УДК 004.056.53

К. А. Щеглов, А. Ю. Щеглов

МОДЕЛЬ КОНТРОЛЯ ДОСТУПА К СОЗДАВАЕМЫМ ФАЙЛОВЫМ ОБЪЕКТАМ

Предложены метод контроля доступа к создаваемым файловым объектам, характеризующийся исключением сущности „объект доступа“ из разграничительной политики, и модель контроля доступа, с использованием которой разработаны требования к принудительному управлению потоками информации, обеспечивающие построение безопасной системы.

Ключевые слова: *защита информации, разграничение доступа, создаваемый файловый объект, контроль доступа, информационный поток.*

Введение. Основу защиты обрабатываемой компьютером информации от несанкционированного доступа составляет построение разграничительной политики доступа к файловым объектам, которую обеспечивает диспетчер доступа, перехватывающий и анализирующий все запросы от субъектов к объектам. Диспетчер однозначно выявляет в запросе доступа субъект и объект и на основании анализа заданного правила контроля принимает решение о предоставлении субъекту запрашиваемого доступа.

При назначении разграничений прав доступа первичен объект, поскольку именно он требует защиты от несанкционированного доступа. Однако применительно к решению задачи контроля доступа файловые объекты принципиально различаются — они могут быть подразделены на статичные (в первую очередь, системные) и создаваемые пользователями в процессе работы системы. Системные объекты присутствуют на момент настройки администратором прав доступа субъектов к объектам, а объекты второй группы еще не созданы. Возникает вопрос: как разграничивать доступ к еще не созданным объектам? А ведь эти объекты (прежде всего, файлы), особенно нуждаются в защите от несанкционированного доступа, поскольку содержат защищаемую конфиденциальную информацию.

Известные методы контроля доступа. Контроль доступа к статичным объектам осуществляется посредством назначения им атрибутов (прав доступа субъектов к объекту). Возможны два варианта назначения прав доступа к создаваемым объектам.

1. Права доступа задаются опосредованно, путем включения в систему на момент задания разграничительной политики доступа соответствующих статичных объектов — папок (своего рода „контейнеров“), доступ субъектов к которым, в том числе и по созданию в них новых объектов, разграничивается администратором. Вследствие разграничений пользователь может создавать новые файловые объекты (файлы) только в специально созданных папках. Создаваемые файловые объекты наследуют разграничения доступа от соответствующих включающих объектов (папок).

Как видим, собственно к создаваемым объектам контроль доступа не осуществляется, файл как сущность „исчезает“ из разграничительной политики доступа — используются субъект доступа и специально созданный контейнер — объект доступа, что определяет сложность задачи администрирования, многократно возрастающую, если в качестве субъекта доступа рассматривать не только учетную запись (пользователя), но и непосредственно процесс (приложение), без чего эффективную защиту в современных условиях не создать [1].

2. В разграничительной политике доступа используется сущность „Владения“, предоставляемая современными универсальными ОС: пользователь, создавший объект („Владелец“) наделяется полномочиями разграничивать права доступа к этому объекту для иных пользователей. В этом случае несмотря на то что права доступа в виде атрибутов назначаются именно

статичному объекту (владельцем, уже после создания им объекта), в определенном смысле можно говорить о контроле доступа к создаваемым файлам, поскольку права доступа на созданный файл назначаются пользователем-владельцем уже после задания администратором разграничительной политики доступа к объектам. Таким образом, изменяются собственно схема администрирования, способ задания разграничительной политики — реализуется непрерывное администрирование в процессе функционирования системы.

Однако функции администрирования при этом возлагаются уже не на администратора, а непосредственно на пользователя, что недопустимо в современных условиях.

Модель контроля доступа к статичным файловым объектам. Для оценки безопасности системы, основанной на контроле доступа к статичным файловым объектам, на практике используется модель Харрисона—Уззо—Ульмана [2]. Если считать, что $C = \{C_1,\ldots, C_l\}$ и $O = \{O_1,\ldots, O_k\}$ — соответственно линейно упорядоченные множества субъектов и объектов доступа, а $R = \{R_1,\ldots, R_m\}$ — конечное множество прав доступа (чтение, запись, удаление, исполнение и т.д.), то разграничительная политика доступа субъектов к объектам описывается матрицей доступа M, где M[C, O] — ячейка матрицы, которая содержит набор прав доступа субъекта из множества C к объекту из множества O. В любой момент времени система описывается текущим состоянием $Q = (C, O, \mathrm{M})$.

