Уровень процессора компьютера пользователя
%USERDOMAIN%	Домен, в котором находится учетная запись

	пользователя
%USERNAME%	Имя пользователя

Назначение сценариев входа учетным записям пользователей и групп.

Для того чтобы назначить сценарий входа учетным записям пользователей и групп, с помощью оснастки Local User Manager 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 **Profile** окна свойств учетной записи вы можете назначить сценарии входа учетным записям пользователей, введя в поле **Logon Script** (Сценарий входа) с клавиатуры имя файла и путь к нему. При регистрации сервер, аутентифицирующий пользователя, находит файл сценария (если таковой существует) с помощью указанного в учетной записи имени и пути к сценариям входа, определенного на сервере локально (как правило, это \\C:\WINNT\System32\Repl\Import\Scripts). Если перед именем файла указан относительный путь, сервер ищет сценарий входа в подкаталоге основного локального пути сценариев.

Данные поля Logon Script определяют только имя файла и относительный путь, но не содержат самого сценария входа. После создания файл сценария с определенным именем помещается в соответствующий каталог экспортирующего сервера репликации

Сценарий входа можно поместить в локальный каталог компьютера пользователя. Но подобный подход, как правило, используется 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. Путь сценариев входа для компьютера Windows 2000 имеет следующий вид:

Systemroot\System32\Repl\Import\Scripts

Помимо оснастки Local User Manager сценарии входа могут быть назначены и с помощью оснастки Group Policy Editor.

Переменные окружения пользователя.

При управлении множеством учетных записей пользователей и групп часто возникает необходимость одновременно выполнить одинаковые изменения в нескольких учетных записях. Вместо конкретных имен или меток в сценарий входа вводится одна общая переменная (переменная окружения пользователя), замещаемая реальными данными в процессе выполнения сценария.

Изменение системных переменных окружения.

Для поиска программ, выделения пространства памяти определенным программам и контроля программ операционная система Windows 2000 требует определенной информации, называемой *переменными окружения системы и пользователя.* Их можно просмотреть на вкладке **Advanced** окна **System** Панели управления, нажав кнопку Environment Variables. Эти переменные окружения похожи на переменные, которые устанавливались в операционной системе MS- DOS, например PATH и TEMP. Переменные окружения системы одинаково определяются в Windows 2000 независимо от того, кто зарегистрировался на компьютере. Если вы зарегистрировались как член группы Administrators, вы можете добавить новые переменные или изменить их значения

Переменные окружения пользователя могут устанавливаться индивидуально для каждого пользователя одного и того же компьютера. Сюда включаются любые переменные окружения, которые вы хотите определить, или переменные, определенные вашим приложением, например путь к файлам приложения. После изменения переменных окружения их новые величины сохранятся в реестре, после чего они становятся автоматически доступными при следующем запуске компьютера. Если между переменными окружения возникает конфликт, он разрешается следующим способом:

1. Устанавливаются системные переменные окружения.
2. Устанавливаются переменные, определенные в файле Autoexec.bat (за исключением переменных PATH). Они перезаписывают системные переменные.
3. Устанавливаются переменные окружения пользователя, определенные в окне System. Они перезаписывают как системные переменные, так и переменные файла Autoexec.bat.
4. Устанавливаются переменные PATH файла Autoexec.bat.

Системные переменные окружения как пути в профилях пользователей, домашних каталогах и сценариях входа.

Любая переменная окружения компьютера клиента, где работает программное обеспечение Windows 2000 Professional, может быть использована в пути профиля, задаваемого в учетной записи пользователя, пути сценария входа, пути домашнего каталога и внутри самого сценария входа. Для подобного использования системной переменной окружения ее следует заключить в знаки процента (%). Например, для того чтобы использовать в пути профиля пользователя переменную окружения servername. В поле **Profile Path** окна учетной записи следует ввести \\%servername%\scripts.

Подобный подход очень удобен при использовании профилей пользователя в сетях, включающих каналы WAN. Особенно если ваши пользователи работают с обеих сторон

этого канала. Предположим, что сеть состоит из двух площадок, разделенных глобальным каналом. На каждом из компьютеров, работающих на одной стороне канала, переменная servername устанавливается в соответствии с именем компьютера резервного контроллера домена данной площадки. На другой стороне канала переменная servername устанавливается аналогичным образом, но ее значение соответствует имени резервного контроллера домена данной площадки. Теперь, в пути сценария входа каждой учетной записи пользователя домена, используется %servername%. Когда пользователь регистрируется в сети, его сценарий входа загружается с сервера, определяемого переменной окружения, расположенного локально относительно глобального канала.

Сервер сценариев Windows (WSH).

Сервер сценариев Microsoft Windows (Windows Scripting Host, WSH) не зависит от языка сценария и устанавливается во всех системах Windows 2000 как стандартное средство. Он предназначен для 32-разрядных операционных систем Windows. Компания Microsoft разработала ядро сценариев как для Visual Basic, так и для Java Script. Специалисты Microsoft ожидают, что другие компании, разрабатывающие программное обеспечение, создадут ядро сценариев ActiveX для таких языков, как Perl, TCL, REXX и Python.

Сервер сценариев может быть запущен с помощью исполняемого файла для Windows (WSCRIPT.EXE) и с помощью команды оболочки (CSCRIPT.EXE).

Назначение сервера сценариев.

Сервер сценариев позволяет применять в операционных системах Windows простые мощные и гибкие сценарии. Раньше единственным языком сценариев, поддерживаемым операционной системой Windows, был язык команд MS-DOS (командный файл). Хотя это быстрый и компактный язык в сравнении с языками VBScript и Jscript, он обладает весьма ограниченными возможностями. В настоящее время архитектура сценариев ActiveX позволяет в полной мере использовать все средства таких языков сценариев, как VBScript и Jscript, одновременно сохраняя совместимость с набором команд MS-DOS.

Компания Microsoft поставляет три сервера, предназначенных для выполнения языков сценариев на платформах Windows:

- Internet Explorer
- Internet Information Server
- Windows Scripting Host

Internet Explorer позволяет выполнять сценарии на машинах клиентов внутри HTML-страниц. Internet Information Server поддерживает работу со страницами Active Server,

позволяющими выполнять сценарии на Web-сервере. Другими словами, выполнение сценариев на сервере становится возможным в сетях Internet и Intranet.

Сервер сценариев Windows позволяет выполнять сценарии прямо на рабочем столе операционной системы Windows или на командной консоли. При этом полностью отсутствует необходимость встраивать выполняемые сценарии в HTML- документ. Выполнение сценария на рабочем столе инициируется щелчком мыши на файле сценария. В процессе работы сервер сценариев чрезвычайно экономно использует память, что очень удобно для выполнения не интерактивных сценариев, например сценария входа в сеть, административного сценария, и автоматизации операций, выполняемых на машине.

Объектная модель сервера сценариев.

Объектная модель сервера сценариев Windows состоит из двух основных категорий интерфейсов ActiveX:

- Выполнение сценария и поиск ошибок - свойства и методы, напрямую связанные с выполнением сценария. Этот набор интерфейсов позволяет сценариям манипулировать сервером WSH, отображать сообщения и выполнять такие основные функции, как, например, CreateObject и GetObject.
- Функции справки: - свойства и методы, которые подключают сетевые устройства, присоединяются к принтерам, считывают значения переменных окружения и модифицируют их, манипулируют ключами реестра. Это позволяет администраторам использовать WSH для создания простых сценариев входа в систему.

Кроме объектных интерфейсов, содержащихся в сервере сценариев Windows, администраторы могут использовать любые элементы управления ActiveX, что позволяет работать с интерфейсами автоматизации и выполнять различные задачи на платформе Windows. Например, администраторы могут написать сценарии, использующие Active Directory для управления Windows 2000.

Создание ядра для сервера сценариев Windows.

Основное требование, предъявляемое к ядру сценариев, разрабатываемому третьей фирмой, заключается в обязательности поддержки интерфейса ActiveX IActiveScriptParse. Сервер сценариев Windows читает содержимое файла сценария и с помощью метода IActiveScriptParse::ParseScriptText передает его зарегистрированному ядру. Ядро сценария может вызвать элемент управления ActiveX Wshom.ocx, поддерживающий интерфейс COM IDispatch. Если ядро поддерживает создание в сценарии элементов управления ActiveX, Wshom.ocx может быть вызван из самого сценария.

Если Wshom.ocx вызывается ядром сценариев, следует использовать модули Iwshom.idl, Iwshom.h и Iwshom.iid, которые можно найти в Platform SDK. С помощью этих файлов вы сможете вызывать методы, находящиеся в Wshom.ocx, используя обычную вызывающую последовательность COM. Wshom.ocx поддерживает двойной интерфейс и интерфейс COM IerrorInfo.

Запуск сервера сценариев из командной строки.

Для запуска сервера сценариев из командной строки используйте утилиту CSCRIPT.EXE в соответствии со следующим синтаксисом

Cscript *[параметры сервера сценариев]* имя *сценария [параметры сценария]*

где

Параметры сервера сценариев включают и отключают различные средства сервера сценариев. Они всегда предваряются двумя слэшами (//).

Имя сценария - это имя файла сценария, например, CHART.VBS.

Параметры сценария передаются в сценарий. Они всегда предваряются одним слэшем.

Ни один из параметров не является обязательным. Однако нельзя указать параметры сценария без указания самого сценария. Если вы не указываете ни одного параметра, CSCRIPT.EXE выдает на экран синтаксис своего запуска и допустимые параметры сервера сценариев (см. табл.).

Параметр	Описание
//I	Интерактивный режим (выбирается по умолчанию; режим, обратный задаваемому параметром //B)
//B	Пакетный режим. Не отображает на экране сообщений об ошибках и приглашения пользователей
//T:nn	Тайм-аут в секундах. Максимальное время, в течение которого может выполняться сценарий. (По умолчанию ограничение не устанавливается). Этот параметр используется для предотвращения слишком длительного выполнения сценариев. Устанавливается специальный таймер. Когда время выполнения превышает установленное значение, CSCRIPT прерывает работу ядра сценариев с помощью метода lactiveScript::InterruptThread и завершает процесс.
//logo	Отображает на экране сообщение о выполнении (выбирается по умолчанию; режим, обратный задаваемому пара метром //NoLogo)
//nologo	Запрещает вывод сообщения о выполнении
//H:Cscript или Wscript	Устанавливает CSCRIPT.EXE или WSCRIPT.EXE приложением, выбираемым по умолчанию для выполнения сценариев. По умолчанию устанавливается WSCRIPT.EXE
//S	Сохраняет текущие параметры командной строки для этого пользователя
//?	Показывает пара метры и синтаксис команды CSCRIPT.EXE

Дистрибутив сервера сценариев содержит несколько простых примеров сценариев. Например, для того чтобы запустить CHART.VBS:

1. В меню Start выберите команду Programs и запустите командную строку MS- DOS.

2. В командной строке выполните следующие команды

cscript //logo "Устройство: "\ "Здесь находится ваш каталог с примерами "\chart.vbs

cscript //nologo "Устройство: "\ "Здесь находится ваш каталог с примерами "\chart.vbs

В операционной системе Windows 2000 не нужно указывать расширение сценариев. Здесь можно просто набрать с клавиатуры имя сценария или щелкнуть на нем мышью в окне Explorer.

Запуск сценариев с помощью сервера, работающего в среде Windows.

Сценарий в среде Windows можно запустить тремя способами:

- Двойным щелчком мыши на файле сценария или соответствующем значке в окнах **My Computer, Explorer** или **Find;**
- В окне **Run** введите с клавиатуры полное имя выполняемого сценария и нажмите кнопку **ОК;**
- В окне **Run** введите WSCRIPT.EXE с указанием имени сценария и необходимых параметров сервера и сценария.

При запуске сценария с помощью WSH можно указать, какое приложение следует использовать - CSCRIPT.EXE или WSCRIPT.EXE. Приложение сервера, выбираемое по умолчанию, может быть установлено с помощью команды cscript/*//H:Имя_сервера_сценариев.*

Например, если вы устанавливаете в качестве приложения, выбираемого по умолчанию, WSCRIPT.EXE и выполняете сценарий с именем CHART.VBS, WSCRIPT. EXE будет выбираться по умолчанию для всех файлов сценариев, имеющих расширение .vbs.

Страница свойств сервера сценариев Windows позволяет устанавливать параметры, приведенные ниже.

Свойство	Применение	Эквивалент параметра команды cscript
Stop scripts after specified number of seconds (Остановить сценарий после *nn* секунд)	Максимальное количество секунд, в течение которых можно выполнять сценарий (По умолчанию ограничение не устанавливается)	*//T:nn*
Display logo when scripts executed in command console . (Отображать сообщение-логотип при выполнении сценариев из командной строки)	Отображать логотип утилиты. (Обратное параметру //nologo. Устанавливается по умолчанию)	//logo или //nologo

Настройка индивидуальных свойств сценария.

Файл с расширением .wsh.

С помощью страницы свойств модуля WSCRIPT.EXE можно установить глобальные параметры, касающиеся сразу всех сценариев, выполняемых на локальной машине. Однако также можно настроить индивидуальные параметры отдельно взятого сценария, позволяющие осуществлять жесткий контроль его выполнения. Свойства конкретного сценария сохраняются в файле с расширением .wsh. Для его создания просто установите указатель мыши на файле сценария в окне Explorer и нажмите правую кнопку. В появившемся контекстном меню выберите команду Properties. Настройте индивидуальные свойства сценария, например максимальное время исполнения, и нажмите кнопку **ОК**. В результате в каталоге, где находится сценарий, будет создан файл с расширением .wsh, имя которого совпадает с именем сценария. Он содержит индивидуальные настройки сценариев для WSH. Функции этого файла сходны с функциями файла PIF 16-разрядных приложений.

