Дисциплина «Защита в операционных системах»

Лабораторная работа № 5

Тема: Подсистема аудита и шифрующая файловая система (EFS) в операционных системах семейства Windows.

Цель: изучить защитные механизмы операционной системы Windows, получить навыки практического использования ее средств обеспечения безопасности.

Время выполнения лабораторной работы (аудиторные часы) – 4 часа.

Оборудование и программное обеспечение: работа выполняется на ПЭВМ типа IBM PC с использованием стандартных функций ОС.

1. Теоретические сведения

1.1 Подсистема аудита

Важный элемент политики безопасности - аудит событий в системе. ОС Windows ведет аудит событий по 9 категориям:

1. Аудит событий входа в систему.
2. Аудит управления учетными записями.
3. Аудит доступа к службе каталогов.
4. Аудит входа в систему.
5. Аудит доступа к объектам.
6. Аудит изменения политики.
7. Аудит использования привилегий.
8. Аудит отслеживания процессов.
9. Аудит системных событий.

Рассмотрим более подробно, какие события отслеживает каждая из категорий.

Аудит событий входа в систему

Аудит попыток пользователя войти в систему с другого компьютера или выйти из нее, при условии, что этот компьютер используется для проверки подлинности учетной записи.

Аудит управления учетными записями

Аудит событий, связанных с управлением учетными записями на компьютере: создание, изменение или удаление учетной записи пользователя или группы; переименование, отключение или включение учетной записи пользователя; задание или изменение пароля.

Аудит доступа к службе каталогов

Аудит событий доступа пользователя к объекту каталога Active Directory, для которого задана собственная системная таблица управления доступом (SACL).

Аудит входа в систему

Аудит попыток пользователя войти в систему с компьютера или выйти из нее.

Аудит доступа к объектам

Аудит событий доступа пользователя к объекту - например, к файлу, папке, разделу реестра, принтеру и т. п., - для которого задана собственная системная таблица управления доступом (SACL).

Аудит изменения политики

Аудит фактов изменения политик назначения прав пользователей, политик аудита или политик доверительных отношений.

Аудит использования привилегий

Аудит попыток пользователя воспользоваться предоставленным ему правом.

Аудит отслеживания процессов

Аудиту таких событий, как активизация программы, завершение процесса, повторение дескрипторов и косвенный доступ к объекту.

Аудит системных событий

Аудит событий перезагрузки или отключения компьютера, а также событий, влияющих на системную безопасность или на журнал безопасности.

Решения об аудите конкретного типа событий безопасности принимаются в соответствии с политикой аудита локальной системы. Политика аудита, также называемая локальной политикой безопасности (local security policy), является частью политики безопасности, поддерживаемой LSASS в локальной системе, и настраивается с помощью редактора локальной политики безопасности (Оснастка **gpedit.msc, Конфигурация компьютера - Конфигурация Windows - Параметры безопасности - Локальные политики - Политика аудита,** рис. 7.1).