Требование к безопасности системы в рассматриваемом случае может быть сформулировано следующим образом: „Для заданной системы состояние $Q_0 = (C_0, O_0, \mathrm{M}_0)$ следует считать безопасным относительно некоторого права R, если не существует применимой к Q_0 последовательности действий, в результате выполнения которой субъектом C_0 приобретается право R доступа к объекту O_0, исходно отсутствующее в ячейке матрицы $\mathrm{M}_0[C_0, O_0]$“. Если право R, отсутствующее в ячейке матрицы $\mathrm{M}_0[C_0, O_0]$, приобретается субъектом C_0, то можно говорить, что произошла утечка права R, и относительно R его система небезопасна.

Поскольку последовательность действий генерирует субъект доступа, пользователь (действия администратора не имеет смысла рассматривать при анализе изменений состояния системы), то для рассматриваемой модели о безопасности системы можно говорить исключительно в предположении, что действия пользователя не приведут к утечке права R. Однако при реализации сущности „Владения“ именно пользователь, создавший объект, наделяет правом доступа R к этому объекту других пользователей. Как следствие, о безопасности системы в данном случае можно говорить исключительно в предположении, что субъект доступа (пользователь) не несет угрозы генерирования утечки права R с целью хищения, несанкционированной модификации или удаления обрабатываемой на предприятии информации. Однако в современных условиях, особенно применительно к корпоративным приложениям, где пользователь обрабатывает не собственную, а корпоративную информацию, и понятие „Владелец“ (не в технологическом смысле) к нему малоприменимо, нельзя сделать подобное предположение даже с большими оговорками. Например, в исследовании [3] сделан вывод о том, что наибольшую опасность для компаний сегодня представляет именно несанкционированный доступ собственных сотрудников к конфиденциальной информации предприятия.

Вывод. В безопасной системе контролем доступа должно осуществляться принудительное для пользователя управление потоками информации (или информационными потоками), сущность „Владения“ должна быть исключена из схемы контроля доступа.

Замечание. Перенос информации от C к O — информационный поток записи, от O к C — информационный поток чтения. Управление потоками — предоставление/изменение права R генерирования потока информации между C и O.

Правила управления потоками информации. Предлагаемый метод контроля доступа к создаваемым файловым объектам базируется на реализации принципов, изложенных в статье [4]. „Объект“ исключается из схемы реализации разграничительной политики — используются две сущности: идентификатор (учетная информация) субъекта, создавшего объект, и

идентификатор субъекта, запрашивающего доступ к созданному объекту. Новым файлом, созданным субъектом, наследуется учетная информация этого субъекта доступа. Учетная информация субъекта доступа, в общем случае используемая при реализации разграничительной политики, определяется следующими сущностями: исходный идентификатор пользователя (учетная запись, под которой осуществлен вход в систему); „полнопутевое“ имя исполняемого файла процесса, запрашивающего доступ к ресурсу; эффективный идентификатор пользователя (учетная запись, от которой осуществлен запрос доступа к ресурсу) [1].

При запросе доступа к любому файлу диспетчер доступа анализирует наличие, а при наличии — содержимое унаследованной файлом учетной информации создавшего его субъекта доступа. Матрица доступа здесь приобретает совершенно иной вид, поскольку из разграничительной политики доступа исключена сущность „Объект“. Если считать, что $C = \{C_1, \ldots, C_l\}$ — линейно упорядоченное множество субъектов доступа, а $R = \{R_1, \ldots, R_m\}$ — конечное множество прав доступа (r — чтение, w — запись, d — удаление, x — исполнение и т.д., 0 — отсутствие прав доступа) субъекта C_i к объекту, созданному субъектом C_j (i=1, …, l, j=1, …, l) то матрица доступа M, используемая для реализации разграничительной политики методом контроля доступа с принудительным управлением потоками информации имеет следующий вид (условимся в строках матрицы указывать учетную информацию субъектов, запрашивающих доступ к объектам, а в столбцах — учетную информацию субъектов, унаследованную созданными объектами):

$$\mathrm{M} = \begin{array}{c} \\ C_1 \\ C_2 \\ . \\ . \\ . \\ C_{l-1} \\ C_l \end{array} \begin{array}{c} \begin{array}{ccc} C_1 & C_2 & C_l \end{array} \\ \begin{bmatrix} r,w,d & w & 0 \\ r & r,w,d & 0 \\ & & \\ & ..... & \\ & ..... & \\ 0 & 0 & r \\ 0 & w & r,w,d \end{bmatrix} \end{array}.$$

В любой момент времени система описывается своим текущим состоянием $Q = (C,C,\mathrm{M})$, $\mathrm{M}[C,C]$ — ячейка матрицы, которая содержит набор прав доступа. Разрешение права доступа субъекта C_j к объектам, созданным субъектом C_i (i=1, …, l; j=1, …, l, где $R = \{x,w,R,d\}$), обозначим как $C_j(R)C_i$.