Для запуска сценария, для которого создан файл с расширением .wsh, следует дважды щелкнуть мышью на файле *.wsh в окне Explorer или использовать этот файл в качестве параметра для программ WSCRIPT.EXE или CSCRIPT.EXE в командной строке. Например:

C:\>cscript Myscript.wsh

Поскольку в файле с расширением .wsh хранятся значения параметров, используемых сценарием при выполнении, системный администратор может создать несколько версий файла с параметрами, ориентированных на различные группы пользователей внутри организации. Набор файлов с расширением .wsh, относящийся к одному сценарию, может быть использован следующим образом:

- Администратор может создать отдельный файл .wsh для определенной группы пользователей внутри организации. Это позволит осуществлять индивидуальный контроль определенных сценариев, выполняющихся в течение дня.
- Администратор может создать индивидуальные файлы .wsh для конкретных пользователей внутри организации. Это позволяет осуществлять полный контроль ряда сценариев, используемых внутри организации.
- Индивидуальные файлы с расширением .wsh могут быть созданы для сценариев входа пользователей в систему. Это позволяет администратору осуществлять индивидуальный контроль над рядом свойств сценариев, выполняемых на клиентских машинах при регистрации пользователя в системе.

Файл с расширением .wsh представляет собой простой текстовый файл, формат которого сходен с форматом файла с расширением .inf. Ниже приведен пример содержимого файла .wsh:

```
[ScriptFile]
Path=C:\WINNT\Samples\WSH\showprop.vbs
[Options]
Timeout=0
DisplayLogo=l
BatchMode=0
```

Параметр Path В разделе [ScriptFile] определяет местоположение файла сценария, с которым связан данный файл .wsh. Параметры, значения которых устанавливаются в разделе [Options], соответствуют настройкам вкладки **Script** окна **Properties.**

После двойного щелчка мышью на файле с расширением .wsh или выполнения его в командной строке программа CSCRIPT.EXE или WSCRIPT.EXE считывает его и определяет специфические параметры, которые следует использовать при выполнении соответствующего сценария. В результате сценарий будет выполняться с необходимыми параметрами, заданными в файле .wsh. Обратите внимание, что при запуске файла с расширением .wsh необходимо присутствие соответствующего ему сценария. Если выполнение сценария посредством файла .wsh закончилось неудачно, проверьте запись Path=. Она должна указывать на тот сценарий, который вы хотите выполнить.

Порядок выполнения части 1.

1. Ознакомиться с теоретическими сведениями о *консоли управления Microsoft* ОС Windows 2000 и основных оснастках.

2. Выполнить основные настройки, связанные с управлением пользователей и группами.

Отчет должен включать:

- название работы и ее цель;
- описание шагов настройки.

Альтернативный вариант – *пройти тест в день выполнения данной работы* (в этом случае отчет не нужен).

Часть 2. Исследование методов защиты.

Степень защищенности компьютера во многом зависит от совершенства установленной на нем операционной системы, которая должна обладать развитыми средствами шифрования данных и иметь инструменты настройки безопасности, как для одиночного компьютера, так и компьютера работающего в сети.

Цель работы - исследование методов защиты операционных систем и информации в операционной системе Windows NT, 2000 и выше.

Продолжительность работы - 4 часа.

Краткие теоретические сведения

Для защиты информации используется целый комплекс средств, включающий в себя технические, программно-аппаратные средства и административные меры. По мере развития средств защиты компьютерных систем развиваются и средства нападения.

Для примера рассмотрим атаки защищенной системы посредством программных закладок. *Программная закладка* – это программа или фрагмент программы, скрытно внедряемый в защищенную систему и позволяющий злоумышленнику осуществлять в несанкционированный доступ к ресурсам защищенной системы.

Основная опасность программных закладок заключается в том, что, являясь частью защищенной системы, она способна принимать активные меры по маскировке своего присутствия в системе. При внедрении в систему закладки в защищенной системе создается скрытый канал информационного обмена, который, как правило, остается незамеченным для администраторов системы в течение длительного времени. Практически все известные программные закладки, применявшиеся в разное время различными злоумышленниками, были выявлены либо из-за ошибок, допущенных при программировании закладки, либо чисто случайно.

Наиболее распространенный класс программных закладок – закладки, перехватывающие пароли пользователей операционных систем (*перехватчики паролей*). Перехватчики паролей были разработаны в разное время для целого ряда операционных систем, включая многие версии Windows и UNIX. Перехватчик паролей, внедренный в ОС, получает доступ к паролям, вводимым пользователями при входе в систему. Перехватив очередной пароль, закладка записывает его в специальный файл или в любое другое место, доступное злоумышленнику, внедрившему закладку в систему.

Перехватчики паролей ***первого рода*** действуют по следующему алгоритму. Злоумышленник запускает программу, которая имитирует приглашение пользователю для входа в систему и ждет ввода. Когда пользователь вводит имя и пароль, закладка

сохраняет их в доступном злоумышленнику месте, после чего завершает работу и осуществляет выход из системы пользователя-злоумышленника (в большинстве операционных систем выход пользователя из системы можно осуществить программно). По окончании работы закладки на экране появляется настоящее приглашение для входа пользователя в систему.

Пользователь, ставший жертвой закладки, видит, что он не вошел в систему, и что ему снова предлагается ввести имя и пароль. Пользователь предполагает, что при вводе пароля произошла ошибка, и вводит имя и пароль повторно. После этого пользователь входит в систему, и дальнейшая его работа протекает нормально. Некоторые закладки, функционирующие по данной схеме, перед завершением работы выдают на экран правдоподобное сообщение об ошибке, например: “Пароль введен неправильно. Попробуйте еще раз”.

Основным достоинством этого класса перехватчиков паролей является то, что написание подобной программной закладки не требует от злоумышленника никакой специальной квалификации. Любой пользователь, умеющий программировать хотя бы на языке BASIC, может написать такую программу за считанные часы.

Перехватчики паролей первого рода представляют наибольшую опасность для тех операционных систем, в которых приглашение пользователю на вход имеет очень простой вид. Например:

```
login: user

password:
```

Задача создания программы, подделывающей такое приглашение, тривиальна.

Усложнение внешнего вида приглашения на вход в систему несколько затрудняет решение задачи перехвата паролей, однако не создает для злоумышленника никаких принципиальных трудностей. Для того, чтобы существенно затруднить внедрение в систему перехватчиков паролей первого рода, необходимы более сложные меры защиты. Примером операционной системы, где такие меры реализованы, является Windows NT.

В Windows NT обычная работа пользователя и аутентификация пользователя при входе в систему осуществляются на разных *рабочих полях* (desktops). Рабочее поле Windows NT представляет собой совокупность окон, одновременно видимых на экране. Только процессы, окна которых расположены на одном рабочем поле, могут взаимодействовать между собой, используя средства Windows GUI. Понятие рабочего поля Windows NT близко к понятию терминала UNIX.

Процесс Winlogon, получающий от пользователя имя и пароль, выполняется на отдельном рабочем поле (*рабочем поле аутентификации*). Никакой другой процесс, в том числе и перехватчик паролей, не имеет доступа к этому рабочему полю. Поэтому приглашение пользователю на вход в систему, выводимое перехватчиком паролей первого рода, может располагаться только на рабочем поле прикладных программ, где выполняются все программ, запущенные пользователем.

Переключение экрана компьютера с одного рабочего поля на другое производится при нажатии комбинации клавиш Ctrl-Alt-Del. Win32-подсистема Windows NT обрабатывает эту комбинацию по-особому – сообщение о нажатии Ctrl-Alt-Del посылается только процессу Winlogon. Для всех других процессов, в частности, для всех прикладных программ, запущенных пользователем, нажатие этой комбинации клавиш совершенно незаметно.

При старте системы на экран компьютера вначале отображается рабочее поле аутентификации. Однако пользователь вводит имя и пароль не сразу, а только после нажатия Ctrl-Alt-Del. Когда пользователь завершает сеанс работы с системой, на экран также выводится рабочее поле аутентификации, и, так же как и в предыдущем случае, новый пользователь может ввести пароль для входа в систему только после нажатия Ctrl-Alt-Del.

Если в систему внедрен перехватчик паролей первого рода, то, для того, чтобы он смог перехватить пароль пользователя, он должен по крайней мере обработать нажатие пользователем Ctrl-Alt-Del. В противном случае при нажатии пользователем этой комбинации клавиш произойдет переключение на рабочее поле аутентификации, рабочее поле прикладных программ станет неактивным, и перехватчик паролей просто не сможет ничего перехватить – сообщения о нажатии пользователем клавиш будут приходить на другое рабочее поле. Однако для всех прикладных программ факт нажатия пользователем Ctrl-Alt-Del всегда остается незамеченным. Поэтому пароль будет воспринят не программной закладкой, а процессом Winlogon.

Для обеспечения надежной защиты от перехватчиков паролей первого рода необходимо два условия:

1. Программа, получающая от пользователя имя и пароль при входе в систему, выполняется на изолированном терминале (*терминале аутентификации*), недоступном прикладным программам.
2. Факт переключения пользовательской консоли на терминал аутентификации незаметен прикладным программам. Прикладные программы не могут запретить переключение консоли на терминал аутентификации.

Перехватчики паролей ***второго рода*** перехватывают все данные, вводимые пользователем с клавиатуры. Простейшие программные закладки данного типа просто сбрасывают все эти данные на жесткий диск компьютера или в любое другое место, доступное злоумышленнику. Более совершенные закладки анализируют перехваченные данные и отсеивают информацию, заведомо не имеющую отношения к паролям.

Такие закладки представляют собой резидентные программы, перехватывающие одно или несколько прерываний процессора, имеющих отношение к работе с клавиатурой. Информация о нажатой клавише и введенном символе, возвращаемая этими прерываниями, используется закладками для своих целей.

В конце 1997 года на хакерских серверах Internet появились перехватчики паролей второго рода для Windows 3.x и Windows 95.

Программные интерфейсы Win16 и Win32 поддерживают специальный механизм фильтров (hooks), который может быть использован для перехвата паролей пользователей. С помощью этого механизма прикладные программы и сама операционная система решают целый ряд задач, в том числе и задачу поддержки национальных раскладок клавиатуры. Любой русификатор клавиатуры, работающий в среде Windows, перехватывает всю информацию, вводимую пользователем с клавиатуры, в том числе и пароли. Несложно написать русификатор так, чтобы он, помимо своих основных функций, выполнял бы и функции перехватчика паролей. Написание программы локализации клавиатуры является достаточно простой задачей.

Также Windows поддерживает цепочки фильтров, с помощью которых несколько программ могут одновременно получать доступ к информации, вводимой с клавиатуры, и обрабатывать ее так, как считают нужным, при необходимости передавая обработанную информацию дальше по цепочке. Можно встроить перехватчик паролей в цепочку фильтров перед русификатором или после него, так, что вся информация, вводимая пользователем с клавиатуры, проходит и через русификатор, и через перехватчик паролей.

В большинстве случаев верно следующее утверждение. *Если ОС допускает переключение раскладки клавиатуры при вводе пароля, то для этой ОС можно написать перехватчик паролей второго рода*. Действительно, если для операционной системы существует программа локализации раскладки клавиатуры, и если эта программа используется при вводе пароля, после незначительного изменения исходного текста эта программа превращается в перехватчик паролей второго рода.

Для организации защиты от перехватчиков паролей второго рода необходимо добиться выполнения в ОС следующих трех условий:

1. Переключение раскладки клавиатуры в процессе ввода пароля невозможно. В противном случае задача создания перехватчика паролей второго рода существенно упрощается.
2. Конфигурирование цепочки программных модулей, участвующих в получении операционной системой пароля пользователя, доступно только администраторам системы.

3. Доступ на запись к файлам программных модулей, участвующих в получении пароля пользователя, не предоставляется никому. Доступ на запись к атрибутам защиты этих файлов предоставляется только администраторам. Любые обращения с целью записи к этим файлам, а также к их атрибутам защиты, регистрируются в системном журнале аудита.

Для выполнения перечисленных условия выполнялись, необходимо, чтобы подсистема защиты операционной системы поддерживала разграничение доступа и аудит.

К перехватчикам паролей ***третьего рода*** относятся программные закладки, полностью или частично подменяющие собой подсистему аутентификации операционной системы. Поскольку задача создания такой программной закладки гораздо сложнее, чем задача создания перехватчика паролей первого или второго рода, этот класс программных закладок появился недавно.

Перехватчик паролей третьего рода может быть написан для любой многопользовательской операционной системы. Сложность создания такого перехватчика паролей зависит от сложности алгоритмов, реализуемых подсистемой аутентификации, сложности интерфейса между ее отдельными модулями, а также от степени документированности подсистемы аутентификации операционной системы. В целом задача создания перехватчика паролей третьего рода гораздо сложнее задачи создания перехватчика паролей первого или второго рода.

Поскольку перехватчики паролей третьего рода частично берут на себя функции подсистемы защиты операционной системы, перехватчик паролей третьего рода при внедрении в систему должен выполнить по крайней мере одно из следующих действий:

- подменить собой один или несколько системных файлов;
- внедриться в один или несколько системных файлов по одному из “вирусных” алгоритмов;

- использовать поддерживаемые операционной системой интерфейсные связи между программными модулями подсистемы защиты для встраивания себя в цепочку программных модулей, обрабатывающих введенный пользователем пароль;
- использовать для той же цели низкоуровневые интерфейсные связи операционной системы, используемые подсистемой защиты для решения своих задач.

Каждое из этих действий оставляет в операционной системе следы, которые могут быть выявлены с помощью следующих мер защиты:

1. Соблюдение адекватной политики безопасности. Подсистема аутентификации должна быть самым защищенным местом операционной системы. Меры, необходимые для поддержания адекватной политики безопасности, сильно различаются для разных операционных систем.
2. Контроль целостности исполняемых файлов операционной системы. Необходимо контролировать не только файлы, входящие в состав подсистемы защиты, но и библиотеки, содержащие низкоуровневые функции операционной системы.
3. Контроль целостности интерфейсных связей внутри подсистемы защиты, а также интерфейсных связей, используемых подсистемой защиты для решения низкоуровневых задач.