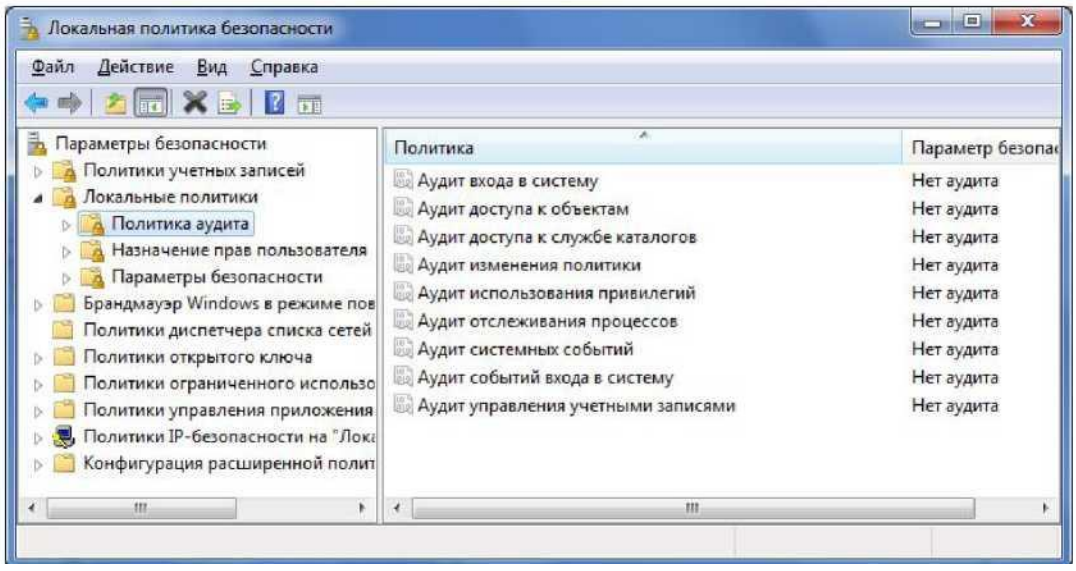

Рис. 7.1 Конфигурации политики аудита редактора локальной политики безопасности

Для каждого объекта в SD содержится список SACL, состоящий из записей ACE, регламентирующих запись в журнал аудита удачных или неудачных попыток доступа к объекту. Эти ACE определяют, какие операции, выполняемые

над объектами конкретными пользователями или группами, подлежат аудиту. Информация аудита хранится в системном журнале аудита. Аудиту могут подлежать как успешные, так и неудачные операции. Подобно записям ACE DACL, правила аудита объектов могут наследоваться дочерними объектами. Процедура наследования определяются набором флагов, являющихся частью структуры ACE.

Настройка списка SACL может быть осуществлена в окне дополнительных свойств объекта (пункт “Дополнительно”, закладка “Аудит”, рис. 7.2).

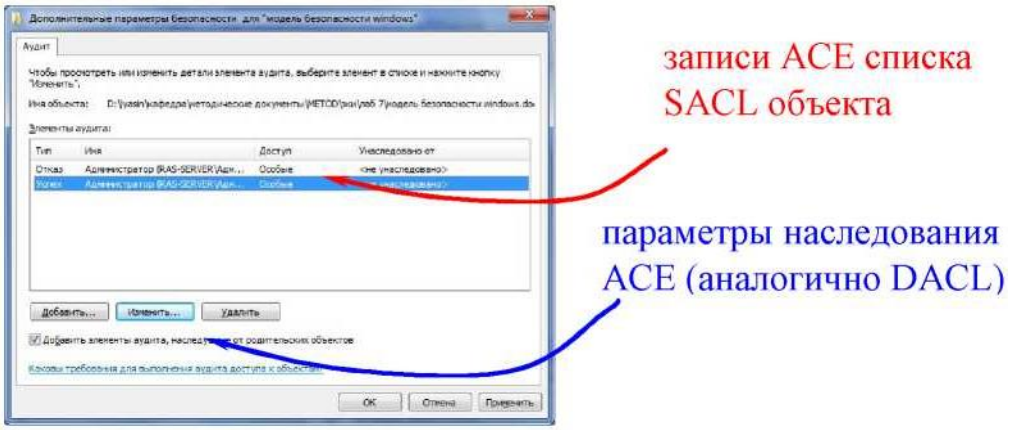

Рис. 7.2 Интерфейс редактирования правил аудита для объекта

Для программного просмотра и изменения списков SACL можно использовать API-функции **GetSecutitylnfo** и **SetSecutitylnfo**.

При инициализации системы и изменении политики LSASS посылает SRM сообщения, информирующие его о текущей политике аудита. LSASS отвечает за прием записей аудита, генерируемых на основе событий аудита от SRM, их редактирование и передачу Event Logger (регистратору событий). SRM посылает записи аудита LSASS через свое LPC-соединение. После этого Event Logger заносит записи в журнал безопасности.

