Математические структуры и моделирование
2011, вып. 22, с. 121–130

УДК 004.056.52

ИССЛЕДОВАНИЕ БЕЗОПАСНОСТИ ДИСКРЕЦИОННОГО РАЗДЕЛЕНИЯ ДОСТУПА В ОС WINDOWS

С. В. Белим, Д. М. Бречка

На основе модели HRU исследуется дискреционное разделение доступа в операционных системах семейства Windows.

Введение

На сегодняшний день разработано несколько математических моделей подсистем безопасности компьютерных систем, допускающих гарантированную защищённость от утечек информации. Однако при попытке применения полученных результатов к реальным системам приходится сталкиваться с рядом трудностей. В данной статье предпринимается попытка описания механизма дискреционного разделения доступа подсистемы безопасности ОС Windows в рамках модели HRU [2–5, 8].

Рассмотрим основные моменты разграничения доступа в ОС Windows [7, 9, 10]. Модель защиты Windows требует, чтобы поток заранее указывал операции, которые он собирается выполнить с объектом. Система проверяет права доступа, запрошенные потоком, и, в случае наличия разрешений, выдаёт потоку дескриптор, позволяющий осуществлять операции с объектом.

Права доступа к конкретному объекту хранятся в структурах, называемых списками контроля доступа ACL (Access Control List). Ссылки на ACL объекта хранятся в дескрипторе безопасности объекта DS. Для каждого объекта существуют системный ACL (SACL, System ACL) и избирательный ACL (DACL, Discretionary ACL). DACL администрируется владельцем объекта и предназначен для разделения доступа к объекту. SACL объекта администрируется системным администратором и предназначен для аудита действий с объектом. Списки контроля доступов состоят из записей ACE (Access Control Entry), каждая из которых отвечает за разрешение или запрет одного права доступа.

В операционных системах семейства Windows существует три вида прав доступа к объектам:

1) стандартные права доступа — это права доступа, применимые к любому объекту (изменение владельца объекта, изменение списка DACL объекта, удаление объекта и т. д.);

Омский государственный университет им. Ф. М. Достоевского.
E-mail: sbelim@omsu.ru

2) специальные права доступа — это права доступа, применимые только к объектам данного вида (чтение данных из объекта, запись данных в объект, чтение атрибутов объекта, выполнение программного файла и т. д.);

3) общие права доступа — комбинации специальных и стандартных прав доступа. Общие права доступа вводятся для того, чтобы при установке разрешений на доступ абстрагировать владельца от конкретной реализации объекта.

Процесс преобразования общего права доступа к объекту в набор специальных и стандартных прав называется отображением права доступа.

Определены следующие общие права доступа:

1) чтение, включающее в себя чтение DACL объекта, чтение данных из объекта, чтение его атрибутов и расширенных атрибутов, использование объекта для синхронизации;

2) запись, включающая в себя чтение DACL объекта, запись и добавление данных в объект, запись его атрибутов и расширенных атрибутов, использование объекта для синхронизации;

3) выполнение, включающее в себя чтение DACL объекта, чтение его атрибутов, выполнение программного файла и использование объекта для синхронизации;

4) все действия с объектом.

Модель HRU названа по первым буквам фамилий ученых, предложивших её (Харисон, Руззо, Ульман). Основное назначение этой модели состоит в анализе политики безопасности, построенной на основе матрицы доступов. Всю совокупность DACL объектов компьютерной системы можно представить в виде одной таблицы, называемой матрицей доступов M. В модели HRU определены шесть примитивных операторов, преобразующих матрицу доступов:

1. $Enter(r, M[S, O])$ — внести право $r \in R$ в ячейку $M[S, O]$, т. е. объект S получает право доступа r к объекту O.

2. $Delete(r, M[S, O])$ — удалить право r из ячейки $M[S, O]$, т. е. запретить субъекту S доступ r к объекту O.