Построение абсолютно надежной защиты против перехватчиков паролей третьего рода представляется невозможным. Поскольку машинный код перехватчиков паролей третьего рода выполняется не в контексте пользователя, а в контексте операционной системы, перехватчик паролей третьего рода может принимать меры, затрудняющие его обнаружение администраторами системы, в частности:

- перехват системных вызовов, которые могут использоваться администраторами для выявления программной закладки в целях подмены возвращаемой информации;
- фильтрация регистрируемых сообщений аудита.

Для того чтобы защита от перехватчиков паролей была эффективной, политика безопасности, принятая в системе, должна удовлетворять следующим требованиям:

1. Конфигурирование цепочки программных модулей, участвующих в получении операционной системой пароля пользователя, доступно только администраторам системы.
2. Доступ на запись к файлам этих программных модулей предоставляется только администраторам системы.
3. Конфигурирование подсистемы аутентификации доступно только администраторам системы.
4. Доступ на запись к системным файлам предоставляется только администраторам системы.

1. Общие вопросы безопасности.

Операционная система Windows 2000 обладает развитыми средствами шифрования данных с *открытым ключом* (public key), представляющими собой дальнейшее развитие служб шифрования информации Windows NT, входящих в состав более ранних версий операционной системы.

Windows 2000 располагает интегрированным набором служб и инструментов администрирования, предназначенных для создания, реализации и управления приложениями, использующими алгоритмы шифрования с *открытым* ключом. Это позволит независимым разработчикам программного обеспечения применять в своих продуктах технологию *общего ключа* (shared key).

Применение алгоритмов шифрования с открытым ключом.

Криптография - это наука о защите данных. Алгоритмы криптографии математическими методами комбинируют входной открытый текст и ключ шифрования, генерируя, таким образом, зашифрованные данные. Применяя хороший алгоритм шифрования, можно сделать практически невозможным с точки зрения необходимых вычислительных и временных ресурсов, взлом защиты и получения открытого текста подбором ключа. Для быстрого выполнения подобного преобразования необходим *расшифровывающий ключ.* В традиционном шифровании с секретным ключом (симметричном шифровании) зашифровывающий и расшифровывающий ключи совпадают. Стороны, обменивающиеся зашифрованными данными, должны знать секретный ключ. Процесс обмена информацией о секретном ключе является узким местом в вопросах безопасности.

Фундаментальное отличие шифрования с открытым ключом заключается в том, что зашифровывающий и расшифровывающий ключи не совпадают. Шифрование информации является односторонним процессом: открытые данные шифруются с помощью зашифровывающего ключа, однако с помощью того же ключа нельзя осуществить обратное преобразование и опять получить открытые данные. Для этого необходимо применить расшифровывающий ключ, который *связан* с зашифровывающим ключом, но *не совпадает* с ним. Подобная технология шифрования предполагает, что каждый пользователь имеет в своем распоряжении пару ключей - открытый ключ и личный ключ. Открыто распространяя открытый ключ, вы даете возможность другим пользователям посылать вам зашифрованные данные, которые могут быть расшифрованы с помощью известного только вам личного ключа. Аналогично с помощью личного ключа вы можете преобразовать данные так, чтобы другая сторона убедилась в том, что

информация пришла именно от вас. Эта возможность применяется при работе с электронными подписями.

Появление пары "личный ключ/открытый ключ" привело к возникновению нескольких новых технологий. Наиболее важными из которых являются электронные подписи, распределенная аутентификация, соглашение о секретном ключе и шифрование больших объемов данных без предварительного соглашения о секретном ключе.

Существует несколько хорошо известных алгоритмов шифрования с открытым ключом. Некоторые из них, например RSA (Rivest-Shamir- Adelman) и шифрование с помощью эллиптической кривой (Elliptic Curve Criptography, ECC), являются алгоритмами общего употребления в том смысле, что они поддерживают все упомянутые выше операции. Другие алгоритмы поддерживают только некоторые операции. К ним относятся: алгоритм электронной подписи (Digital Signature Algorithm, DSA), используемый только для работы с электронными подписями, и алгоритм Diffie-Hellman (D-H), применяемый только для соглашения о секретных ключах.

Электронные подписи.

Наиболее ярким проявлением всех преимуществ шифрования с открытым ключом является технология *электронных подписей.* Она основана на математическом преобразовании, комбинирующим данные с секретным ключом таким образом, что:

- Только владелец секретного ключа может создать электронную подпись.
- Любой пользователь, обладающий соответствующим открытым ключом, может проверить истинность электронной подписи.
- Любая модификация подписанных данных (даже изменение одного бита) делает неверной электронную подпись.

Электронные подписи представляют собой данные, которые могут быть переданы вместе с подписанной информацией. Кроме того, цифровая подпись позволяет проверить, что данные не были изменены при передаче.

Шифрование с открытым ключом применяется для создания надежной службы *распределенной аутентификации,* гарантирующей, что данные пришли получателю от истинного корреспондента.

Шифрование с открытым ключом позволяет двум сторонам, используя открытый ключ в незащищенной сети, договориться о секретном ключе. Обе стороны посылают друг другу половины секретного ключа, зашифрованных соответствующими открытыми ключами. Каждая из сторон получает возможность расшифровать полученную половину секретного ключа и, соединив ее со своей половиной ключа, получить весь секретный ключ.

Поскольку алгоритмы шифрования с открытым ключом требуют значительно больших, по сравнению с алгоритмами секретного ключа, вычислительных ресурсов, они плохо подходят для шифрования больших объемов данных. Поэтому существует технология, комбинирующая оба алгоритма. В соответствии с ней весь объем данных шифруется с помощью секретного ключа, который в свою очередь шифруется с открытым ключом и посылается корреспонденту вместе с зашифрованными данными. Приемная сторона расшифровывает секретный ключ с помощью своего личного ключа, а затем расшифровывает данные с помощью полученного секретного ключа.

При шифровании с открытым ключом жизненно важным является абсолютно достоверная ассоциация открытого ключа и передавшей его стороны, поскольку в обратном случае возможна подмена открытого ключа и осуществление несанкционированного доступа к передаваемым зашифрованным данным. Необходим механизм, гарантирующий достоверность корреспондента. Одним из таких механизмов является применение сертификата, созданного авторизованным генератором сертификатов.

Сертификат - это средство, позволяющее гарантированно установить связь между переданным открытым ключом и передавшей его стороной, владеющей соответствующим личным ключом. Сам сертификат представляет собой набор данных, закрытых цифровой подписью. Информация сертификата подтверждает истинность открытого ключа и владельца соответствующего личного ключа.

Обычно сертификаты содержат дополнительную информацию, позволяющую идентифицировать владельца личного ключа, соответствующего данному открытому ключу. Сертификат должен быть подписан авторизованным генератором сертификатов.

Наиболее распространенным на данный момент стандартом сертификатов является ITU-T X.509. Это фундаментальная технология, используемая в Windows 2000. Однако это не единственная форма сертификатов.

Поставщик сертификатов, или центр авторизации (Certificate authority - CA) это организация или служба, создающая сертификаты. CA выступает в качестве гаранта истинности связи между открытым ключом субъекта и идентифицирующей этот субъект информацией, содержащейся в сертификате. Различные CA могут применять для проверки связи различные средства, поэтому перед выбором достойного доверия CA важно хорошо понять политику данного CA и применяемые им процедуры проверки.

После получения подписанного сообщения встает вопрос: насколько данной подписи можно доверять? Действительно ли подпись была поставлена тем, кого она представляет? Математическую верность подписи можно проверить после получении подписанного сообщения. Для этого применяется открытый ключ. Но при этом нет полной уверенности в

том, что используемый открытый ключ действительно принадлежит корреспонденту, от которого получено подписанное сообщение. Возникает необходимость проверки принадлежности открытого ключа. Она может быть проведена с помощью сертификата, созданного центром авторизации, пользующимся доверием у стороны, получившей подписанное сообщение. В сертификате должна содержаться следующая информация:

- Криптографически верная подпись, идентифицирующая создателя сертификата;
- Подтверждение связи между стороной, приславшей подписанное сообщение, и ее открытым ключом;
- Сертификат должен быть создан CA, которому приемная сторона доверяет.
- Истинность полученного сертификата может быть проверена с открытым ключом, принадлежащим создавшему этот сертификат CA. Однако здесь возникает еще одна проблема. А действительно ли открытый ключ принадлежит данному CA? Для получения ответа на этот вопрос необходимо применить еще один сертификат. В результате возникает цепочка сертификатов, начинающаяся с сертификата открытого ключа стороны, передавшей подписанное сообщение, и заканчивающаяся сертификатом CA, которому приемная сторона безоговорочно доверяет. Такой сертификат называется *корневым сертификатом доверия* (trusted root certificate), поскольку он формирует *корень* (верхний узел) *иерархии открытых ключей и идентификаторов связи,* которую приемная сторона рассматривает как достойную доверия. Если приемная сторона определила CA, которому она будет безоговорочно доверять, то полным доверием будут пользоваться и все дочерние CA, идентифицированные корневым сертификатом доверия.

Описанная выше модель сертификации истинности передающей стороны предполагает, что секретно приемная сторона должна получить только информацию о наборе корневых сертификатов доверия.

Компоненты Windows 2000 для шифрования с открытым ключом.

На рис. 1 показана взаимосвязь средств Windows 2000, позволяющих применять шифрование с открытым ключом.

Рисунок 1 – Схема взаимосвязи средств шифрования.

Изображенные средства могут находиться на одном компьютере и эффективно работать. Ключевым звеном всей схемы является *служба сертификатов Microsoftt* (Microsoft Certificate Services). Она позволяет создать один или несколько СА предприятия, поддерживающих создание и отзыв сертификатов. Они интегрированы в Active Directory, где хранится информация о политике СА и их местоположении. Кроме того, с помощью Active Directory выполняется публикация информации о сертификатах и их отзыве.

Средства работы с открытым ключом не заменяют существующих механизмов доверительных отношений между доменами и аутентификации, основанных на контроллерах доменов и центра распространения ключей Kerberos (Key *Oistriubution Center*, КОС). Напротив, данные средства взаимодействуют с этими службами, что позволяет приложениям безопасно передавать конфиденциальную информацию по Internet и корпоративным глобальным каналам

Поддержка прикладных средств шифрования информации с открытым ключом включена в состав программного обеспечения операционных систем Windows 2000/NT/95/98. На рис. 2 показана структура служб, предназначенных для поддержки прикладных программ.

Рисунок 2 – Службы средств шифрования информации с открытым ключом, поддерживающие прикладные программы

Основанием архитектуры прикладных средств шифрования информации с открытым ключом является библиотека Crypto API. Она позволяет работать со всеми устанавливаемыми провайдерами услуг шифрования (Cryptographic Service Providers, CSP) через стандартный интерфейс. Услуги CSP могут быть реализованы на программном уровне или с помощью специального оборудования. Они поддерживают различные длины ключей и алгоритмы шифрования. Как видно на рис. 2, один из CSP поддерживает смарт-карты. Услугами служб шифрования пользуются службы управления сертификатами. Они соответствуют стандарту X.509 v3 и позволяют организовывать принудительное хранение, службу подсчета и поддержку расшифровывания. Кроме того, эти службы предназначены для работы с различными промышленными стандартами сообщений. В основном они поддерживают стандарты PKCS и разработанный IETF (Internet Ellgil1eering Task Force) еще сырой набор стандартов РЮХ (Public Key Infrastructure, X.509).

Остальные службы используют Crypto API для придания дополнительной функциональности прикладным программам. Защищенный канал (Secure Channel) поддерживает сетевую аутентификацию и шифрование в соответствии со стандартными протоколами TLS и SSL, обращение к которым может быть выполнено с помощью интерфейсов Microsoft WinInet и SSPI. Служба Autl1enticode предназначена для выполнения проверки и подписи объектов, что в основном используется при получении информации через Internet.

В состав программного обеспечения служб поддержки прикладных средств шифрования входит поддержка интерфейса, предназначенного для работы со смарт-картами общего назначения. Они используются при программно независимой интеграции криптографических смарт-карт для выполнения регистрации в компьютере и сети Window 2000.

Политика безопасности шифрования с открытым ключом.

Политики безопасности действуют в масштабах области (site) каталога Active Directory, домена или организационной единицы и влияют на все ассоциированные с этими объектами группы, компьютеры и пользователей. Безопасность шифрования с открытым ключом является одним из аспектов общей политики безопасности Windows 2000 и интегрирована в ее структуру. Это механизм, с помощью которого можно посредством объектов политики безопасности централизованно осуществлять настройку и управление глобальной политикой работы с открытым ключом.

С помощью политики открытого ключа можно определять следующие аспекты безопасности Windows 2000:

- Доверенные корни CA;
- Регистрация и обновление сертификатов;
- Регистрация в системе с помощью смарт-карты.

Групповые политики безопасности.

В предыдущих версиях Windows NT Server *политика безопасности домена* хранилась в базе данных Диспетчера безопасности учетных записей SAM (Security Account Manager). Политика состояла из *дескриптора безопасности,* предоставляющего доступ к выполнению операций (таких, например, как создание учетной записи и просмотр учетных записей) и *свойств,* описывающих политики в отношении паролей и блокировки учетных записей пользователей. *Локальная политика* хранилась в базе данных политик и состояла из информации о привилегиях пользователей и конфигурации аудита. Она реплицировалась между контроллерами домена, поэтому все контроллеры получали одинаковые настройки аудита и привилегий. Политика домена действовала в отношении всего домена, но не могла быть обшей для нескольких доменов.

