МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»
(СГУ)

Кафедра теоретических основ компьютерной безопасности и криптографии

Диагностика состояния компьютера на основе дескриптора безопасности

АВТОРЕФЕРАТ

дипломной работы

студента 6 курса 631 группы
специальности 10.05.01 Компьютерная безопасность
факультета компьютерных наук и информационных технологий
Генералова Александра Сергеевича

Научный руководитель
доцент, к.ю.н. _____ А.В. Гортинский
18.01.2019 г.

Заведующий кафедрой
д. ф.-м. н., доцент _____ М. Б. Абросимов
18.01.2019 г.

Саратов 2019

ВВЕДЕНИЕ

Одной из главных проблем, с которой столкнулось современное общество, стало обеспечение эффективной защиты безопасности в различных автоматизированных компьютерных системах, которые получили повсеместное применение благодаря развитию новых информационных технологий, расширению возможностей программирования и web-разработки.

Сейчас можно выделить несколько базовых принципов, которые лежат в основе любой системы обеспечения защиты информации, а именно:

• гарантия целостности данных;

• гарантия конфиденциальности информации;

• гарантия защищенного доступа в систему только для зарегистрированных пользователей. Важным аспектом работ по защите данных, которые сегодня активно проводятся как профессиональными компаниями, так и обычными студентами, будет являться не только теоретическое обоснование внедренных мер по защите информации, но и практическая реализация или модернизация определённого набора компонентов, которые отвечают за сохранность информации и цикличную проверку системы на наличие уязвимостей.

Если один из принципов был нарушен злоумышленником, то специалисту по компьютерной безопасности или компьютерному криминалисту нужно выяснить причину возникновения уязвимости. При этом необходимо найти следы, оставленные злоумышленником, и попробовать воспроизвести его действия с целью предотвращения данной уязвимости в системе. Одним из способов получения следов является файловая система операционной системы.

Целью данной дипломной работы является создание программного комплекса по работе со структурами защиты данных (дескрипторами

безопасности, уникальные идентификаторы пользователей) в файловой системой NTFS, которая используется в ОС Windows.

Для достижения поставленных целей требуется решить следующие задачи:

- Изучить порядок обеспечения доступа к файлам в файловой системе NTFS, используя информацию, хранящуюся в MFT.
- Изучить правила хранения и применения дескриптора безопасности при обращении к файлам в режиме защиты доступа.
- Разработать алгоритм доступа к файлу и извлечения дескриптора безопасности в условиях отсутствия доступа к файловой системе стандартными средствами ОС.
- Разработать программу поиска всех файлов в разделе NTFS, имеющих выбранный дескриптор безопасности или соответствующий ему SID, а также изменяющую дескриптор безопасности выбранного файла на один из имеющихся.
- Обеспечить возможность получения информации о зарегистрированном имени пользователя по SID в случае исследования диска с установленной операционной системой.

Дипломная работа состоит из введения, 3 разделов, заключения, списка использованных источников и 2 приложений. Общий объем работы – 89 страниц, из них 52 страницы – основное содержание, включая 62 рисунков и 4 таблиц, список использованных источников из 11 наименований.

КРАТКОЕ СОДЕРЖАНИЕ

Во введении ставятся цели и задачи дипломной работы.

В первом разделе «Основные компоненты NTFS» приводятся сведения для работы с данной файловой системой.

Все файловые операции системы выполняются в режиме ядра с помощью системных файлов драйверов для NTFS. Они используют MBR и загрузочный сектор, чтобы найти свой путь на диске. MBR (Master Boot Record) или Главная Загрузочная Запись — это первый сектор (самые первые 512 байт участка памяти) носителя информации (будь то жесткий диск (HDD) или твердотельный накопитель (SSD)). MBR содержит:

1. Содержит код и данные (446 байт — начальный загрузчик) которые необходимы BIOS, для начала загрузки ОС.

2. Содержит информацию о разделах жесткого диска (4 первичных раздела по 16 байт каждый). Эта информация называется таблица разделов (Partition Table). Здесь же содержится информация о ФС.

3. Биты, указывающие на конец записи (0xAA55, размер — 2 байта).

Также на носителе содержится «сердце» NTFS — главная файловая таблица MFT (Master File Table), содержащая информацию обо всех файлах и каталогах, и её копия (Master File Table Copy). MFT имеет очень большие записи. Действительно, многие небольшие файлы полностью хранятся в MFT. Как правило, MFT выделено около 12,5% от размера раздела, хотя это значение может быть изменено для того, чтобы поместить аварийный образ файловой системы. При необходимости часть этого распределения используется для размещения других файлов. Оставшаяся часть больших файлов хранятся в области системных данных файла, который составляет основную часть раздела. Копия таблицы файлов хранит первые четыре записи системы.

Первые 16 файлов NTFS (метафайлы) носят служебный характер. Каждый из них отвечает за какой-либо аспект работы системы.

Метафайлы находятся в корневом каталоге NTFS диска — они начинаются с символа имени «$», хотя получить какую-либо информацию о них стандартными средствами сложно. Любопытно, что и для этих файлов указан вполне реальный размер — можно узнать, например, сколько операционная система тратит на каталогизацию всего диска, посмотрев размер файла $MFT.

