ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И СИСТЕМЫ

УДК 004.056.53
DOI: 10.17586/0021-3454-2015-58-3-157-166

НОВЫЙ ПОДХОД К ЗАЩИТЕ ДАННЫХ В ИНФОРМАЦИОННОЙ СИСТЕМЕ

К. А. Щеглов, А. Ю. Щеглов

Университет ИТМО, 197101, Санкт-Петербург, Россия
E-mail: info@npp-itb.spb.ru

Рассмотрен новый подход к защите обрабатываемых в информационной системе данных, основанный на реализации методов контроля и разграничения прав доступа к создаваемым объектам — файловым объектам и буферу обмена, что позволяет исключить объект доступа из разграничительной политики за счет автоматической разметки создаваемых объектов. Практическая реализация предложенного подхода проиллюстрирована на примере запатентованного и апробированного технического решения. Показано, что поскольку использование предложенного подхода принципиально меняет требования к реализации других механизмов защиты, решающих совсем иные задачи, чем контроль и разграничение прав доступа, совокупность полученных решений образует новую технологию защиты данных, обрабатываемых в информационной системе.

Ключевые слова: *информационная система, защита данных, несанкционированный доступ, контроль и разграничение прав доступа, разграничительная политика, создаваемый объект.*

Введение. Решение задачи защиты информации от несанкционированного доступа в любой информационной системе основано на контроле и разграничении прав доступа (далее — контроль доступа) субъектов к объектам — защищаемым ресурсам. Целью контроля доступа является ограничение производимых в системе действий, которые потенциально могут нанести вред. На практике используется одна из абстрактных моделей дискреционного, мандатного и ролевого контроля доступа [1, 2].

Дискреционный (называемый также избирательным) контроль доступа (Discretionary Access Control — DAC) основан на реализации модели Харрисона — Руззо — Ульмана [3]. В основе построения разграничительной политики доступа лежит задание администратором матрицы доступа — списка правил доступа к объектам субъектов. Дискреционный метод контроля доступа предусматривает для пользователей либо произвольный, либо принудительный способ управления потоками информации [4]. При произвольном управлении предполагается, что владелец объекта (пользователь, создавший объект) может задавать правила доступа к нему — назначать, кто имеет доступ к объекту и вид доступа. Этот метод широко используется в современных универсальных операционных системах (ОС). Альтернативным решением является исключение сущности владения из схемы управления с возложением всех задач настройки разграничительной политики доступа на выделенного субъекта — администратора.

Мандатный контроль доступа (Mandatory Access Control — MAC) основан на абстрактной модели Белла — ЛаПадулы [5]. Каждому субъекту и объекту системы назначается некоторый уровень безопасности — присваивается метка безопасности. Контроль доступа состоит

в арифметическом сравнении меток субъекта и объекта при анализе запроса доступа. Субъекту предоставляется доступ к объекту, если выполняется заданное отношение — формализованное правило, предполагающее арифметическое сравнение меток, соответствующих уровням безопасности объекта и запросившего к нему доступ субъекта. Эта модель имеет ряд ограничений по практическому использованию. Во-первых, разграничительная политика доступа реализуется не между отдельными субъектами-пользователями, а между группами субъектов, характеризуемых одним уровнем безопасности (в общем случае требуется еще и разграничивать права доступа к объектам между субъектами, отнесенными к одному уровню безопасности); во-вторых, подобной классификацией сложно воспользоваться для системных субъектов и объектов, прежде всего исполняемых. Поэтому в критичных приложениях требуется применять одновременно дискреционный и мандатный методы, что и регламентируется действующим сегодня нормативным документом в области информационной безопасности [6]. Этим же документом регламентируется применение в критичных приложениях принудительного управления потоками информации, что крайне важно, поскольку санкционированный пользователь должен рассматриваться в современных условиях как потенциально возможный нарушитель безопасности [4].