Реализация политик безопасности в Windows 2000 предоставляет значительно более широкие возможности. В случае необходимости вы можете устанавливать политики для всего дерева доменов. Различные контроллеры в пределах одного домена могут обладать индивидуальными политиками безопасности. Установив политику безопасности в одном

месте, администраторы могут контролировать безопасность всех серверов и рабочих станций домена.

Групповая политика имеет следующие преимущества:

- Основываясь на службе Active Directory системы Windows 2000, позволяет выполнять как централизованное, так и децентрализованное управление параметров политики.
- Обладает гибкостью и масштабируемостью. Применяется в широком наборе конфигураций системы, предназначенных как для малого бизнеса, так и для больших корпораций.
- Дает возможность использовать простой интегрированный инструмент управления политикой. Редактор групповой политики представляет собой оснастку консоли управления Microsoft, расширяющую такие инструменты администрирования, как, например, Active Directory Manager и Cотpl1ter Managemel1t. Администраторы могут делегировать право управления объектами групповой политики.
- Редактор групповой политики обладает простым и хорошо понятным интерфейсом.
- Обладает высокой степенью надежности и безопасности.

Групповые политики расширяют и используют преимущества Active Directory. Их настройки находятся в объектах групповых политик, которые в свою очередь ассоциируются с такими контейнерами Active Directory, как области, домены и организационные единицы.

Политики безопасности Windows 2000 хранятся в двух типах объектов политик каталога: *объекты локальной "политики безопасности* и *объекты политики безопасности домена.*

В объекте политики безопасности, определяющем безопасность домена или организационной единицы, хранится следующая информация:

- **Политика пароля** - устанавливает политику паролей для учетных записей домена.
- **Политика блокировки учетных записей** - включает или отключает политику блокировки учетных записей.
- **Дескриптор безопасности для операций масштаба домена** – устанавливает безопасность для определенного объекта политики домена. Определяет, кто может изменять настройки свойств объекта политики безопасности домена.
- **Политика регистрации Kerberos** (обновление билета).
- **Политика восстановления данных EFS.**
- **Ссылка на назначенный объект политики безопасности компьютера**

В домене могут быть созданы свои собственные объекты политик безопасности или храниться реплика и ссылка на политику безопасности другого домена в дереве. Если политика реплицируется из другого домена, репликацией должен управлять сервер

глобального каталога. Ссылка на объект политики безопасности домена хранится в объекте "домен".

В объекте локальной политики безопасности хранится следующая информация:

- **Политика аудита** - включает и отключает локальный аудит. Устанавливает, какие события должны быть подвергнуты аудиту.
- **Политика привилегий** - определяет, какие права и разрешения получают учетные записи.
- **Локальные роли** - определяет, какие роли будут играть такие пользователи и группы, как Administrators, Account Operators или Print Operators.
- **Разрешения и настройки аудита** для объекта политики безопасности компьютера. Устанавливают безопасность для объекта политики безопасности компьютера, выбираемой по умолчанию. Эти настройки определяют, кто может изменять свойства объекта безопасности компьютера.

Отдельно стоящая машина или машина, работающая в составе рабочей группы, получает локальную политику безопасности. Компьютер, присоединенный к домену, получает сначала локальную политику, а затем политику, определенную в масштабах домена. Это значит, что политика безопасности масштаба домена имеет приоритет над локальной политикой безопасности. Каждая политика безопасности представляет собой *объект групповой политики* (Group Policy Object, GPO). В основном GPO используются для определения политики безопасности домена. Однако на каждом компьютере находится один GPO, необходимый для задания локальной политики безопасности.

Объекты групповой политики, используемые для формирования безопасности домена, объединяются в *скелет групповой политики* (Group Policy Framework, GPF). Сюда включаются следующие политики безопасности:

- **Политики учетных записей** *(Account* Policies). Позволяют настраивать политики безопасности учетных записей (как в масштабах домена, так и локальных учетных записей). Здесь определяются политика паролей, политика блокировки паролей и новая политика Kerberos, распространяющаяся на весь домен.
- **Локальные политики** (Local Policies). Позволяют настраивать политику аудита, назначать права пользователей и различные параметры безопасности, которые могут быть настроены в системе Windows 2000.
- **Ограниченное членство в группах** (Restricted *Groups).* Такие политики позволяют настраивать членство в специфических группах. Сюда обычно включают встроенные группы, такие как Administrators, Backup Operators 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** (System Services). Позволяют 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. Например, расширение File File Sharing Service позволяет настраивать политику безопасности для службы создания общего доступа к файлу (активизация подписи SMB, ограничение анонимного доступа к общим ресурсам, формирование безопасности различных сетевых общих ресурсов и т.д.).
- **Реестр** (Registry). Позволяет .настраивать безопасность различных ключей. реестра.
- **Файловая система** (File System). Позволяет настраивать безопасность определенных файлов.
- **Политики открытых ключей (**Public Key Policies). Позволяют настраивать политики безопасности в отношении шифрования информации с помощью EFS, авторизации корневого сертификата в масштабах домена, авторизации доверенного сертификата и т.д.
- **Политика безопасности IP** (IPSEC). Позволяет настраивать политику безопасности **IP** для компьютеров, находящихся в определенной области действия.
- **Политики безопасности, определяемые GPO из структуры групповой политики.** Действуют на компьютеры и частично на пользователей. Все политики, о которых говорилось выше, действуют только на компьютеры. Даже политики учетных записей распространяются на весь компьютер, хотя они влияют на учетные записи.

Поскольку политика безопасности Windows 2000 Server значительно отличается от политик предыдущих версий Windows NT, при переходе к Windows 2000 низкоуровневые политики безопасности не переносятся. Если при переходе создается новое дерево доменов, одновременно создается и новая политика безопасности, назначаемая по умолчанию. Если при переходе домен присоединяется к уже существующему дереву, политика безопасности берется от родительского домена.

Отличие локальных учетных записей от доменных.

В каждом компьютере Windows 2000, являющемся клиентом или отдельно стоящей машиной, создаются локальные учетные записи и группы. Они отличаются от аналогичных сетевых объектов, поддерживаемых контроллерами доменов Windows NT. Отличие заключается в том, что учетная запись, созданная в сетевом домене, позволяет регистрироваться в сети с помощью любого компьютера, присоединенного к домену. С другой стороны, локальные учетные записи могут быть использованы для регистрации

только на том компьютере, где они находятся. Большинство политик безопасности учетных записей (например, политика паролей) действует на все учетные записи, включая и локальные. Исключение составляют политики безопасности Kerberos. Они доступны только в контексте домена Windows NT.

Локальные политики всегда действуют на индивидуальные машины, независимо от того, является данная машина контроллером домена или нет. Они могут быть установлены независимо от политик самого домена.

Инструменты настройки безопасности.

Обеспечение эффективной безопасности системы имеет несколько аспектов. Во-первых, операционная система должна соответствовать базовым требованиям к безопасности. Например, требования к системе, соответствующей уровню безопасности C2, включают защиту памяти, дискреционное управление доступом и наличие средств аудита.

Во-вторых, для создания эффективной безопасности следует создать инструмент, позволяющий управлять всеми средствами обеспечения безопасности, заложенными в операционной системе. В Windows 2000 для управления безопасностью системы при меняется Security Configuration Tool Set (Набор инструментов настройки безопасности). В него включены параметры безопасности операционной системы, собранные в единый блок управления, и ряд программных инструментов, позволяющих управлять этими параметрами.

Набор инструментов настройки безопасности состоит из следующих компонентов:

- *Security Configuration Service* (Служба настройки безопасности - реализована в виде оснастки MMC) - служба, являющаяся ядром Security Configuration Tool Set. Она работает на любой машине и отвечает за выполнение функций, связанных с настройкой безопасности и ее анализом. Эта служба является центральной для всей инфраструктуры безопасности системы.
- *Setup Security* (Безопасность установки) - первоначальная конфигурация безопасности, создаваемая при установке Windows 2000 с помощью заранее определенного шаблона, поставляемого вместе с системой. На каждом компьютере Windows 2000 формируется первоначальная база данных безопасности, называемая "Локальная политика компьютера" (Local Computer Policy).
- *Security Configuration Editor* (Редактор настройки безопасности) - этот инструмент позволяет определять конфигурации безопасности, не зависящие от машины, которые хранятся в виде текстовых файлов.

- *Security Configuration Manager* (Диспетчер настройки безопасности) - этот инструмент позволяет импортировать одну или несколько хранящихся конфигураций безопасности в базу данных безопасности (это может быть база данных Локальной политики компьютера или любая другая личная база). Импорт конфигураций создает специфическую для машины базу данных безопасности, которая хранит композитную настройку. Ее можно активизировать на компьютере и проанализировать состояние текущей конфигурации безопасности по отношению к композитной настройке, хранящейся в базе данных.
- *Security Settings Extension to Group Policy Editor* (Расширение Редактора групповой политики) - эта программа расширяет Редактор групповой политики и позволяет определять конфигурацию безопасности как часть объекта групповой политики.

2. Безопасность одиночного компьютера.

Управление локальными политиками безопасности.

В операционной системе Windows 2000 реализована новая концепция управления политиками безопасности компьютера. Каждый компьютер является элементом безопасности, обладающим правами доступа и локальной политикой безопасности. Одновременно с доменом создается и объект политики безопасности компьютера, который впоследствии может быть отредактирован. Кроме того, вы можете создать другой объект безопасности компьютера, имеющий необходимые свойства. Политика безопасности может быть создана для каждой функциональной категории компьютеров (например, для контроллеров домена, серверов приложений, серверов печати и др.). Свой объект политики могут иметь отдельные компьютеры домена. Каждый компьютер может иметь только *одну политику безопасности,* "ассоциированную с ним в данный момент времени. В Windows 2000 управление политикой безопасности компьютеров домена дает администраторам возможность лучше контролировать компьютеры, входящие в состав домена. Отдельно стоящие компьютеры и машины, находящиеся в составе рабочих групп, обладают локальным *объектом групповой политики (*Group Policy Object, GPO), определяющим их локальную политику. Он может быть отредактирован аналогично любому другому GPO каталога. Работа политик безопасности компьютера подчиняется следующим принципам:

- При включении компьютера в домен по умолчанию используется политика безопасности компьютера, определенная локально. Однако администратор домена имеет возможность изменить ее и заставить все компьютеры домена использовать политику безопасности компьютера, сконфигурированную в масштабе домена. Это позволяет получить больший контроль над политиками безопасности, используемыми различными машинами, входящих в состав домена.
- Если администратор хочет, чтобы компьютер, входящий в состав домена, использовал политику безопасности, сконфигурированную в масштабе домена, он должен создать в домене объект "компьютер".

Для того чтобы компьютер использовал политику безопасности компьютера, сконфигурированную в масштабе всего домена необходимо:

- Запустите оснастку Active Directory Manager.
- Установите указатель мыши на имени компьютера, который должен использовать политику масштаба домена, и нажмите правую кнопку. В появившемся меню выберите команду **Properties.**
- В окне свойств компьютера перейдите на вкладку **General** (Общие) и просмотрите информацию в поле **Computer security policy** (Политика

безопасности компьютера). По умолчанию должно быть установлено значение **locally defined security policy** (Политика безопасности, определенная локально).

- Для выбора другой политики безопасности нажмите кнопку **Change** (Изменить).
- Появится окно диалога **Change Computer Security Policy** (Изменение политики безопасности компьютера). Возможен один из следующих вариантов:
  - Использовать политику безопасности домена, установленную по умолчанию.
  - Использовать другую политику безопасности, существующую в данном домене. Для выбора нужного объекта политики безопасности компьютера нажмите кнопку **Browse** (Обзор). Этот вариант следует применять, если вам необходимо, чтобы определенный набор компьютеров домена использовал одинаковые политики безопасности, которые отличаются от политики безопасности домена, установленной по умолчанию. Например, вы можете сконфигурировать все контроллеры домена с политикой безопасности, которая будет отличаться от политики безопасности рабочих станций, входящих в состав домена.
  - Использовать локальную политику безопасности. В этом случае выполняется эмуляция работы политики безопасности предыдущих версий Windows NT. Локальные администраторы компьютеров смогут самостоятельно определять и редактировать политики безопасности компьютеров. Этот вариант устанавливается в компьютерах по умолчанию.
  - Нажмите кнопку ОК**.**

В следующих разделах приведены некоторые примеры применения политик безопасности компьютеров.

Блокирование локальных учетных записей.

После нескольких неудачных попыток регистрации в системе учетная запись пользователя должна быть заблокирована. Вы можете установить допустимое максимальное количество неудачных попыток. Следует заметить, что разблокировать учетную запись может только администратор.

Для модификации локальной политики учетных записей:

1. Установите указатель мыши на значке My Computer и нажмите правую кнопку. В появившемся контекстном меню выберите команду Manage (Управление) . Запустится оснастка Computer Management.
2. В разделе System Tools (Системные инструменты) выберите Group Policy Editor (GPE).
3. В GPE выберите раздел Security Settings (Настройки безопасности), в нем выберите Account Policies (Политики учетных записей) и внутри этого раздела выберите Account Lockout Policy (Политика блокировки пароля). Появится окно, показанное на рис. 17.2, в котором будет отображен набор настроек политики блокировки.
4. Для запуска средства редактирования сделайте двойной щелчок на разделе, настройку которого вы хотите изменить, например, Account lockout count.
5. В появившемся окне введите с клавиатуры новое значение и снимите флажок Exclude this setting from configuration (Исключить эту настройку из конфигурации).
6. Нажмите кнопку ОК. в результате будет установлено новое значение параметра.
7. Сделайте двойной щелчок на разделе Lockout account for (Блокировать учетную запись для). В появившемся окне установите длительность блокировки учетной записи (от 0 до бесконечности (Forever». Нажмите кнопку ОК. (Как и в предыдущем случае следует снять флажок Exclude this setting from configuration.).
8. Для сохранения сделанных изменений завершите работу программы Computer Management.