Начиная с Windows Vista поддерживаются две категории журналов событий: **журналы Windows** и **журналы приложений и служб**. **Журналы Windows** - регистрируют общесистемных событий, и ведутся самой ОС. **Журналы приложений и служб** - индивидуальны для конкретных типов приложений и компонент (Internet Explorer, MediaCenter, PowerShell и др.). События аудита записываются в журналы Windows следующих типов (на примере Windows 7):

1. **Журнал приложений.** В журнале приложений содержатся данные, относящиеся к работе приложений и программ.
2. **Журнал безопасности.** Журнал безопасности содержит записи о таких событиях, как успешные и безуспешные попытки доступа в систему, а также о событиях, относящихся к использованию ресурсов.
3. **Журнал системы.** В журнале системы содержатся события системных компонентов Windows. Например, в журнале системы регистрируются сбои при загрузке драйвера или других системных компонентов при запуске системы.

4. **Журнал установки.** Фиксирует события, связанные с установкой или удалением компонент системы.

5. **Журнал перенаправления.** Фиксирует события, перенаправленные с соседних компьютеров.

Просмотр журнала безопасности осуществляется в оснастке «Просмотр событий» (**eventvwr.msc,** рис. 7.3). Сами журналы хранятся в файлах с расширением *.evtx в папке **%SystemRoot%\System32YWinevt\Logs\.**

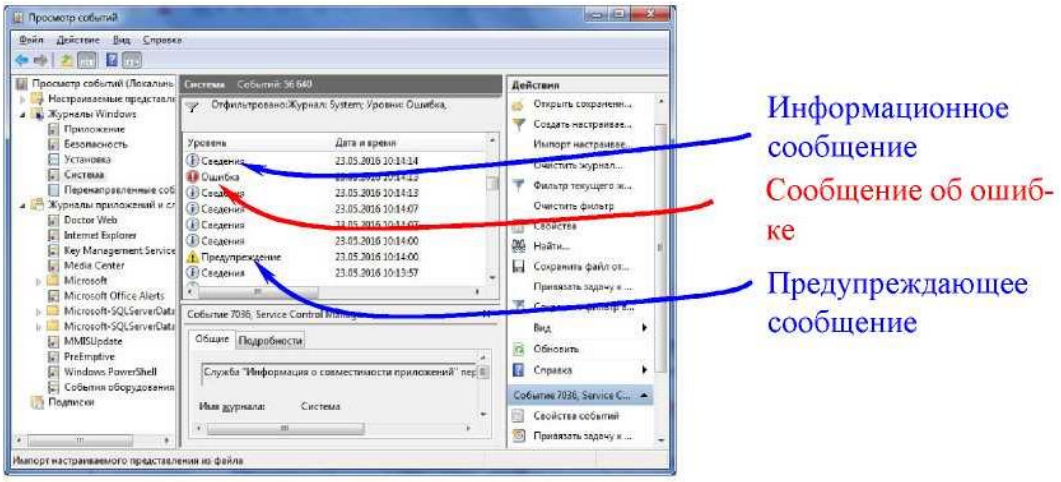

Рис. 7.3 Оснастка Windows «Просмотр событий»

В журнал записываются события различных типов:

- **Сведение** - сигнализирует об изменении в приложении или компоненте, например, успешном доступе к ресурсу, запуске приложения или службы;
- **Предупреждение** - сигнализирует о потенциально опасном событии, возникшем в приложении или компоненте, которые не мешают его работе, но могут стать причиной проблем в будущем;
- **Ошибка** - сигнализирует о проблему, сказывающемся на окружени приложения или компонента, вызвавших событие;
- **Критическая ошибка** - соответствует сбою, критичному для приложения или компонента, после которого они не могут продолжать работу;

1.2 Шифрующая файловая система.

Начиная с версии Windows 2000, в операционных системах семейства Windows NT поддерживается шифрование данных на разделах файловой системы NTFS с использованием *шифрующей файловой системы* (**Encrypted File System, EFS**). Основное ее достоинство заключается в обеспечении конфиденциальности данных на дисках компьютера за счет использования надежных симметричных алгоритмов для шифрования данных в реальном режиме времени.

Для шифрации данных EFS использует симметричный алгоритм шифрования (AES или DESX) со случайным ключом для каждого файла (**File Encryption Key, FEK**). По умолчанию данные шифруются в Windows 2000 и Windows XP по алгоритму DESX, а в Windows XP с Service Pack 1 (или выше) и

Windows Server 2003 — по алгоритму AES. В версиях Windows, разрешенных к экспорту за пределы США, драйвер EFS реализует 56-битный ключ шифрования DESX, тогда как в версии, подлежащей использованию только в США, и в версиях с пакетом для 128-битного шифрования длина ключа DESX равна 128 битам. Алгоритм AES в Windows использует 256-битные ключи.