Основная информация о файле содержится в файловой записи (File Record) размером которой 1 КБ таблицы MFT.

Файловая запись состоит из заголовка (Header) и набора атрибутов (Attribute). В заголовке содержится служебная информация о файловой записи, например, её тип и размер. Все данные, относящиеся непосредственно к файлу, хранятся в виде атрибутов. Названия атрибутов также как и системных файлов начинаются со знака "$".

Во втором разделе «Сведения, связанные с доступом к данным и безопасности» описываются структуры данных, которые отвечают за выполнение базовых принципов систем обеспечения защиты информации.

Дескрипторы безопасности используются для определения политики контроля доступа к файлам или каталогам и хранятся в файле метаданных файловой системы $Secure.

Атрибут $STANDARD_INFORMATION любого файла или каталога содержит числовой код, называемый идентификатором безопасности (Security ID). Его значение используется для индексирования файла $Secure для поиска соответствующего дескриптора. Эти 32-разрядные идентификаторы безопасности отличаются от идентификаторов безопасности Windows (SID), присваиваемых системой пользователям. Идентификаторы безопасности уникальны только в рамках файловой системы, тогда как коды SID глобально-уникальны.

Файл $Secure содержит два индекса ($SDH и $SII) и один атрибут $DATA ($SDS). Атрибут $DATA содержит дескрипторы безопасности, а два индекса

используются при ссылках на дескрипторы. Индекс $SII сортируется по идентификатору безопасности, хранящемуся в атрибуте $STANDARD_INFORMATION каждого файла. Индекс $SII используется для поиска дескриптора безопасности файла при известном идентификаторе безопасности. С другой стороны, индекс $SDH сортируется по хеш-коду дескриптора безопасности. Операционная система использует этот индекс, когда к файлу или каталогу применяется новый дескриптор безопасности. Если хеш-код нового дескриптора найти не удается, система создает новый дескриптор и идентификатор безопасности и включает их в оба индекса.

SID — это структура данных переменной длины, которая идентифицирует учетную запись пользователя, группы или компьютера. Каждой учетной записи ставится в соответствие уникальный идентификатор, SID, в момент создания учетной записи. Система оперирует с SID учетных записей, а не их именами. Это значит, что если в системе был пользователь с именем «Иван», а потом его удалили и создали заново с тем же именем, то SID у него уже будет другой.

В третьем разделе «Практическая часть» описывается алгоритм получения файлов по дескриптору безопасности и по SID пользователя, изменения дескриптора безопасности файла с целью получения к нему доступа и рассматриваются возможные варианты работы программного комплекса с точки зрения обнаружения следов несанкционированного доступа.

Алгоритм получения файлов по дескриптору безопасности:

1. Выбираем диск с ФС NTFS.
2. Находим в MBR информацию о ФС. В данном случае нам потребуются:
    а) Смещение файловой таблицы MFT.
    б) Количество байтов в секторе.
    в) Количество секторов в кластере.

3. Переходим по смещению из пункта 1 и находим метафайл $Secure. По идентификатору 80 находим смещение до Run List, о котором было рассказано в пункте 1.4. Именно здесь находятся вся информация о дескрипторах безопасности, а именно где они расположены и сколько кластеров занимают. Вхождений может быть несколько.

4. Извлекаем информацию о дескрипторах безопасности, пока список вхождений не пуст. Будем получать информацию о:

а) Номер дескриптора безопасности (Security Id).

б) SID владельца файла.

в) SID группы.

5. Выбираем дескриптор безопасности, по которому будем искать.

6. Аналогично пункту 2 находим для метафайла $MFT по идентификатору 80 смещение до Run List. Здесь будет храниться информация о расположениях всех записей MFT. Вхождений может быть несколько.

7. Извлекаем информацию о записях MFT, пока список вхождений не пуст. Будем получать информацию о:

а) MFT ID – номер записи в MFT, находится в заголовке записи MFT по смещениям 0x34-0x37.

б) Security Id – собственно номер дескриптора безопасности. Находится в атрибуте $STANDART_INFORMATION по смещениям 0x34-0x37.

в) Parent directory – MFT ID родительской записи. Находится в атрибуте $FILE_NAME по смещениям 0x00-0x08.

г) File names – имена файла в различных форматах (DOS, Unicode). Каждое имя находится в отдельном атрибуте $FILE_NAME. 0x40 – размер имени, 0x41 – код пространства имени файла (1 – DOS, 3 –

Unicode), 0x42-0x42+(2*N) – имя файла, где N – значение по смещению 0x40.

8. Сравниваем номер выбранного на шаге 4 дескриптора безопасности с номером дескриптором безопасности просматриваемого файла. Если они совпадают, то добавляем в ответ полное имя файла, которое мы получаем рекурсивным обходом всех родительских каталогов.