3. $CreateS(S)$ — создать объект S. При этой операции в матрице доступов появляются дополнительный столбец и дополнительная строка. Вновь создаваемый субъект не имеет никаких прав.

4. $CreateO(O)$ — добавление нового объекта O. В матрице доступов появляется новый столбец, на который ни у кого нет никаких прав доступа.

5. $DeleteS(S)$ — уничтожить субъект S. В таблице удаляется одна строка и один столбец.

6. $DeleteO(O)$ — уничтожить объект O. В таблице удаляется один столбец.

Из примитивных операторов могут составляться команды. Каждая команда состоит из двух частей:

1. Условия, при которых возможно выполнение команды.

2. Последовательность примитивных операторов.

$$\begin{array}{ll} command & C(X_1, ..., X_k) \\ & if\ r_1 \in M[X_1]\ and ... and\ r_k \in M[X_k]\ then \\ & \alpha_1;\ \alpha_2;\ ...\ \alpha_n; \\ end. & \end{array}$$

Здесь через $X_i (i = \overline{1,k})$ обозначены пары $X_i = (S_{im}, O_{ij})$. Условия в теле команды являются необязательными, если известно, что система гарантированно находится в нужном состоянии.

1. Описание прав доступа ОС Windows в рамках модели HRU

Для корректного описания системы в рамках модели HRU необходимо корректно определить множество возможных прав доступа. Как видно из спецификации подсистемы безопасности ОС Windows, элементарные операции задаются стандартными и специальными правами доступа, тогда как общие права доступа являются комбинацией стандартных и специальных прав, т. е. командами.

Для записи конкретной команды, соответствующей некоторому общему праву доступа, будем рассматривать объект операционной системы как совокупность объектов. Такое рассмотрение допустимо в рамках субъектно-объектного подхода, который требует, чтобы конкатенация двух объектов также была объектом.

Для обозначения какой-то части объекта будем указывать общий идентификатор объекта и идентификатор соответствующей части, разделённые точкой. Например, содержимое файла F будет иметь обозначение $F.Data$. При работе с файлом важными структурами являются его заголовок $F.Header$, дескриптор безопасности $F.DS$ и списки контроля доступа $F.DACL$, $F.SACL$.

Выпишем команды, соответствующие общим правам доступа к файлу F.

1. Чтение r файла F неким процессом S:

$$\begin{array}{ll} command & Read\_File((S, F)) \\ & Enter(r, M[S, F.Header]); \\ & Enter(r, M[S, F.Data]); \\ & Enter(r, M[S, F.DS]); \\ & Enter(r, M[S, F.DACL]); \\ & Enter(r, M[S, F.SACL]); \\ end. & \end{array}$$

2. Запись w в файл F неким процессом S:

$$
\begin{aligned}
&\mathit{command} \quad Write\_File((S,F))\\
&\qquad\qquad Enter(w, M[S, F.Header]);\\
&\qquad\qquad Enter(w, M[S, F.Data]);\\
&\qquad\qquad Enter(w, M[S, F.DS]);\\
&\qquad\qquad Enter(r, M[S, F.DACL]);\\
&\qquad\qquad Enter(w, M[S, F.SACL]);\\
&\mathit{end.}
\end{aligned}
$$

3. Запуск исполненяемого файла F неким процессом S:

$$
\begin{aligned}
&\mathit{command} \quad Execute\_File((S,F))\\
&\qquad\qquad Enter(r, M[S, F.Header]);\\
&\qquad\qquad Enter(x, M[S, F.Data]);\\
&\qquad\qquad Enter(r, M[S, F.DS]);\\
&\qquad\qquad Enter(r, M[S, F.DACL]);\\
&\qquad\qquad Enter(r, M[S, F.SACL]);\\
&\mathit{end.}
\end{aligned}
$$