При этом для обеспечения секретности самого ключа FEK шифруется асимметричным алгоритмом RSA открытым ключом пользователя, результат шифрации FEK - **Data Decryption Field, DDF** - добавляется в заголовок зашифрованного файла (рис. 7.4). Такой подход обеспечивает надежное шифрование без потери эффективности процесса шифрования: данные шифруются быстрым симметричным алгоритмом, а для гарантии секретности симметричного ключа используется асимметричный алгоритм шифрования.

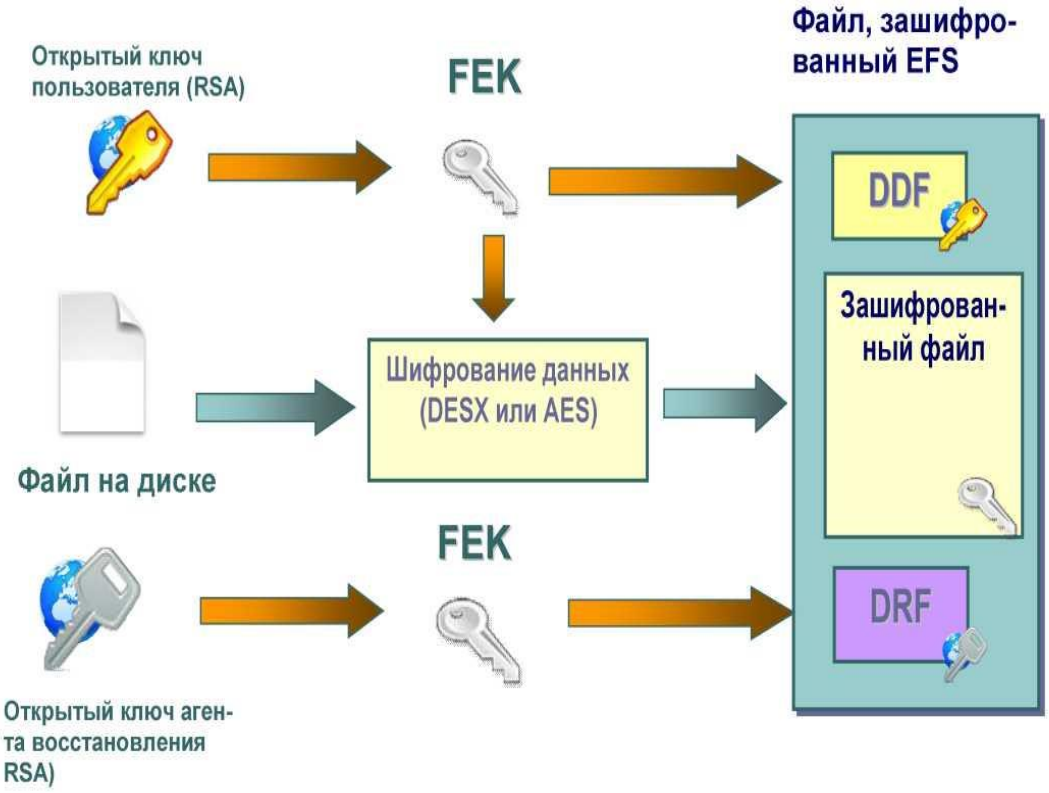

Рис.7.4 Схема шифрации файла в EFS

Для шифрации файлов с использованием EFS можно использовать графический интерфейс или команду **cipher**.

Графический интерфейс доступен в стандартном окне свойств объекта по нажатию кнопки **«Дополнительно»** (рис. 7.5). Зашифрованные объекты в стандартном интерфейсе Windows Explorer отображаются зеленым цветом.

Необходимо отметить, что EFS позволяет разделять зашифрованный файл между несколькими пользователями. В этом случае FEK шифруется открытыми ключами всех пользователей, которым разрешен доступ к файлу, и каждый результат шифрации добавляется в DDF.