Настройка безопасности для ключа реестра.

Ниже показано, как можно настроить безопасность для ключа реестра HKEY_LOCAL_MACHINE\Software\Microsoft\ Windows NT Администраторы получат полный контроль, а все остальные аутентифицированные пользователи будут иметь доступ только на чтение.

Для настройки безопасности для ключа реестра:

- Запустите оснастку Compiter Management и выберите Group Policy Editor.
- В GPE укажите мышью на раздел Security Settings, выберите папку Registry (Реестр). В правом подокне будут отображены ключи реестра, на которые распространяется политика безопасности, установленная по умолчанию.
- Установите указатель мыши на папке **Registry** и нажмите правую кнопку. В появившемся меню выберите команду **Add Key** (Добавить ключ). Запустится браузер реестра.
- Перейдите к **Machine\Software\Microsoft\ Windows NT** и выберите **Windows NT**. (В качестве альтернативы можно ввести полный путь к редактируемому полю.).
- Нажмите кнопку **ОК.** Появится окно диалога, с помощью которого можно настроить безопасность выбранного ключа.

- Для изменения списка объектов, имеющих разрешение доступа к данному ключу, следует использовать кнопки Add (Добавить) и Remove (Удалить). Здесь вы сможете задать полный доступ только для администраторов и доступ только на чтение для остальных пользователей. После завершения корректировки данных в окне Configuration Security for (Конфигурация безопасности для) нажмите кнопку ОК.
- При настройке безопасности ключа реестра можно задать три типа действия безопасности:
    - Наследовать (Inberit) - безопасность, установленная для данного ключа реестра, будет автоматически наследоваться ниже по дереву. Это значит, что все напрямую заданные права доступа к подключам будут сохранены. К ним будут добавлены унаследованные разрешения доступа.
    - Перезаписать (Overwrite) - безопасность, установленная для данного ключа реестра, будет также автоматически наследоваться ниже по дереву. Но в данном случае любые напрямую заданные права доступа к подключам будут перезаписаны новыми установками. Действовать будет только вновь созданная безопасность.
    - Игнорировать (Ignore) - это исключение из предыдущего случая. Например, для родительского ключа установлена новая безопасность. Она наследуется всеми подключами и перезаписывает все остальные прямые установки, но один из подключей может быть установлен таким образом, что настройки безопасности родительского ключа будут игнорироваться. В этом случае на подключ и дерево под ним не будут распространяться настройки безопасности родительского ключа.
    - Закройте программы настройки. Все сделанные вами изменения в настройках будут сохранены и начнут работать в локальной политике безопасности.

Настройка безопасности для диска.

С помощью политики безопасности вы сможете установить различные права доступа к устройству для разных пользователей (например, полный контроль для администраторов и права на чтение и создание подкаталогов, а также полный контроль над создаваемыми файлами).

Для настройки безопасности устройства:

1. Запустите оснастку Computer Management и выберите Group Policy Editor.
2. В GPE укажите мышью на раздел Security Settings, выберите. папку FileSystem (Файловая система).

3. Установите указатель мыши на папке File System, щелкните правой кнопкой и выберите команду Add Folder (Добавить папку). Запустится браузер файловой системы.
4. В окне диалога Browse for Folder (Обзор папки) выберите Local Disk (C:) и нажмите кнопку ОК.
5. В появившемся окне диалога File and Registry Object Configuration (Конфигурация объекта файловой системы или реестра) выберите тип действия безопасности и нажмите кнопку Edit Security (Редактировать безопасность).
6. В появившемся окне диалога Access Control Settings for (Настройка управления доступом для) с помощью кнопки Remove можно очистить список пользователей. С помощью кнопки Add можно добавить в поле Permissions (Разрешения) следующие объекты:
    - *Администраторы и система* получают полный доступ, который распространяется на все подкаталоги.
    - *Аутентифицированные пользователи* получают права Traverse Folder (Проходить сквозь папку), List Folder (Просматривать папку), Read Attributes (Читать атрибуты), Read Extended Attributes (Читать расширенные атрибуты), Read Permissions (Читать права доступа) и Create Files and Folders (Создавать файлы и папки). Они должны распространяться только на данную папку. Эта настройка не должна наследоваться.
    - *Создатель/владелец* получает полный контроль. Это позволит пользователям иметь полный доступ к тем объектам файловой системы, которые они создали, и ни к чему больше. В соответствии с предыдущей настройкой пользователи могут создавать подкаталоги, теперь они получают над ними полный контроль.

7. Окно диалога, с помощью которого можно настраивать безопасность папки, дает возможность задать следующие типы действия безопасности: наследовать, перезаписать и игнорировать.

Шифрующая файловая система EFS.

Для обеспечения безопасности данных на персональных компьютерах рекомендуется загружать операционную систему не с жесткого, а с гибкого диска. Это позволит избежать отказов

в работе жесткого диска и разрушения загрузочных разделов. К сожалению, с помощью гибкого диска можно загрузить несколько различных операционных систем, поэтому любой пользователь, получивший физический доступ к компьютеру, может обойти встроенную систему управления доступом файловой системы Windows 2000 (NTFS) и с помощью определенных инструментов' прочесть информацию жесткого диска. Многие конфигурации оборудования позволяют применять пароли, регулирующие доступ при загрузке. Однако такие средства не имеют широкого распространения. Кроме того, в случае, если на компьютере работает несколько пользователей, подобный подход не дает хороших результатов, да и сама защита с помощью пароля недостаточно надежна. Ниже рассмотрены типичные примеры того, как может быть осуществлен несанкционированный доступ к данным:

- *Хищение компьютера типа Laptop.* Любой злоумышленник может незаметно и быстро похитить компьютер типа Laptop, а затем получить доступ к конфиденциальной информации, находящейся на его жестком диске.
- *Неограниченный доступ.* Компьютер оставлен в рабочем состоянии и за ним никто не наблюдает. Любой пользователь может подойти к такому компьютеру и получить доступ к конфиденциальной информации.
- Основной целью создания системы безопасности является зашита конфиденциальной информации, которая обычно находится в незащищенных файлах на жестком диске, от несанкционированного доступа. Доступ к данным можно ограничить с помощью средств NTFS. Такой подход обеспечивает хорошую степень защиты, если единственной загружаемой операционной системой является Windows 2000, жесткий диск не может быть физически удален из компьютера, и данные находятся в разделе NTFS. Если кто-либо захочет получить доступ к данным, он может осуществить свое желание, получив физический доступ к компьютеру или жесткому диску. Существуют *инструменты, позволяющие получить доступ к файлам, находящимся в разделе NTFS,* из операционных систем MS-DOS или UNIX в обход системы безопасности NTFS.

Из приведенных выше соображений следует вывод: единственно надежный способ зашиты информации - это шифрующая файловая система. На рынке программного обеспечения существует целый набор продуктов, обеспечивающих шифрование данных с помощью образованного от пароля ключа на уровне приложений. Однако такой подход имеет ряд ограничений:

- **Ручное шифрование и расшифровывание.** Службы шифрования большинства продуктов не прозрачны для пользователей. Пользователю приходится расшифровывать файл перед каждым его использованием, а затем опять зашифровывать. Если пользователь забывает зашифровать файл после окончания работы с ним, информация остается незащищенной. Поскольку каждый раз необходимо указывать, какой файл должен быть зашифрован (и расшифрован), применение такого метода защиты информации сильно затруднено.

- **Утечка информации из временных файлов и файлов подкачки.** Практически все приложения в процессе редактирования документов создают временные файлы. Они остаются на диске незашифрованными несмотря на то, что оригинальный файл - зашифрован. Кроме того, шифрование информации на уровне приложений выполняется в режиме пользователя Windows 2000. Это значит, что ключ, применяющийся для такого типа шифрования, может храниться в файле подкачки. В результате, с помощью изучения данных файла подкачки можно получить ключ и расшифровать все документы пользователя.
- **Слабая криптостойкость ключей.** Ключи образуются от паролей или случайных фраз. Поэтому в случае, если пароль был легко запоминаемым, атаки с помощью словарей могут легко привести к взлому системы защиты.
- **Невозможность восстановления данных.** Большинство продуктов, позволяющих шифровать информацию, не предоставляют средств восстановления данных, что является дополнительным поводом для пользователей не применять средства шифрования. Это особенно касается тех работников, которые не хотят запоминать дополнительный пароль. С другой стороны, средство восстановления данных с помощью пароля, представляет собой еще одну брешь в системе защиты информации. Все, что необходимо злоумышленнику, - это пароль, предназначенный для запуска механизма восстановления данных, который позволит получить доступ к зашифрованным файлам.

Все перечисленные выше проблемы позволяет решить *Шифрующая файловая система* (Encrypting File System, EFS), реализованная в Windows 2000 и работающая только на NTFS 5.0. Следующие разделы содержат детальное описание технологии шифрования, места шифрования в операционной системе, взаимодействия с пользователями и способа восстановления данных.

Технология шифрования EFS.

Система EFS основана на шифровании с открытым ключом и использует все возможности архитектуры CryptoAPI в Windows 2000. Каждый файл шифруется с помощью случайно сгенерированного ключа, зависящего от пары открытого (public) и закрытого (secret) ключей пользователя. Подобный подход в значительной степени затрудняет осуществление большого набора атак, основанных на *криптоанализе.* При криптозащите файлов может быть использован любой алгоритм симметричного шифрования. Текущая версия EFS использует алгоритм DES. В дальнейшем, вероятно, появится возможность применения альтернативных алгоритмов. EFS позволяет осуществлять шифрование и дешифрование файлов, находящихся на удаленных файловых серверах. (в данном случае EFS может работать только с файлами, находящимися на диске. Шифрующая файловая система не осуществляет криптозащиту данных, передаваемых по сети. Для шифрования передаваемой информации в операционной системе Windows 2000 следует использовать специальные сетевые протоколы, например SSL/PCT.

Система EFS хорошо интегрирована в NTFS. При создании временных файлов они наследуют атрибуты оригинальных файлов (если файлы находятся в разделе NTFS). При

шифровании файла шифруются также и его временные копии. EFS находится в ядре Windows 2000 и использует для хранения ключей специальный пул, невыгружаемый на жесткий диск. Поэтому ключи никогда не попадают в файл подкачки.

Взаимодействие с пользователем.

Конфигурация EFS, устанавливаемая по умолчанию, позволяет пользователю шифровать свои файлы без всякого вмешательства со стороны администратора. В этом случае EFS автоматически генерирует для пользователя пару открытого и закрытого ключей, применяемую для криптозащиты данных.

Шифрование и дешифрование файлов может быть выполнено как для определенных файлов, так и для целого каталога. Криптозащита каталога прозрачна для пользователя. Все файлы и подкаталоги, созданные в зашифровываемом каталоге, шифруются автоматически. Каждый файл обладает уникальным ключом, позволяющим легко выполнять операцию переименования. Если вы переименовываете файл, находящийся в зашифрованном каталоге, и переносите его в незашифрованный каталог, сам файл остается зашифрованным. Службы шифрования и дешифрования доступны через Windows NT Explorer. Кроме того, можно полностью использовать все доступные средства криптозащиты данных с помощью набора утилит командной строки и интерфейсов администрирования.

Система EFS исключает необходимость предварительного расшифровывания данных при доступе к ним. Операции шифрования и дешифрования выполняются автоматически при записи или считывании информации. EFS автоматически распознает зашифрованный файл и найдет соответствующий ключ пользователя в системном хранилище ключей. Поскольку *механизм хранения ключей* основан на использовании CryptoAPI, пользователи получают возможность хранить ключи на защищенных устройствах, например смарткартах.

Текущая версия EFS не позволяет создавать общие ресурсы. Однако ее архитектура предполагает возможность создания общих ресурсов, к которым пользователи могут обращаться с помощью их общих ключей. Впоследствии каждый пользователь может независимо расшифровать файл с использованием его закрытого ключа. Пользователи могут быть легко добавлены (если они имеют сконфигурированную пару ключей) или удалены из группы, имеющей разрешение доступа к общему ресурсу.

Система EFS располагает встроенными средствами восстановления данных. Инфраструктура системы безопасности Windows 2000 позволяет создать необходимые для этого ключи. Шифрование файлов может быть использовано только в случае, если в системе создан один или несколько ключей восстановления информации. EFS позволяет *агентам восстановления данных* создавать открытые ключи, которые затем используются при восстановлении файлов. Используя ключ восстановления, можно получить только сгенерированный случайным образом ключ, с помощью которого был зашифрован конкретный файл. Поэтому агенту восстановления не может случайно стать доступной другая конфиденциальная информация.

Средство восстановления данных предназначено для использования в разнообразных конфигурациях вычислительных сред. Оно применяется в случае, например, когда организации

необходимо получить доступ к информации, зашифрованной уволенным пользователем, или когда ключ, с помощью которого данные были зашифрованы, утерян. Политика восстановления может быть определена на контроллере домена Windows 2000. В этом случае она распространяется на все компьютеры домена и находится под контролем администраторов домена, которые с помощью службы Active Directory могут делегировать право управления политикой восстановления файлов учетным записям администраторов безопасности. Это позволяет получить возможность следить за кругом лиц, имеющих право восстанавливать зашифрованные файлы, а также гибко предоставлять пользователям это право или лишать его. Кроме того, с помощью создания нескольких конфигураций ключей восстановления EFS позволяет одновременно применять несколько агентов, что делает процедуры восстановления значительно более устойчивыми и гибкими.

Система EFS может быть использована и в домашних условиях. Она автоматически генерирует ключи восстановления в отсутствие домена Windows 2000 и хранит их как ключи машины. Для восстановления данных в домашних условиях пользователи могут использовать утилиты командной строки и учетную запись администратора.

Конфигурация ключей пользователей.