4. Полный доступ к файлу F неким процессом S:

$$
\begin{aligned}
&\mathit{command} \quad All\_File((S,F))\\
&\qquad\qquad Enter(r, M[S, F.Header]);\\
&\qquad\qquad Enter(w, M[S, F.Header]);\\
&\qquad\qquad Enter(r, M[S, F.Data]);\\
&\qquad\qquad Enter(w, M[S, F.Data]);\\
&\qquad\qquad Enter(x, M[S, F.Data]);\\
&\qquad\qquad Enter(r, M[S, F.DS]);\\
&\qquad\qquad Enter(w, M[S, F.DS]);\\
&\qquad\qquad Enter(r, M[S, F.DACL]);\\
&\qquad\qquad Enter(r, M[S, F.SACL]);\\
&\qquad\qquad Enter(w, M[S, F.SACL]);\\
&\mathit{end.}
\end{aligned}
$$

При рассмотрении безопасности системы важным является вопрос организации журнала аудита системы. Если аудит доступа к объектам системы фиксирует каждый доступ со стандартными или специальными правами, то данная система монооперационная и является безопасной [2, 4]. Однако, как легко понять в ОС Windows, в журнале аудита могут фиксироваться и общие права доступа, т. е. система перестаёт быть монооперационной и вопрос о её безопасности остаётся открытым, так как для произвольной системы задача проверки безопасности алгоритмически не разрешима [2, 4].

2. Базисный подход

Для исследования безопасности механизма дискреционного разделения доступа в ОС Windows применим базисный подход к модели HRU, развитый в [1,6]. Для этого рассмотрим представление прав доступа на битовом уровне с помощью маски доступов [7, 9, 10]. Запрос субъекта на конкретный вид доступа к объекту преобразуется в маску доступа, которая сравнивается с масками разрешённых и запрещённых доступов в элементах частного списка контроля доступов (DACL) объекта.

Маска доступа, содержащаяся в элементе DACL, представляет собой значение длиной 32 бита. Первые 16 битов определяют специальные права доступа, биты с 16 до 23 — стандартные права доступа, бит 24 — право ACCESS_ SYSTEM_ SECURITY, бит 25 — право MAXIMUM_ ALLOWED (полный доступ), биты 26 и 27 зарезервированы для дальнейшего использования, биты с 28 по 31 определяют общие права доступа, отображаемые в специальные и стандартные права при попытке доступа к объекту.

Как было показано в [1, 6], в модели HRU можно перейти к новому базису, работающему с битовой строкой. Будем использовать следующие обозначения: M — матрица доступов (совокупность DACL всех объектов системы), $\mathbf{S}$ — множество субъектов компьютерной системы, $\mathbf{O}$ — множество объектов компьютерной системы, $M[S, O]$ — элемент матрицы доступов, определяющий права доступа субъекта $S \in \mathbf{S}$ к объекту $O \in \mathbf{O}$.

1. $AddO(O, M)$ — добавить столбец, соответствующий объекту O в матрицу доступов M.

2. $DelO(O, M)$ — удалить столбец, соответствующий объекту O в матрице доступов M.

3. $AddS(S, M)$ — добавить строку и столбец, соответствующие субъекту S в матрицу доступов M.

4. $DelS(S, M)$ — удалить строку и столбец, соответствующие субъекту S в матрице доступов M.

5. $Inv(M[S, O], k)$ — инвертировать в элементе матрицы доступов $M[S, O]$ бит с номером k. По сути эта операция выдает или отменяет право доступа a_k.

Выпишем команды для всех стандартных, специальных и общих прав доступа в данном базисе.

Специальные права доступа

Специальные права и их значения приведены в табл. 1.

Команды в базисе для специальных прав приведены ниже.