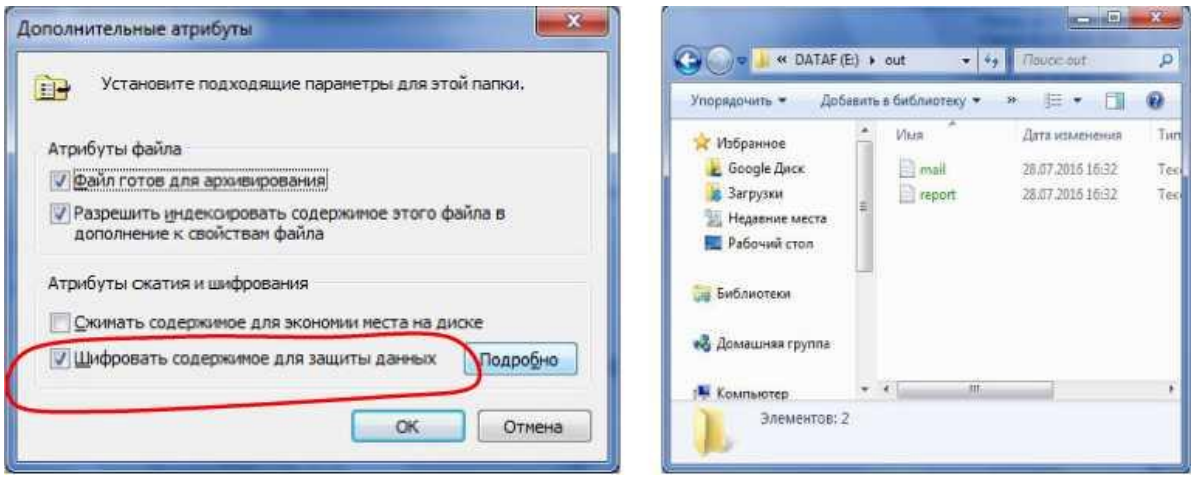

Рис. 7.5 Графический интерфейс шифрования файла с использованием EFS

Шифрование файла с использованием EFS защищает файл комплексно: пользователю, не имеющему права на дешифрацию файла, недопустимы, в том числе, такие операции, как удаление, переименование и копирование файла. Необходимо помнить, что EFS является частью файловой системы NTFS, и в случае копирования защищенного файла авторизованным пользователем на другой том с файловой системой, на поддерживающей EFS (например, FAT32), он будет дешифрован и сохранен на целевом томе в открытом виде.

Консольная команда cipher может быть использована для шифрации/дешифрации файлов из командной строки или в bat-сценарии.

cipher [{/e|/d}] *[^.каталог]* [/a] [/i] [/f] [/q] [/h] [/k] [/u[/n]] *[путь* [...]] |
[/r *:имя_файла_без_расширения*]

Назначения параметров команды приведены в таблице 7.1.

Таблица 7.1. Параметры команды cipher

/e	Шифрует указанные папки. Папки помечаются таким образом, чтобы файлы, которые будут добавляться в папку позже, также шифровались.
/d	Расшифровывает указанные папки. Папки помечаются таким образом, чтобы файлы, которые будут добавляться в папку позже, не будут шифроваться
/s: *каталог*	Выполняет выбранную операцию над указанной папкой и всеми подпапками в ней.
/a	Выполняет операцию над файлами и каталогами
/i	Продолжение выполнения указанной операции даже после возникновения ошибок. По умолчанию выполнение cipher прекращается после возникновения ошибки
/f	Выполнение повторного шифрования или расшифровывания указанных объектов. По умолчанию уже зашифрованные или расшифрованные файлы пропускаются командой cipher
/k	Создание ключа шифрования файла для пользователя, выполнившего команду cipher. Если используется данный параметр, все остальные параметры команды cipher не учитываются.

/u	Обновление ключа шифрования файла пользователя или ключа агента восстановления на текущие ключи во всех зашифрованных файлах на локальном диске (если эти ключи были изменены). Этот параметр используется только вместе с параметром /п.
/n	Запрещение обновления ключей. Данный параметр служит для поиска всех зашифрованных файлов на локальных дисках. Этот параметр используется только вместе с параметром /и.
путь	Указывает шаблон, файл или папку.
/r: *имя_файла*	Создание нового сертификата агента восстановления и закрытого ключа с последующей их записью в файлах с именем, указанным в параметре *имя файла* (без расширения). Если используется данный параметр, все остальные параметры команды cipher не учитываются.