Пользователи могут генерировать, экспортировать, импортировать открытые ключи, используемые в EFS для шифрования, а также управлять ими. Конфигурация интегрирована с другими настройками безопасности, устанавливаемыми пользователем. Этот пункт предназначен для опытных пользователей, которые хотят иметь средство управления собственными ключами. Обычно пользователям не приходится самостоятельно выполнять конфигурацию, поскольку EFS автоматически генерирует ключи шифрования файлов.

Утилита cipher.

Эта утилита командной строки позволяет шифровать и расшифровывать файлы. Описание ее ключей приведено ниже:

/E - Шифрует указанные в качестве параметра файлы. Каталоги помечаются как зашифрованные, все файлы, которые будут помещены в них впоследствии, шифруются автоматически.

/D - Расшифровывает все указанные после ключа файлы. Каталоги помечаются как незашифрованные, все файлы, которые будут помещены в них впоследствии, шифроваться не будут.

/R - Расшифровывает файлы с помощью информации восстановления.

/S - Выполняет определенную ключом операцию над файлами и всеми подкаталогами, находящимися в данном каталоге.

/I - Продолжает выполнение указанной операции даже после возникновения ошибочной ситуации. По умолчанию при появлении ошибки программа cipher останавливается.

/F - Осуществляет принудительное шифрование всех файлов, указанных после ключа, даже если они уже зашифрованы. По умолчанию уже зашифрованные файлы не подвергаются вторичному шифрованию.

/Q - Выдает только краткую информацию.

/P - Устанавливает политику восстановления для данной машины или домена. Если существует файл ключа, cipher использует его содержимое для установки политики. Если файл ключа отсутствует, политика восстановления генерируется автоматически, а результат записывается в файл ключа/

/K - Сохраняет ключи EFS пользователя в заданном файле ключа.

/L - Загружает ключи, находящиеся в заданном файле ключа, в текущий контекст.

Параметр *filename* может быть маской, файлом или каталогом. При вводе команды cipher без параметров, она выдает информацию о том, зашифрован ли данный каталог или файлы, находящиеся в нем. Если параметр *filename* присутствует, то имен файлов может быть и несколько. Между собой параметры должны быть разделены пробелом.

Для того чтобы зашифровать каталог My *Documents* введите команду: c:\cipher /E "My Documents"

Для того чтобы зашифровать все файлы, имена которых содержат набор символов "cndfl", введите команду: c:\cipher /E /S *cndfl*

Команда *copy* Команда copy расширена. Она получила дополнительные ключи, позволяющие импортировать и экспортировать зашифрованные файлы в промежуточный формат. Синтаксис команды copy имеет следующий вид:

copy [/E 1/1] *sourcefile destinationfile*

Новые ключи команды Copy:

/E - Экспортирует зашифрованный файл *(sourcefile)* в виде шифрованного потока битов в файл *destinationfile.* Файл *destinationfile* может не быть томом NTFS, но может быть файлом FAT на гибком диске.

/I - Импортирует шифрованный поток битов из файла *sourcefile* в файл, зашифрованный с помощью EFS на томе NTFS. Файл *sourcefile* может ненаходиться на томе NTFS, но файл *destinationfile* должен находиться на томе NTFS.

(Перечисленные ключи были доступны в Windows 2000 Beta 1)

Шифрование файлов.

Чтобы зашифровать файлы необходимо выбрать их в Windows NT Explorer, а затем установить на выбранном файле указатель мыши и нажать правую кнопку. Появится контекстное меню, в котором нужно выбрать команду **Encrypt** (Шифровать) и EFS зашифрует выбранные файлы.

После завершения шифрования файлы сохраняются на диске. Теперь при выполнении операций чтения и записи файлы будут расшифровываться и зашифровываться автоматически. Для того, чтобы узнать, зашифрован ли файл, пользователь может просмотреть свойства файла и выяснить установлен ли флаг Encrypted.

Поскольку шифрование выполняется автоматически, пользователь может работать с файлом так же, как и до установки его криптозащиты. редактировать его, как и прежде. Все остальные пользователи, которые попытаются получить доступ к зашифрованному файлу, получат сообщение об ошибке доступа, поскольку они не владеют необходимым секретным ключом, позволяющим им расшифровать файл. Заново данные шифруются только в случае, если файл копируется в зашифрованный каталог.

С помощью контекстного меню Explorer пользователи могут шифровать не только файлы, но и целые каталоги. Если сделать каталог зашифрованным, это значит, что все находящиеся в нем или появившиеся впоследствии файлы и подкаталоги, будут подвергаться шифрованию по умолчанию. Список файлов каталога не шифруется, поэтому при наличии соответствующих разрешений доступа он может быть просмотрен обычным способом.

Операция шифрования каталогов аналогична шифрованию файлов - просто выделите определенный каталог, а затем выберите в контекстном меню Windows NT Explorer команду **Encrypt.** В этом случае каталог помечается как зашифрованный, а все файлы и подкаталоги, находящиеся в нем шифруются. Создавая зашифрованные каталоги и копируя туда файлы, содержащие конфиденциальную информацию, пользователи могут осуществлять криптозащиту данных.

Дешифрование файлов и каталогов.

При выполнении обычных операций пользователю не нужно заботиться о дешифровании файлов, поскольку EFS автоматически шифрует и дешифрует данные при их записи и чтении. Однако при организации общего доступа к файлам возникает необходимость принудительного дешифрования данных.

Дешифровать файл или каталог можно с помощью контекстного меню Windows NT Explorer. Эта операция практически идентична операции шифрования, за исключением выбираемой команды - **Decrypt** (Дешифровать) вместо **Encrypt.** В результате выполнения этой операции EFS дешифрует все выбранные файлы и пометит их незашифрованными. В случае снятия криптозащиты с каталога контекстное меню позволяет рекурсивно дешифровать все находящиеся в каталоге файлы и подкаталоги.

Операция восстановления.

По умолчанию на всех компьютерах, принадлежащих к домену, установлена *локальная система безопасности.* В этом случае пользователи могут сразу же работать с EFS. Однако, если несколько компьютеров объединены в домен, на них устанавливается *система безопасности домена* (индивидуальная или выбираемая по умолчанию). В домене пользователи не смогут применять EFS до тех пор, пока не будет установлена политика восстановления, которая представляет собой элемент либо политики безопасности домена Windows 2000, либо политики безопасности локального компьютера для отдельно стоящего сервера или рабочей станции. В масштабе домена, политика восстановления распространяется на все компьютеры домена. Интерфейс пользователя политики EFS входит в состав общего интерфейса настройки политики домена или локальной политики. Он дает возможность пользователю при помощи средств управления ключами генерировать, экспортировать, импортировать и архивировать *ключи*

восстановления. Интеграция политики восстановления в общую политику безопасности позволяет создать последовательную и хорошо понятную модель обеспечения защиты информации. Подсистема безопасности Windows 2000 обеспечивает настройку, репликацию и кэширование политики EFS. Поэтому пользователи имеют возможность использовать шифрование на компьютере, временно отключенном от общей вычислительной системы, например, на портативном компьютере, а также могут регистрироваться в системе с помощью своей учетной записи, используя кэшированную информацию входа.

Восстановить файл, секретная часть ключа которого утеряна, можно с помощью утилиты командной строки cipher:

- Если файл находится на компьютере домена, то в профиль агента восстановления необходимо загрузить ключи восстановления. Для этого с клавиатуры введите следующую команду: cipher /1: <*.имя файла ПОЛИТИКИ восстановления, устанавливаемой* в *домене*>
- Если файл находится на отдельно стоящем компьютере, то для расшифрования файла с помощью ключей восстановления, загруженных в профиль агента восстановления, необходимо ввести команду: cipher /d <Путь к восстанавливаемому файлу>

Восстановление данных, зашифрованных с помощью неизвестного закрытого ключа.

Необходимо, чтобы политика восстановления была установлена на уровне домена (или локально, если машина не является членом домена) *до* использования EFS. Политику восстановления устанавливают администраторы домена или пользователи *(агенты восстановления,* которым делегированы соответствующие права), контролирующие ключи восстановления всех машин домена.

Если пользователь забыл закрытый ключ, файл, зашифрованный с помощью такого ключа, может быть восстановлен после его экспортирования и пересылки по электронной почте одному из агентов восстановления, который импортирует файл на защищенную машину, где находится закрытый ключ восстановления. С помощью полученного ключа и с использованием соответствующей утилиты командной строки файл может быть дешифрован и возвращен пользователю. В небольших вычислительных системах или домашних условиях, где домен отсутствует, восстановление может быть выполнено на самом отдельно стоящем компьютере.

Архитектура EFS.

Принцип шифрования.

В EFS шифрование и дешифрация информации выполняется с помощью алгоритма открытого ключа. Данные зашифровываются с помощью симметричного алгоритма с применением ключа шифрования файла (File Encryption Key, FEK). FEF - это сгенерированный случайным образом ключ, имеющий определенную длину.

В свою очередь, FEK шифруется с помощью одного или нескольких общих ключей, предназначенных для криптозащиты ключа. Таким образом, получают список зашифрованных FEK, что позволяет организовать доступ к файлу со стороны нескольких пользователей. Для шифрования набора FEK используется общая часть пары ключей пользователя. Список зашифрованных FEK хранится вместе с зашифрованным файлом в специальном атрибуте EFS, называемом *поле дешифрования данных* (Data Decryption Field, DDF). Информация, требуемая для дешифрования, привязывается к самому файлу. Секретная часть ключа пользователя используется при дешифровании FEK, Она хранится в безопасном месте, например на смарт-карте или другом устройстве, обладающем высокой степенью защищенности. (Шифрование с использованием ключа пользователя может быть выполнено с помощью симметричного алгоритма, применяющего ключ, образованный из пароля. EFS не поддерживает этот подход, поскольку схема, основанная на пароле пользователя, не обладает необходимой устойчивостью к атакам с применением словарей.)

Ключ FEK применяется для создания ключей восстановления. Для этого FEK шифруется с помощью одного или нескольких общих ключей восстановления. Для шифрования набора FEK используется общая часть каждой пары ключей. Список PEK, зашифрованных для целей восстановления, хранится

вместе с зашифрованным файлом в специальном атрибуте EFS, называемом *поле восстановления данных* (Data Recovery Field, DRF). Благодаря существованию набора зашифрованных PEK файл может восстановить несколько агентов. Для шифрования FEK в DRF необходима только общая часть пары ключей восстановления, ее присутствие в системе необходимо в любой момент времени для нормального функционирования файловой системы. Сама процедура восстановления выполняется довольно редко, когда пользователь увольняется из организации или забывает секретную часть ключа. Поэтому агенты восстановления могут хранить секретную часть ключей восстановления в безопасном месте, например на смарт-картах или других устройствах, обладающих высокой степенью защищенности.

Шифрование данных производится по следующим этапам:

- Незашифрованный файл пользователя шифруется с помощью сгенерированного случайным образом FEK.
- FEK шифруется с помощью общей части пары ключей пользователя и помещается в DDF.
- FEK шифруется с помощью общей части ключа восстановления и помещается в DRF.

Дешифрование данных производится по следующим этапам:

- Из DDF извлекается зашифрованный FEK и дешифруется с помощью секретной части ключа пользователя.
- Зашифрованный файл пользователя дешифруется с помощью FEK, полученного на предыдущем этапе. При работе с большими файлами дешифруются только отдельные блоки, что значительно ускоряет выполнение операций чтения.

Процесс восстановления файла после утраты секретной части ключа.

Восстановление данных производится:

- Из DDF извлекается зашифрованный FEK и дешифруется с помощью секретной части ключа восстановления.
- Зашифрованный файл пользователя дешифруется с помощью FEK, полученного на предыдущем этапе.

Описанная выше общая система криптозащиты позволяет применять максимально надежную технологию шифрования и дает возможность многим пользователям и агентам восстановления получать общий доступ к зашифрованным файлам. Она полностью независима от применяемого алгоритма шифрования, что очень важно, поскольку позволит в будущем легко перейти на новые, более эффективные алгоритмы.

Структура EFS.

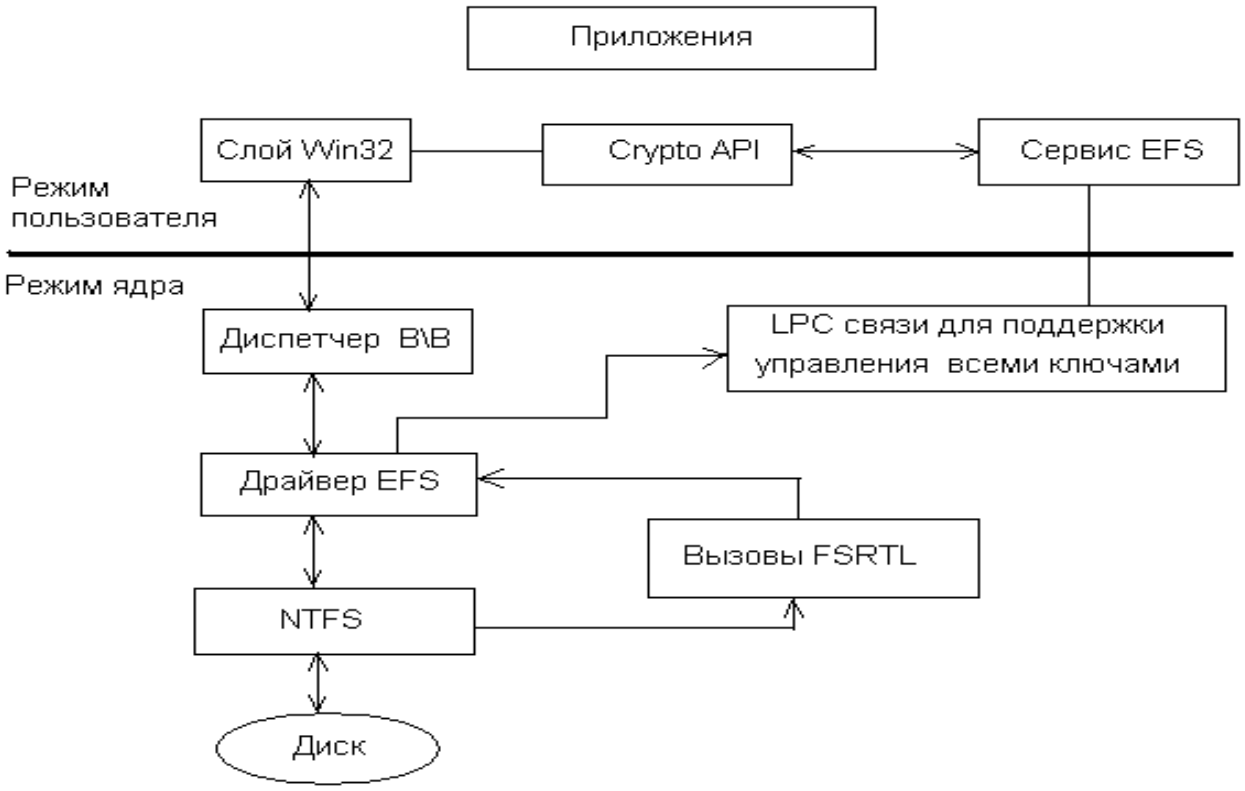

Рисунок 3 – Структурная схема EFS.