1. Чтение данных файла F:

$$
\begin{aligned}
&command \quad && File\_Read\_Special(F)\\
&&& if(b_0 == FALSE);\\
&&& Inv(F, 0);\\
&end.
\end{aligned}
$$

Таблица 1

Специальные права и их значения

Название права	*Значение*	*Описание права*
FILE_READ_DATA	0x00000001	Чтение данных
FILE_WRITE_DATA	0x00000002	Запись данных
FILE_APPEND_DATA	0x00000004	Добавление данных
FILE_READ_EA	0x00000008	Чтение расширенных атрибутов
FILE_WRITE_EA	0x00000010	Запись расширенных атрибутов
FILE_EXECUTE	0x00000020	Исполнение
FILE_DELETE	0x00000040	Удаление
FILE_READ_ATTRIBUTES	0x00000080	Чтение атрибутов
FILE_WRITE_ATTRIBUTES	0x00000100	Запись атрибутов

2. Запись данных в файл F:

$$\begin{aligned} command \quad & File\_Write\_Special(F) \\ & if(b_1 == FALSE); \\ & Inv(F,1); \\ end. & \end{aligned}$$

3. Добавление данных в файл F:

$$\begin{aligned} command \quad & File\_Uppend\_Special(F) \\ & if(b_2 == FALSE); \\ & Inv(F,2); \\ end. & \end{aligned}$$

4. Чтение расширенных атрибутов файла F:

$$\begin{aligned} command \quad & File\_Read\_EA\_Special(F) \\ & if(b_3 == FALSE); \\ & Inv(F,3); \\ end. & \end{aligned}$$

5. Запись расширенных атрибутов файла F:

$$\begin{aligned} command \quad & File\_Write\_EA\_Special(F) \\ & if(b_4 == FALSE); \\ & Inv(F,4); \\ end. & \end{aligned}$$

6. Запуск исполненяемого файла F:

$$\begin{aligned} command \quad & File\_Execute\_Special(F) \\ & if(b_5 == FALSE); \\ & Inv(F,5); \\ end. & \end{aligned}$$

7. Удаление файла F:

$$\begin{aligned} command \quad & File\_Delete\_Special(F) \\ & if(b_6 == FALSE); \\ & Inv(F, 6); \\ end. & \end{aligned}$$

8. Чтение атрибутов файла F:

$$\begin{aligned} command \quad & File\_Read\_Attributes\_Special(F) \\ & if(b_7 == FALSE); \\ & Inv(F, 7); \\ end. & \end{aligned}$$

9. Запись атрибутов файла F:

$$\begin{aligned} command \quad & File\_Write\_Attributes\_Special(F) \\ & if(b_8 == FALSE); \\ & Inv(F, 8); \\ end. & \end{aligned}$$

Стандартные права доступа
Стандартные права доступа приведены в табл. 2.

Таблица 2

Стандартные права доступа

Название права	*Значение*	*Описание права*
OBJECT_DELETE	0x00010000	Удаление объекта
OBJECT_READ_CONTROL	0x00020000	Чтение DS объекта
OBJECT_WRITE_DACL	0x00040000	Чтение DACL объекта
OBJECT_CHANGE_OWNERSHIP	0x00080000	Изменение права собственности объекта
OBJECT_SYNCHRONIZE	0x00100000	Использование объекта для синхронизации

Ниже приведены стандартные права, выраженные в базисе.
1. Удаление объекта F:

$$\begin{aligned} command \quad & Object\_Delete\_Standard(F) \\ & if(b_{16} == FALSE); \\ & Inv(F, 16); \\ end. & \end{aligned}$$

2. Чтение DS объекта F:

$$\begin{aligned} command \quad & Object\_Read\_Control\_Standard(F) \\ & if(b_{17} == FALSE); \\ & Inv(F, 17); \\ end. & \end{aligned}$$

3. Запись DACL объекта F:

$$\begin{aligned} command \quad & Object\_Write\_DACL\_Standard(F) \\ & if(b_{18} == FALSE); \\ & Inv(F, 18); \\ end. & \end{aligned}$$

4. Изменение прав владения объекта F:

$$\begin{aligned} command \quad & Object\_Change\_Ownership\_Standard(F) \\ & if(b_{19} == FALSE); \\ & Inv(F, 19); \\ end. & \end{aligned}$$

5. Изменение свойства синхронизации объекта F:

$$\begin{aligned} command \quad & Object\_Synchronize\_Standard(F) \\ & if(b_{20} == FALSE); \\ & Inv(F, 20); \\ end. & \end{aligned}$$

Общие права доступа

В табл. 3 приведены общие права доступа, ниже эти права представлены в базисе.