Была создана программа на языке программирования Java, использующая библиотеку JavaFX для создания графического интерфейса, а также Apache Maven для конфигурации и сборки программы на ОС Windows. Для работы с диском, как с простым файлом, использовался реализованный стандартный класс RandomAccessFile. Для корректного функционирования программы необходимо запускать её от имени администратора, так как только ему позволено работать с диском на таком уровне.

Также была написана программа на языке программирования C++, которая позволяет произвести запись информации на диск. Аналогично описанной выше программе для корректного функционирования программы необходимо запускать её от имени администратора.

ЗАКЛЮЧЕНИЕ

Обеспечение эффективной защиты безопасности в различных автоматизированных компьютерных системах является наиболее острой проблемой в сферы защиты информации. Злоумышленники каждый раз придумывают новые методы по получению несанкционированного доступа к системе, в результате чего раскрытие их следов может занять продолжительное время. Поэтому специалистам в сфере безопасности необходимы новые методы по выявлению и предотвращению уязвимостей.

В результате проделанной работы был разработан программный комплекс, который позволяет:

1. Поиск файлов по дескриптору безопасности для заданного диска.
2. Поиск всех SID на диске и соответствие с именами пользователей системы.
3. Поиск файлов по SID для заданного диска.
4. Определение команды для смены дескриптора безопасности файлов.
5. Чтение информации с диска в виде байтов с проверкой на права доступа.
6. Запись информации на диск с проверкой на корректность записи.

Также были проведены эксперименты с различными пользователями (администратор, зарегистрированный пользователь до и после удаления, а также с пользователями другого компьютера) и состояниями файловой системы. По итогам установленных экспериментов можно сделать следующие выводы:

1. Программный комплекс позволяет определять зарегистрированных и незарегистрированных в системе пользователей, что может помочь

эксперту в определении количества пользователей, использовавших данный жесткий диск.

2. Программный комплекс позволяет работать при поврежденной файловой системе. Эксперту не потребуются никакие дополнительные действия, чтобы программа заработала.
3. Программный комплекс позволяет реагировать на изменения в системе (в данном случае на изменение дескриптора безопасности файла и на удаление или появление новых пользователей). Эксперту не потребуется перезагружать систему или чистить кэш после каждого действия.
4. Программный комплекс позволяет писать в любой раздел любого диска. Это дает эксперту широкие возможности в проведении экспериментов и установлении истинных причин изменения информации на диске.
5. Программный комплекс позволяет работать с внешними жесткими дисками и давать эксперту информацию о вероятном количестве пользователей, использовавших этот жесткий диск на другом компьютере, а также принадлежность тех или иных файлов к дескрипторам безопасности данного диска.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кэрриэ Б. Криминалистический анализ файловых систем. – СПб.: Питер, 2007. – 480 с.: ил.
2. Идентификаторы учетных записей в Windows 2000 / XP / 2003 / VISTA [электронный ресурс]: статья, открытый доступ.
   URL: http://ntinside.narod.ru/sid.html (дата обращения 09.01.19) Загл. с экрана. Яз. рус.
3. COEN 252 Computer Forensics NTFS [электронный ресурс]: статья, открытый доступ.
   URL:
   http://www.cse.scu.edu/~tschwarz/coen252_07Fall/Lectures/NTFS.html
   (дата обращения 09.01.19) Загл с экрана. Яз. англ.
4. Руссинович М., Соломон Д. Внутреннее устройство Microsoft Windows. – СПб.: Питер, 2013. –800с.: ил.
5. Лекция 11: файловая система NTFS [электронный ресурс]: лекция, открытый доступ.
   URL: http://www.intuit.ru/studies/courses/10471/1078/lecture/16586
   (дата обращения 09.01.19) Загл. с экрана. Яз. рус.
6. Practical Digital Forensics at Accession for Born-Digital Institution Records [электронный ресурс]: статья, открытый доступ.
   URL: http://journal.code4lib.org/articles/11239 (дата обращения 09.01.19) Загл с экрана. Яз. англ.
7. Хорошо известные идентификаторы безопасности в операционных системах Windows [электронный ресурс]: статья, открытый доступ.
   URL: https://support.microsoft.com/ru-ru/help/243330/well-known-security-identifiers-in-windows-operating-systems (дата обращения 09.01.19) Загл с экрана. Яз. рус.

8. А.П. Побегайло. Системное программирование в Windows. –СПб.: БХВ-Петербург, 2006. – 1056 с.: ил.
9. FAT32 vs NTFS In a Forensic Environment [электронный ресурс]: статья, открытый доступ.

   URL: https://www.giac.org/paper/gsec/3622/fat-32-ntfs-forensic-environment/105216 (дата обращения 09.01.19) Загл с экрана. Яз. англ.
10. File system forensic analysis [электронный ресурс]: статья, открытый доступ.

    URL:

    http://www.campus64.com/digital_learning/data/cyber_forensics_essentials/info_file_system_forensic_analysis.pdf (дата обращения 09.01.19) Загл с экрана. Яз. англ.
11. Файловая система NTFS [электронный ресурс]: статья, открытый доступ.

    URL: https://www.ixbt.com/storage/ntfs.html (дата обращения 09.01.19) Загл с экрана. Яз. рус.