Система шифрования состоит из следующих компонентов:

Драйвер EFS. Драйвер EFS находится над NTFS. Он обменивается данными со службой EFS - запрашивает ключи шифрования, наборы DDF, DRF, - а также с другими службами управления ключами. Полученную информацию драйвер EFS передает библиотеке реального времени файловой системы EFS (File System Run- Time Library, FSRTL), которая прозрачно для операционной системы выполняет различные операции, характерные для файловой системы (чтение, запись, открытие файла, присоединение информации).

Библиотека реального времени файловой системы EFS. FSRTL - это модуль, находящийся внутри драйвера EFS, реализующий вызовы NTFS, выполняющие такие операции, как чтение, запись и открытие зашифрованных файлов и каталогов, а также операции, связанные с шифрованием, дешифрованием и восстановлением файлов при их чтении или записи на диск. Хотя драйверы EFS и FSRTL реализованы в виде одного компонента, они никогда не обмениваются данными напрямую. Для передачи сообщений друг другу они используют механизм вызовов (callouts) NTFS, предназначенный для управления файлами. Это гарантирует, что вся работа с файлами происходит при непосредственном участии NTFS. С помощью механизма управления файлами операции записи значений атрибутов EFS (DDF и DRF) реализованы как обычная модификация атрибутов файла. Кроме того, передача FEK, полученного службой EFS, в FSRTL

выполняется таким образом, чтобы он мог быть установлен в контексте открытого файла. Затем контекст файла используется для автоматического выполнения операций шифрования и дешифрования при записи и чтении информации файла.

Служба EFS. Служба EFS (EFS Service) является частью системы безопасности операционной системы. Для обмена данными с драйвером EFS она использует порт связи LPC, существующий между локальной подсистемой безопасности (Local *Security Authority,* LSA) и монитором безопасности, работающим в привилегированном режиме. В режиме пользователя для создания ключей шифрования файлов и генерирования данных для DDF и DRF служба EFS использует CryptoAPI. Она также поддерживает набор API дляWin32.

Набор API для Win32. Этот набор интерфейсов прикладного программирования позволяет выполнять шифрование файлов, дешифрование и восстановление зашифрованных файлов, а также их импорт и экспорт (без предварительного дешифрования). Эти API поддерживаются стандартным системным модулем DLL - advapi32.dll.

3. Безопасность компьютеров в сети.

Создание объектов политики безопасности в Active Directory.

Каждый домен должен обладать ассоциированной с ним *политикой безопасности.* Объект групповой политики безопасности (Group Policy Object, GPO) автоматически создается при создании домена. Впоследствии этот установленный по умолчанию объект может быть отредактирован. Используя политики безопасности, можно одновременно управлять поведением всех компьютеров в домене. Кроме того, можно легко проконтролировать, *кто* имеет право их администрировать. В каждом домене должна существовать *только одна* политика безопасности.

При формировании безопасности сети предприятия, очень важно *правильно организовать* политики безопасности, что позволит минимизировать избыточность параметров и оптимизировать управление сетью. К сожалению, эти цели находятся в некотором противоречии друг с другом. Для минимизации избыточности объекты GPO должны с максимальной точностью описывать каждую из политик, действующих на конкретные домены и организационные единицы (OU), что приводит к увеличению числа GPO. Для улучшения управления сетью следует, наоборот, создавать небольшое число GPO. Поэтому в каждом конкретном случае необходимо правильно сбалансировать количество объектов политик безопасности.

Для правильного планирования структуры политик безопасности сети следует сделать следующее:

- Разделить политики на логические группы. Например, политики учетных записей могут быть объединены в одну группу.
- Для каждой логической группы создать один или несколько объектов GPO, имеющих различные настройки политик безопасности.
- Распределить объекты-компьютеры по иерархическим древовидным структурам, состоящие из организационных единиц. В качестве критерия при таком распределении следует выбрать функцию, которую несет данный компьютер.

В основном, каждая OU должна иметь определенную политику безопасности, действующую на все находящиеся в ней компьютеры. Зачастую это сделать непросто, поскольку OU могут определять географическое расположение организации, а также и иерархию управления сетью.

В случаях, когда политики безопасности должны распространяться на подмножество компьютеров организации, можно сделать следующее:

- Создать в различных OU организации внутренние OU, переместить туда нужные объекты-компьютеры и назначить внутренним OU собственную политику безопасности.

- Если вы не хотите создавать внутренние OU, можно использовать схему фильтрации GPO, основанную на разрешениях доступа. С ее помощью вы сможете задать соответствие компьютеров и действующих на них GPO.

Протокол аутентификации Kerberos.

Протокол Kerberos представляет собой набор методов идентификации и проверки истинности партнеров по обмену информацией (рабочих станций, пользователей или серверов) в открытой (незащищенной) сети. Процесс идентификации *не зависит* от аутентификации, выполняемой сетевой операционной системой, *не основывается* в принятии решений на адресах хостов и *не предполагает* обязательную организацию физической безопасности всех хостов сети. Кроме того, допускается, что пакеты информации, передаваемые посети, могут быть модифицированы, прочитаны и переданы в любой момент времени. Следует, однако, отметить, что большинство приложений использует функции протокола Kerberos только при создании сеансов передачи потоков информации. В этом случае предполагается, что последующее несанкционированное разрушение потока данных невозможно. Поэтому применяется прямое доверие, основанное на адресе хоста. Kerberos выполняет аутентификацию как доверенная служба третьей стороны, используя криптографию с помощью общего секретного ключа (shared secret key).

Аутентификация выполняется следующим образом:

- Клиент посылает запрос серверу аутентификации (Authentication server , AS) на информацию, однозначно идентифицирующую определенный сервер.
- AS передает требуемую информацию, зашифрованную с помощью известного пользователю ключа. Переданная информация состоит из "билета" сервера и временного ключа, предназначенного для шифрования (часто называемого *ключом сеанса).*
- Клиент пересылает серверу билет, содержащий идентификатор клиента и ключ сеанса, зашифрованные с помощью ключа, известного серверу.
- Теперь ключ сеанса известен клиенту и серверу. Он может быть использован для аутентификации клиента, а также для аутентификации сервера. Он может быть применен для шифрования передаваемой в сеансе информации или для взаимного обмена ключами подсеанса, предназначенными для шифрования последующей передаваемой информации.

Работа Kerberos основана на одном или нескольких серверах аутентификации, работающих на физически защищенном хосте. Серверы аутентификации ведут базы данных партнеров по обмену информацией в сети (пользователей, серверов и т.д.) И их секретных ключей. Программный код, осуществляющий функционирование самого

протокола и процесса шифрования данных, находится в специальных библиотеках. Для того чтобы выполнять аутентификацию Kerberos для своих транзакций, приложения должны сделать несколько обращений к библиотекам Kerberos. Процесс аутентификации состоит из обмена необходимыми сообщениями с сервером аутентификации Kerberos.

Протокол Kerberos состоит из нескольких субпротоколов (или обменов сообщениями). Существует два метода, которыми клиент может запросить у сервера Kerberos информацию, идентифицирующую определенный сервер. Первый способ предполагает, что клиент посылает AS простой текстовый запрос билета для конкретного сервера, а в ответ получает данные, зашифрованные с помощью своего секретного ключа. Как правило, в данном случае клиент посылает запрос на *билет, позволяющий получить билет* (Ticket granting ticket, TGT), который в дальнейшем используется для работы с *выдающим билеты сервером* (Ticket granting Server, TGS). Второй способ предполагает, что клиент посылает TGT на TGS так же, как будто он обменивается информацией с другим сервером приложений, требующим аутентификации Kerberos.

Информация, идентифицирующая сервер, может быть использована для идентификации партнеров по транзакции, что позволит гарантировать целостность передаваемых между ними сообщений или сохранить в секрете передаваемую информацию.

Для идентификации партнеров по транзакции клиент посылает билет на сервер. Поскольку посылаемый билет "открыт" (некоторые его части зашифрованы, но они не помешают выполнить посылку копии) и может быть перехвачен и использован злоумышленником, ДЛЯ подтверждения истинности партнера, пославшего билет, передается дополнительная информация, называемая *а у т е н т и ф и к а т о р о м.* Она зашифрована с помощью ключа сеанса и содержит отсчет времени, подтверждающий, что сообщение было сгенерировано недавно и не является копией оригинальной посылки. Шифрование аутентификатора с помощью ключа сеанса доказывает, что информация была передана истинным партнером по обмену данными. Поскольку кроме запрашивающего партнера и сервера никто больше не знает ключа сеанса (он никогда не посылается по сети в открытом виде), с его помощью можно полностью гарантировать истинность партнера.

Целостность сообщений, которыми обмениваются партнеры, гарантируется с помощью ключа сеанса (передается в билете и содержится в информации идентификации партнера). Этот подход позволяет обнаружить атаки типа посылки злоумышленником перехваченной копии запроса и модификации потока данных. Это достигается генерированием и пересылкой *контрольной суммы* (хэш-функции) сообщения клиента, зашифрованной с помощью ключа сеанса. Безопасность и целостность сообщений, которыми обмениваются партнеры, может быть обеспечена шифрованием передаваемых

данных с помощью ключа сеанса, передаваемого в билете и содержащегося в информации идентификации партнера.

Описанная выше аутентификация требует доступа на чтение к базе данных Kerberos. Однако иногда записи базы данных могут быть модифицированы. Это происходит, например, при добавлении новых партнеров по обмену информацией или при изменении секретного ключа партнера. Изменения базы данных выполняются с помощью специального протокола обмена между клиентом и сервером Kerberos, применяющимся и при поддержке нескольких копий баз данных Kerberos.

Взаимодействие с удаленными владениями.

Протокол Kerberos может работать вне пределов одной компании. Клиент данной организации может быть аутентифицирован на сервере, находящемся в другой организации. Каждое предприятие, желающее при менять в своей сети Kerberos, должно установить границы своего "владения" (realm). Имя владения, в котором зарегистрирован данный пользователь, составляет часть его имени и может быть использовано конечной службой для принятия решения, должен ли данный запрос быть удовлетворен.

Установив общие ключи для владений, администраторы могут позволить клиентам различных владений выполнять *удаленную аутентификацию.* Конечно, обладая необходимыми разрешениями, клиент может выполнить регистрацию партнера, имеющего не связанное с данным владением имя, и установить нормальный обмен сообщениями. Однако даже при незначительном количестве удаленных регистраций такой подход приводит к большим неудобствам, поэтому рекомендуется применять более автоматизированные методы. При обмене общими во владениях ключами (inter realm keys) (при передаче информации в каждом направлении может быть использован отдельный ключ) службы выдачи билетов регистрируются в противоположном владении в качестве партнера по обмену данными. После этого клиент имеет возможность получать от локальной службы выдачи билетов TGT для такой же службы, находящейся в удаленном владении. При использовании этого TGT для его дешифрования удаленная служба выдачи билетов применяет общий для владений ключ, который, как правило, отличается от ключа TGS. Это гарантирует, что данный TGT был передан собственным TGS клиента. Билеты, посланные удаленной службой выдачи билетов, укажут конечной службе, что аутентификация клиента была выполнена в удаленном владении.

Одно владение может обмениваться информацией с другим владением, если оба они обладают общим ключом, или если локальная служба выдачи билетов обладает общим ключом с промежуточным владением, в свою очередь обладающим общим ключом с целевым владением. Путь аутентификации представляет собой последовательность

промежуточных владений, каждое из которых может обмениваться информацией со своими соседями.

Как правило, владения организованы в иерархическую структуру. Каждое владение обладает общим ключом со своим родителем и отдельным общим ключом с каждым дочерним владением. Если между двумя владениями не существует общего ключа, иерархическая структура позволяет легко установить путь аутентификации. Если иерархическая структура владений не используется, для обнаружения пути аутентификации следует применять базу данных.

Как правило, владения организованы в иерархическую структуру, поэтому аутентификацию можно осуществить, воспользовавшись альтернативными путями, минуя промежуточные владения. Это может позволить оптимизировать обмен данными между двумя владениями. Конечной службе важно знать, через какие владения проходит путь аутентификации, поскольку от этого зависит достоверность всего процесса аутентификации. Для облегчения принятия такого решения каждый билет содержит поле, где хранятся имена всех владений, которые составляют путь аутентификации.

Требования к рабочему окружению.