Например, чтобы определить, зашифрована ли какая-либо папка, необходимо использовать команду:

***cipher* путь\имя_папки**

*Команда **cipher** без параметров выводит статус (зашифрован или нет) для всех объектов текущей папки.*

Для шифрации файла необходимо использовать команду

***cipher /e /a* путь\имя_файла**

Для дешифрации файла, соответственно, используется команда

***cipher /d /a* путь\имя_файла**

Допустима шифрация/дешифрация группы файлов по шаблону:

cipher /e /a d:\work*.doc

Пара открытый и закрытый ключ для шифрации FEK создаются для пользователя автоматически при первой шифрации файла с использованием EFS.

Если некоторый пользователь или группа пользователей зашифровали файл с использованием EFS, то его содержимое доступно только им. Это приводит к рискам утери доступа к данным в зашифрованных файлах в случае утраты пароля данным пользователем (работник забыл пароль, уволился и т.п.). Для предотвращения подобных проблем администратор может определить некоторые учетные записи в качестве агентов восстановления.

Агенты восстановления (Recovery Agents) определяются в политике безопасности **Encrypted Data Recovery Agents (Агенты восстановления шифрованных данных)** на локальном компьютере или в домене. Эта политика доступна через оснастку **Групповая политика (gpedit.msc)** раздел **«Параметры безопасности» -> «Политика открытого ключа» -> «Файловая система EFS». Пункт меню «Действие» -> «Добавить агент восстановления данных»** открывает мастер добавления нового агента.

Добавляя агентов восстановления можно указать, какие криптографические пары (обозначенные их сертификатами) могут использовать эти агенты для восстановления шифрованных данных (рис. 7.6). Сертификаты для агентов восстановления создаются командой **cipher** с ключом **/r** (см. табл. 7.1). Для пользователя, который будет агентом восстановления, необходимо

импортировать закрытый ключ агента восстановления из сертификата, созданного командой cipher. Это можно сделать в мастере импорта сертификатов, который автоматически загружается при двойном щелчке по файлу *.pfx.

EFS создает DRF (**Data Recovery Field**) - элементы ключей для каждого агента восстановления, используя провайдер криптографических сервисов, зарегистрированный для EFS-восстановления. DRF добавляется в зашифрованный файл и может быть использован как альтернативное средство извлечения FEK для дешифрации содержимого файла.