Вывод. Используя предложенный метод контроля доступа к создаваемым файловым объектам, можно построить безопасную систему, поскольку за счет контроля доступа осуществляется принудительное для пользователя управление потоками информации.

Сформулируем основные правила (требования) управления потоками, при реализации которых в системе отсутствует утечка права $R = \{x,w,r,d\}$, что не приводит к несанкционированному обмену информацией между субъектами, который будем обозначать $C_j[R]C_i$, $j \neq i$ (субъект C_j получает несанкционированный доступ к информации, обрабатываемой субъектом C_i):

1. Недопустимо разрешение пользователям права исполнения (x) файла, созданного в процессе функционирования системы. При выполнении данного требования система безопасна относительно x. Заметим, что выполнение именно этого требования обеспечивает эффективную защиту от вредоносных программ [5]. Как следствие, разрешенные права доступа к создаваемым файловым объектам: $R = \{w,r,d\}$.

2. При назначении разграничительной политики „по умолчанию“ должны быть установлены права доступа: $C_i(w,r,d)C_i$ (i=1, …, l). Данное правило обусловливает задание диаго-

нальной („канонической“ [1]) матрицы доступа, характеризуемой условием: $C_i(w,r,d)C_i$; $C_i(0)C_j$ ($i\neq j$, i=1, …, l, j=1, …, l). Реализующая каноническую модель доступа система безопасна относительно прав записи (w) и чтения (r).

3. При расширении канонической матрицы доступа правом чтения $C_i(r)C_j$ и при уже разрешенном в матрице праве чтения $C_k(r)C_i$, одновременно должно также разрешаться право чтения (r): $C_k(r)C_j$ ($i\neq j\neq k$, i=1, …, l; j=1, …, l; k=1, …, l), что предотвращает возможность несанкционированного обмена информацией между субъектами $C_k[R]C_j$, из-за возникающей при этом утечки права чтения $C_k(r)C_j$.

4. При расширении канонической матрицы доступа правом записи (w): $C_i(w)C_j$, при уже разрешенном в матрице праве записи (w): $C_j(w)C_k$, одновременно с этим должно также разрешаться право записи (w): $C_i(w)C_k$ ($i\neq j\neq k$, i=1, …, l; j=1, …, l; k=1, …, l), что предотвращает возможность несанкционированного обмена информацией между субъектами $C_i[R]C_k$, из-за возникающей при этом утечки права записи (w): $C_i(w)C_k$.

5. При расширении канонической матрицы доступа правом удаления (d) включающих объектов (папок, в которых создаются файлы), должно реализовываться следующее правило управления: любой включающий объект (папка) может быть удален любым субъектом при условии отсутствия включенных в него объектов (в первую очередь — файлов).

Замечание. Поскольку предлагаемый метод контроля доступа не предполагает реализации разграничительной политики доступа к статичным объектам — папкам — любой субъект может удалить любую папку, соответственно и все включенные в нее файлы, что приводит к утечке права удаления (d).

Таким образом, предложенный метод относится к дискреционным с принудительным управлением потоками информации. При реализации сформулированных требований метод позволяет построить безопасную систему.

Заключение. Использование предложенного метода контроля доступа к создаваемым файловым объектам позволяет пересмотреть принципы построения разграничительной политики доступа к защищаемым ресурсам, разделив задачи защиты статичных и создаваемых файловых объектов и способы их решения.

СПИСОК ЛИТЕРАТУРЫ

1. *Щеглов А. Ю.* Защита компьютерной информации от несанкционированного доступа. СПб: Наука и техника, 2004. 384 с.

2. *Harrison M., Ruzzo W., Ullman J.* Protection in operating systems // Communication of ACM. 1976. Vol. 19, N 8. P. 461—472.

3. [Электронный ресурс]: <http://www.securitycode.ru/company/news/SC-analytic-2011>.

4. *Щеглов К. А.* Принципы контроля доступа к создаваемым файловым объектам // Сб. тр. молодых ученых и сотрудников кафедры ВТ. СПб: НИУ ИТМО, 2012. Вып. 3. С. 85—86.

5. *Щеглов К. А., Щеглов А. Ю.* Защита от вредоносных программ методом контроля доступа к создаваемым файловым объектам // Вестн. компьютерных и информационных технологий. 2012. № 8. С. 46—51.

Сведения об авторах

Константин Андреевич Щеглов — студент; Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, кафедра вычислительной техники; E-mail: schegl_70@mail.ru

Андрей Юрьевич Щеглов — д-р техн. наук, профессор; Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, кафедра вычислительной техники; E-mail: info@npp-itb.spb.ru

Рекомендована кафедрой вычислительной техники

Поступила в редакцию 08.02.12 г.