Таблица 3

Общие права доступа

Название права	*Значение*	*Описание права*
GENERIC_ALL	0x10000000	Общее право на полный доступ
GENERIC_EXECUTE	0x20000000	Общее право исполнения
GENERIC_WRITE	0x40000000	Общее право записи
GENERIC_READ	0x80000000	Общее право чтения

1. Добавление права на полный доступ на объект F:

$$\begin{aligned} command \quad & Generic\_All(F) \\ & if(b_{28} == FALSE); \\ & Inv(F, 28); \\ end. & \end{aligned}$$

2. Добавление права общего исполнения на объект F:

$$\begin{aligned} command \quad & Generic\_Execute(F) \\ & if(b_{29} == FALSE); \\ & Inv(F, 29); \\ end. & \end{aligned}$$

3. Добавление права общей записи на объект F:

$$\begin{aligned} command \quad & Generic\_Write(F) \\ & if(b_{30} == FALSE); \\ & Inv(F, 30); \\ end. & \end{aligned}$$

4. Добавление права общего чтения на объект F:

$$\begin{aligned} command \quad & Generic\_Read(F) \\ & if(b_{31} == FALSE); \\ & Inv(F, 31); \\ end. & \end{aligned}$$

Таким образом, набор прав доступа, описанный в спецификации ОС Windows, можно считать базисом, а сама ОС Windows, как видно из вышеприведённого, будет монооперационной системой в данном базисе.

3. Заключение

В качестве общего заключения можно сказать, что переход к новому базису в рамках модели HRU позволяет рассматривать широкий спектр реальных компьютерных систем. В частности, возможно рассмотрение дискреционного разделения доступа в подсистеме безопасности ОС Windows на основе модели HRU, а также рассмотрение ОС Windows как монооперационной в некотором базисе.

Литература

1. Бречка Д. М., Белим С. В. Исследование безопасности компьютерных систем в модели дискреционного разделения доступа HRU // Математические структуры и моделирование. 2009. Вып. 19. С. 97–103.
2. Гайдамакин Н. А. Разграничение доступа к информации в компьютерных системах. Екатеринбург: Изд-во Урал. ун-та, 2003.
3. Грушо А. А., Тимонина Е. Е. Теоретические основы защиты информации. М.: Яхтсмен, 1996.

4. Девянин П. Н. Модели безопасности компьютерных систем. М.: Академия, 2005.
5. Зегжда Д. П., Ивашко А. М. Основы безопасности информационных систем. М.: Горячая линия – Телеком, 2000.
6. Проблемы обработки и защиты информации: коллективная монография / под общ. ред. д-ра физ.-мат. наук С. В. Белима. Омск: КАН, 2010. Кн. 1: Модели политик безопасности компьютерных систем.
7. Руссинович М., Соломон Д. Внутреннее устройство Microsoft Windows: Windows Server 2003, Windows XP и Windows 2000. Мастер-класс: пер. с англ. 4-е изд. М.: Русская редакция, 2005.
8. Harrison M. A., Ruzzo W. L., Ullman J. D. Protection in Operating Systems // Communications of the ACM. 1975. P. 14–25.
9. Microsoft Corporation. Microsoft Windows XP Professional. Учебный курс MCSA/ MCSE: пер. с англ. 2-е изд., испр. М.: Русская редакция, 2003.
10. Microsoft Developer Network. URL: http://msdn.microsoft.com (дата обращения: 10.10.2010).