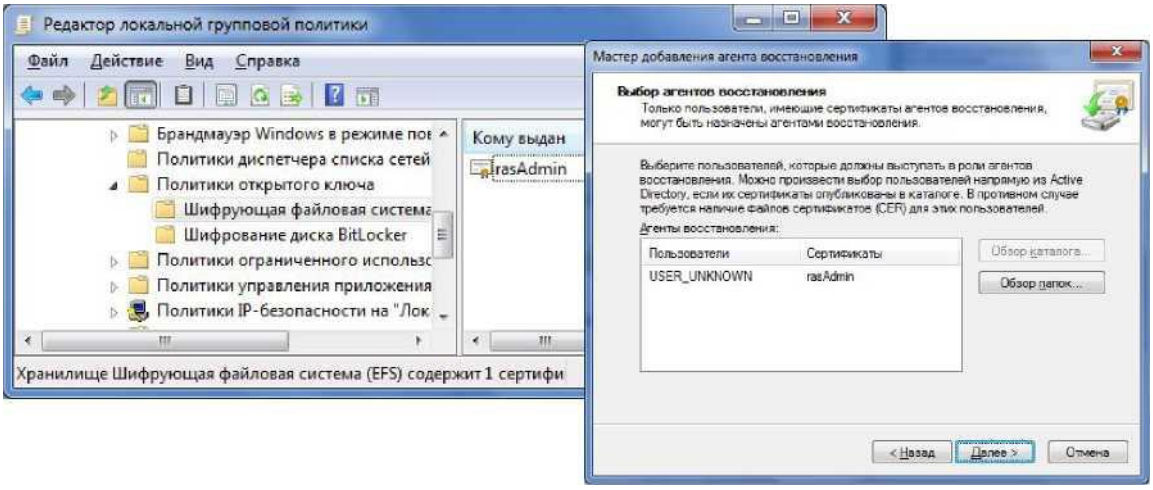

Рис. 7.6 Добавление нового агента восстановления EFS

Windows хранит закрытые ключи в подкаталоге **Application Data\Microsoft\Crypto\RSA** каталога профиля пользователя. Для защиты закрытых ключей Windows шифрует все файлы в папке RSA на основе симметричного ключа, генерируемого случайным образом; такой ключ называется мастер-ключом пользователя. Мастер-ключ имеет длину в 64 байта и создается стойким генератором случайных чисел. Мастер-ключ также хранится в профиле пользователя в каталоге Application Data\Microsoft\Protect и зашифровывается по алгоритму 3DES с помощью ключа, который отчасти основан на пароле пользователя. Когда пользователь меняет свой пароль, мастер-ключи автоматически расшифровываются, а затем заново зашифровываются с учетом нового пароля.

Для расшифровки FEK EFS использует функции Microsoft CryptoAPl (CAPI). CryptoAPI состоит из DLL провайдеров криптографических сервисов (cryptographic service providers, CSP), которые обеспечивают приложениям доступ к различным криптографическим сервисам (шифрованию, дешифрованию и хэшированию). EFS опирается на алгоритмы шифрования RSA, предоставляемые провайдером **Microsoft Enhanced Cryptographic Provider**.

Шифрацию и дешифрацию файлов можно осуществлять программно, используя API-функции EncryptFile и DecryptFile.

2. Задание

2.1. Ознакомьтесь с теоретическими основами защиты информации в ОС семейства Windows в настоящих указаниях и конспектах лекций.

2.2. Выполните задания 2.2.1 - 2.2.6

2.2.1. При выполнении лабораторной работы на компьютерах в учебной лаборатории запустите в программе **Oracle VM Virtualbox** виртуальную машину Win 7. Войдите в систему под учетной записью администратора.

2.2.2. Создайте учетную запись нового пользователя **testUser** в оснастке **«Управление компьютером» (compmgmt.msc)**. При создании новой учетной записи задайте произвольный пароль пользователя **testUser**, запретите пользователю смену пароля и снимите ограничение на срок действия его пароля. Создайте новую группу ”**testGroup**” и включите в нее нового пользователя. Удалите пользователя из других групп.

2.2.3. Создайте в разделе **HKLM\Software** реестра раздел **testKey**. Запретите пользователю **testUser** создание новых разделов в этом разделе реестра. Создайте для раздела **HKLM\Software\testKey** SACL, позволяющий протоколировать отказы при создании новых подразделов, а также успехи при перечислении подразделов и запросе значений (предварительно проверьте, что в локальной политике безопасности соответствующий тип аудита включен). Попробуйте от имени пользователя **testUser** запустить regedit.exe и создать раздел в **HKLM\Software**. Убедитесь, что записи аудита были размещены в журнале безопасности (eventvwr.msc).

2.2.4. Шифрование файлов и папок средствами EFS.

а) От имени пользователя **testUser** зашифруйте какой-нибудь файл на диске. Убедитесь, что после этого был создан сертификат пользователя, запустив оснастку certmgr.msc от имени пользователя (раздел Личные). Просмотрите основные параметры сертификата открытого ключа пользователя **testUser** (срок действия, используемые алгоритмы). Установите доверие к этому сертификату в вашей системе.

б) Создайте в папке **forTesting** новую папку **Encrypt**. В папке **Encrypt** создайте или скопируйте в нее текстовый файл. Зашифруйте папку **Encrypt** и все ее содержимое из меню свойств папки от имени администратора. Попробуйте просмотреть или скопировать какой-нибудь файл этой папки от имени пользователя **testUser**. Объясните результат. Скопируйте зашифрованный файл в незашифрованную папку (например, **forTesting**). Убедитесь, что он остался зашифрованным. Добавьте пользователя **testUser** в список имеющих доступа к файлу пользователей в окне свойств шифрования файла. Повторите попытку получить доступ к файлу от имени пользователя **testUser**.