Протокол Kerberos налагает несколько требований на свое рабочее окружение, в котором он может эффективно работать:

- Kerberos не противодействует атакам типа "Отказ в обслуживании". Ряд особенностей протокола Kerberos позволяют злоумышленнику заставить приложение не принимать участие в процессе аутентификации. Обнаружение и борьба с таким типом атак (некоторые из которых могут проявляться В установлении необычных режимов работы программного обеспечения), как правило, лучше всего осуществляются администраторами систем.
- Партнеры по обмену данными должны хранить свои секретные ключи в надежном месте. Если злоумышленник каким-либо образом похитит секретный ключ, он сможет выдать себя за одного из партнеров или имперсонализировать сервер для законного клиента.
- Kerberos не противодействует атакам типа "Подбор пароля". Если пользователь устанавливает легко угадываемый пароль, злоумышленник, вполне вероятно, сможет его узнать подбором с применением словаря.
- Каждый хост сети должен иметь часы, которые приблизительно синхронизируются с часами других хостов. Синхронизация необходима, чтобы было легче обнаружить факт передачи копии заранее перехваченного сообщения. Степень приблизительности синхронизации может быть установлен индивидуально для

каждого сервера. Сам протокол синхронизации серверов сети должен быть защищен от атак злоумышленников.

- Идентификаторы партнеров не могут быть повторно использованы через небольшой промежуток времени. Как правило, для управления доступом используются *списки управления доступом* (Access Control List, ASL), в которых хранятся разрешения доступа, предоставленные всем партнерам по обмену данными. Если в базе данных списков управления доступом остался список уничтоженного партнера по обмену данными и идентификатор этого партнера используется вторично, новый партнер унаследует все права доступа уничтоженного партнера. Избежать подобной опасности можно, только если запретить использование идентификаторов уничтоженных партнеров в течение продолжительного времени, а лучше вообще сделать идентификаторы уникальными.

Протоколы обмена сообщениями.

Протокол службы аутентификации - предназначен для выполнения обмена информацией между клиентом и AS Kerberos.

Протокол аутентификации клиента и сервера используется сетевыми приложениями для аутентификации клиента в сервере и наоборот. Для успешного выполнения, аутентификации клиент должен заранее получить с помощью службы выдачи билетов или TOS информацию идентификации сервера.

Протокол выдачи билетов предназначен для обмена информацией между клиентом и сервером Kerberos, выдающим билеты (TOS). Он инициируется клиентом при необходимости получить информацию идентификации для определенного сервера (который может быть зарегистрирован в удаленном владении).

Протокол KRB_SAFE используется клиентами при необходимости обнаружения несанкционированной модификации сообщений, которыми они обмениваются. Модификация сообщений обнаруживается с помощью подсчета контрольной суммы данных пользователя и дополнительной контрольной информации. Контрольная сумма шифруется с помощью специального ключа, который выбирается в результате переговоров, или с помощью ключа сеанса.

Протокол KRB_PRIV используется клиентами при необходимости передачи чрезвычайно конфиденциальных данных и обнаружения несанкционированной модификации сообщений. Это делается с помощью подсчета контрольной суммы данных пользователя и дополнительной контрольной информации.

Протокол KRB_CRED используется клиентами при необходимости *от*правки информации идентификации Kerberos от одного хоста другому хосту.

База данных Kerberos.

Сервер Kerberos должен иметь доступ к базе данных, где хранятся информация об идентификаторах партнеров по обмену данными и секретные ключи аутентифицируемых партнеров. Реализация сервера Kerberos не предполагает обязательное расположение сервера и баз данных на одной машине. Существует возможность хранения базы данных партнеров по обмену данными в пространстве имен сети, если записи этой базы защищены от несанкционированного доступа. Однако с точки зрения целостности и безопасности данных этого делать не рекомендуется.

Модель распределенной безопасности Windows 2000.

Модель распределенной безопасности Windows 2000 основана на трех основных концепциях:

- Каждая рабочая станция и сервер имеют прямой доверенный путь (trust path) к контроллеру того домена, членом которого является данная машина. Доверенный путь устанавливается службой NetLogon с помощью аутентифицированного соединения RPC с контроллером домена. Защищенный канал устанавливается и с другими доменами Windows NT с помощью междоменных доверительных отношений. Он используется для проверки информации безопасности, включая идентификаторы безопасности (Security Identifiers, SID) пользователей и групп.
- Перед выполнением запрошенных клиентом операций сетевые службы имперсонализируют контекст безопасности этого клиента. Имперсонализация основана на маркере адреса безопасности, созданном локальной системой безопасности (Local Security Authority, LSA). Он представляет собой авторизацию клиента на сервере. Поток, находящийся на сервере и соответствующий данному клиенту, имперсонализирует контекст безопасности клиента и выполняет операции в соответствии с авторизацией данного клиента, а не в соответствии с идентификатором безопасности сервера. Имперсонализация поддерживается всеми службами Windows 2000, включая, например, службу удаленного файлового сервера CIFS\SNB. Аутентифицированный RPC и DCOM поддерживают имперсонализацию для распределенных приложений. Такие службы BackOffice, как Exchange, SNA Server и Internet Information Server, также поддерживают имперсонализацию.
- Ядро Windows 2000 поддерживает объектно-ориентированное управление доступом, сравнивая SID в маркере доступа с правами доступа, определенными в списке управления доступом данного объекта. Каждый объект Windows 2000 (ключи реестра, файлы и каталоги NTFS, общие ресурсы, объекты ядра, очереди печати и т.д.) имеют собственные списки управления доступом. Ядро Windows 2000 проверяет разрешения доступа при каждой попытке доступа к данному объекту. Управление доступом и аудит осуществляются с помощью настройки свойств безопасности

объекта, позволяющих предоставить доступ к объекту пользователю или группе. Управление авторизацией выполняется централизованно посредством включения пользователей в группы Windows 2000, которым предоставлены необходимые права доступа.

Безопасность IP.

Средства безопасности протокола IP позволяют управлять защитой всего IP трафика от источника информации до ее получателя. Возможности IP Security Management (Управление безопасностью IP) в системе Windows 2000 позволяют назначать и применять политику безопасности, которая гарантирует защищенный обмен информацией для всей сети. Механизм IP Security (Безопасность IP) реализован прозрачно для пользователя, администрирование безопасности централизовано и совмещает гарантии безопасного обмена информацией с легкостью использования.

Операционная система Microsoft Wil1dows 2000 Server упрощает развертывание и управление безопасностью сети при использовании средств Windows IP Security, которые являются гибкой и надежной реализацией Протокола безопасности IP (IP Security, IPSec).

Потребность в защите сетей, основанных на протоколе IP, уже достаточно велика и растет с каждым годом. В настоящее время в широко взаимосвязанном деловом мире сетей internet, intranet, extranet, филиалов и удаленного доступа, по сетям передается важная информация, конфиденциальность которой нельзя нарушать. Одним из основных требований, предъявляемых к сети со стороны сетевых администраторов и прочих профессионалов, обслуживающих и использующих сети, является требование гарантии, что этот трафик будет защищен от:

- Модификации данных во время пути по сети.
- Перехвата, просмотра или копирования.
- Доступа субъектов, не имеющих на это прав.

Эти проблемы характеризуются такими показателями, как *целостность данных, конфиденциальность* и *аутентичность*. Кроме того, защита от *повторного использования* (replay protection) предотвращает принятие повторно посланного пакета.

Преимущества безопасности IP.

Сетевые атаки могут привести к неработоспособности системы, считыванию конфиденциальных данных и к другим дорогостоящим нарушениям. Требуются методы "сильного" шифрования и сертификации, основанные на криптографических алгоритмах, для безопасности обмена информации. Однако высокий уровень безопасности не должен ухудшать производительность труда пользователей или увеличивать затраты на администрирование.

Безопасность IP в Windows 2000 реализована прозрачно для пользователя. Для информационного взаимодействия под управлением средств безопасности IP пользователи не обязательно должны располагаться в одном домене, они могут находиться в любом доверенном (trusted) домене в пределах корпоративной сети. IP Security имеет централизованное

администрирование. Политика безопасности' создается администратором домена для наиболее общих сценариев организации взаимодействия. Эта политика хранится в Active Directory и привязывается к доменной политике. Когда компьютер регистрируется в домене, доменная политика автоматически подбирает политику безопасности, что помогает избежать необходимости конфигурировать каждый компьютер индивидуально.

IP Security в Windows 2000 обеспечивают следующие преимущества, которые помогают достичь высокого уровня безопасности взаимодействия при низких затратах:

- Централизованное администрирование политикой безопасности, что уменьшает затраты на административные издержки. Политика IPSec может быть создана и назначена на уровне домена, что устраняет необходимость индивидуального конфигурирования каждого компьютера. Однако если компьютер имеет уникальные требования, или это автономный компьютер, политика может быть назначена непосредственно.
- Прозрачность безопасности **IP** для пользователей и прикладных программ. Не нужно иметь отдельные программные средства безопасности для каждого протокола в стеке TCP/IP, поскольку приложения, использующие TCP/IP, передают данные уровню протокола IP**,** где они шифруются. Установленная и настроенная служба IPSec являются прозрачной для пользователя, не требуя обучения.
- Гибкость конфигурирования политики безопасности, которая помогает решать задачи в различных конфигурациях. Внутри каждой политики могут быть настроены службы безопасности, чтобы обеспечить потребности на всех уровнях, от индивидуального пользователя до сервера или до уровня предприятия. Политика может быть сконфигурирована так, чтобы соответствовать экспортным правилам и ограничениям.
- Конфиденциальные службы, которые предотвращают попытки несанкционированного доступа к чувствительным данным, в то время как эти данные передаются между поддерживающими связь сторонами.
- Туннелирование. Данные могут быть посланы через безопасные туннели для обмена информацией в Internet и intranet.
- Усиленная служба аутентификации, которая предотвращает интерпретацию данных при помощи использования ложно предоставленных идентификаторов.
- Ключи с большой длиной и динамический повторный обмен ключами в течение текущих сеансов связи, что помогает защитить соединение против атак.
- Безопасная связь от начала до конца для частных пользователей сети внутри одного и того же домена или через любой доверенный домен внутри корпоративной сети.

- Безопасная связь от начала до конца между удаленными пользователями и пользователями в любом домене корпоративной сети, основанная на IP- адресах.
- Промышленный стандарт - IPSec, обеспечивает открытые возможности для частных технологий шифрования IP. Менеджеры сети извлекают выгоду из возникающей в результате использования открытых стандартов способности к взаимодействию с другими платформами и продуктами.
- Сертификаты с открытым ключом и поддержка ключей pre-shared. Это требуется для разрешения установления безопасной связи с компьютерами, которые не являются частью доверенного домена.
- IPSec работает в тандеме с другими механизмами защиты, сетевыми протоколами и базовыми механизмами защиты Windows 2000.
- Поддерживается шифрование сообщений RSVP для реализации QoS и ACS в Windows 2000. Не нужно отключать IPSec, чтобы получить полное преимущество от приоритетного управления шириной полосы пропускания, обеспечиваемого этими службами.

Криптография.

Криптография - набор математических методов для шифрования и дешифрации данных. Ее применение обеспечивает надежную передачу данных и предотвращение их получения несанкционированной стороной. Две функции криптографии, используемые Windows 2000 IP Security: криптография с закрытым ключом и криптография с открытым ключом.

Алгоритмы криптографии.

Методика Diffie- Hellman (D-H) - алгоритм шифрования с открытым ключом (названный по имени изобретателей), который позволяет двум поддерживающим связь сущностям договариваться об общедоступном ключе без требования шифрования во время порождения ключа. Процесс начинается двумя сущностями, обменивающимися общедоступной информацией. Затем каждый объект объединяет общую информацию другой стороны со своей собственной секретной информацией, чтобы сгенерировать секретное общедоступное значение.

Код аутентификации хэшированного сообщения (HMAC, Hash Message Authentication Code HMAC) - алгоритм с закрытым ключом. Целостность, установление подлинности и предотвращение повторного использования сообщений обеспечивается этим алгоритмом. Установление подлинности, использующее функции хэширования (перемешивания), объединено с методом закрытого ключа. Хэшированное значение, известное также как *дайджест*(digest) сообщений, используется для создания и проверки цифровой подписи. Это уникальное значение намного меньшее, чем первоначальное сообщение, созданное из цифровой копии кадра данных. Если передаваемое сообщение изменилось по пути следования, то хэшированное значение будет отличаться от оригинала, а IP-пакет будет отвергнут.

HMAC-MD5 дайджест сообщений-5 (MD5, Message Digest) - функция хэширования, которая порождает 128-разрядное значение. Значение - подпись данного блока данных. Эта

подпись используется для установления подлинности, целостности и предотвращения повторного использования.

HMAC-SHA - безопасный алгоритм хэширования (SHA, Secure Hash Algorithm) - это еще одна функция хэширования, которая порождает 160-разрядное значение подписи, используемое для установления подлинности, целостности и предотвращения повторного использования.

DES-CBC - стандарт шифрования данных (Data Encryption Standard, DES) - формирование цепочки шифрованных блоков (CBC, Cipher Block Chaining) алгоритм шифрования с закрытым ключом, используемый для конфиденциальности. Генерируется случайное число, которое используется совместно с закрытым ключом для шифрования данных.

Порядок выполнения работы.

1. Ознакомиться с теоретическими сведениями работы с MMC.
2. Выполнить основные настройки и операции для работы с консолью.
3. Сделать отчет по основным способам защиты.
4. Пройти тестирование на компьютере по теоретическому материалу.

Требования к отчету.

Отчет должен включать :

- название работы и ее цель;
- результаты основных операций для работы MMC;
- иллюстрации с примерами параметров и настроек MMC

Список литературы.

1. Таллоч М., Нортроп Т., Ханикатт Дж., Вилсон Э. Ресурсы Windows 7. – BHV Русская редакция, 2011 – 1104с.
2. Коварт Р., Уотерс Б. Windows NT Server4. – Питер, 2000- 448с.

.

4.