в) Создайте учетную запись нового пользователя **agentUser**, сделайте его членом группы Администраторы. Определите для пользователя **agentUser** роль агента восстановления **EFS**. Создайте в папке **forTesting** новый текстовый файл с произвольным содержимым. Зашифруйте этот файл от имени пользователя **testUser**. Убедитесь в окне подробностей шифрования файла, что пользователь **agentUser** является агентом восстановления для данного файла. Попробуйте прочитать содержимое файла от имени администратора и от имени пользователя **agentUser**. Объясните результат.

г) Зашифруйте все текстовые файлы папки **forTesting** с использованием консольной команды шифрования cipher от имени пользователя **testUser**.

д) Убедитесь, что при копировании зашифрованных файлов на том с файловой системой, не поддерживающей EFS (например, FAT32 на флеш-

накопителе), содержимое файла дешифруется.

2.2.5. После окончания работы восстановить исходное состояние системы: удалить созданные папки и файлы, разделы реестра, удалить учетную запись созданного пользователя и его группы, снять с пользователя **agentUser** роль агента восстановления.

2.2.6. Подготовьте и представьте отчёт по лабораторной работе преподавателю и отчитайтесь за работу.

3. Контрольные вопросы

1. Какие события подлежат аудиту в ОС Windows?
2. Какие события регистрируются в журнале «Аудит изменения политики»?
3. Каким образом шифруются файлы в файловой системе EFS?
4. Что такое FEK, DDF, DRF?
5. Какие алгоритмы шифрования используются в EFS?
6. Какие функции выполняют агенты восстановления в ОС Windows?
7. Как зашифровать файл, используя систему EFS (приведите все возможные варианты)?

4. Требования к отчёту

Отчёт выполняется каждым студентом индивидуально. Работа должна быть оформлена в электронном виде в формате .doc и распечатана на листах формата А4. На титульном листе указываются: наименование учебного учреждения, наименование дисциплины, название и номер работы, вариант, выполнил: фамилия, имя, отчество, группа, проверил: преподаватель ФИО (образец титульного листа представлен в приложении 1).

Отчет должен содержать:

- название и цель работы;
- краткие теоретические сведения, ответы на контрольные вопросы;
- протокол выполнения лабораторной работы, содержащий список консольных команд, составленных при выполнении работы, и результаты их выполнения (в виде скриншотов).

выводы по результатам работы.

5. Литература

1. Соломон, Руссинович. Внутреннее устройство Microsoft Windows: Windows Server 2003, Windows XP и Windows 2000. 4-е издание, СПб.: Питер, 2008., 992 с.

2. А. Чекмарев, А. Вишневский, О. Кокорева Microsoft Windows Server 2003. Русская версия. Наиболее полное руководство., СПб.: БХВ-Петербург, 2008 г., 1120 с.

3. Лясин Д.Н., Саньков С.Г. Методы и средства защиты компьютерной информации (учебное пособие). - Волгоград, Издательство ВолгГТУ РПК "Политехник”, 2005г.

4. Лясин Д.Н., Саньков С.Г. Методические указания к лабораторным работам по курсу «Защита информации», 2011.

5. У. Р. Станек. Командная строка Microsoft Windows. Справочник администратора. М.: Русская редакция, 2009., 480с.

6. Безопасность Windows Server 2003 в библиотеке Microsoft TechNet. http://technet.microsoft.com/ru-ru/library/dd548350%28WS.10%29.aspx

7. Набор утилит Sysinternals. Утилита ProcessExplorer. https://technet.microsoft.com/ru-RU/sysinternals/bb896653

Приложение 1

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» (ДГТУ)

Факультет Информатика и вычислительная техника

Кафедра Кибербезопасность информационных систем

Лабораторная работа № ____
на тему «_____»

Выполнил обучающийся гр._____

_____
(Фамилия, Имя, Отчество)

Проверил:

_____
(должность, Фамилия, Имя, Отчество)

Ростов-на-Дону
20____