В последнее время активно развивается ролевая модель контроля доступа (Role-Based Access Control — RBAC) [1], основанная на максимальном приближении логики работы системы к реальному разделению функций персонала в организации. Применение данного метода подразумевает определение ролей в системе как совокупность действий и обязанностей, связанных с видом деятельности. Вместо того, чтобы указывать права доступа к каждому объекту для каждого пользователя, требуется реализовать разграничительную политику доступа для ролей, с которыми соотносятся пользователи. На самом деле ролевая модель — это не что иное, как дискреционный контроль доступа при соответствующей групповой политике (разграничительной политике для групп пользователей). К достоинствам данной модели можно отнести возможность формализации ролей, т.е. возможность задания и последующего тиражирования неких типовых разграничительных политик доступа для соответствующих ролей, которые в системе могут задаваться „по умолчанию“ (поставляться вместе с системой). Подобное решение реализовано в системе SELinux (Security-Enhanced Linux) — Linux, обладающей повышенной безопасностью.

Существующие абстрактные модели контроля доступа и реализующие их методы предусматривают использование в разграничительной политике доступа двух равноправных сущностей — субъекта и объекта доступа, а назначением правил задаются права доступа субъектов к объектам (и наоборот). При этом в качестве субъектов доступа в разграничительной политике выступают пользователи, действия которых потенциально могут нанести вред.

Остановимся на ключевом недостатке известных подходов к реализации контроля доступа применительно к защите данных, обрабатываемых в информационных системах. Файлы в системе принципиально различаются своим функциональным назначением. Они могут быть разделены на статичные (системные) и создаваемые пользователями в процессе работы. Принципиальная разница между этими группами файловых объектов при задании разграничительной политики доступа к ним состоит в том, что статичные объекты присутствуют в системе на момент назначения администратором правил доступа субъектов к объектам, а создаваемых объектов еще нет. Возникает вопрос: как же администратору при принудительном управлении потоками информации разграничивать права доступа субъектов к объектам, если они еще не созданы. Это противоречие иллюстрирует всю нелогичность известной схемы контроля доступа при принудительном управлении потоками информации.

В современных условиях процесс (приложение) несет в себе не меньшую, если не большую, угрозу несанкционированного доступа к обрабатываемой информации, чем пользо-

ватель [7, 8]. Как следствие, равноправными сущностями, определяющими субъект доступа в разграничительной политике, должны выступать как пользователь (учетная запись), так и процесс (полнопутевое имя исполняемого файла процесса). Для защиты от обхода в современной разграничительной политике доступа имеет смысл идентифицировать субъект тремя сущностями [9]:

— „исходный идентификатор пользователя“ (учетная запись, под которой запущен процесс);

— „эффективный идентификатор пользователя“ (учетная запись, под которой процесс, соответствующий поток, запрашивает доступ к объекту);

— „процесс“.

В работе [10] предложен метод контроля доступа, в значительной мере упрощающий эту задачу за счет назначения прав доступа субъектов к объектам (а не наоборот). Упрощение задачи управления достигается использованием масок при задании субъектов и объектов доступа в разграничительной политике, что позволяет обеспечить новые свойства защиты [11]. Технические решения для различных способов идентификации субъекта доступа [12, 13] реализованы и апробированы. Однако их целесообразно применять для реализации контроля доступа к статичным объектам — ресурсам, присутствующим в системе на момент задания администратором разграничительной политики доступа (например, к системным файлам, объектам реестра ОС, сетевым объектам, внешним накопителям и т.д. [10]).

Принципы контроля доступа к создаваемым объектам основаны на их автоматической разметке при создании или модификации [14, 15]. Это позволяет исключить сущность „объект доступа“ из разграничительной политики доступа. Правила доступа устанавливаются между сущностями „субъект доступа (учетная информация), запрашивающий доступ к объекту“ и „субъект доступа, создавший этот объект“. При создании (модификации) объекта субъектом объект наследует учетную информацию данного субъекта — реализуется разметка объекта (учетная информация субъекта сохраняется в атрибутах созданного им объекта).

При запросе доступа к любому объекту диспетчер доступа (решающий элемент) получает разметку этого объекта, считывает его атрибуты, анализирует запрос на соответствие заданным правилам доступа и предоставляет запрошенный субъектом доступ к объекту либо отказывает в нем.

Таким образом, реализуется разграничительная политика (задаются правила доступа) не для субъектов к объектам, а между субъектами доступа к создаваемым ими объектам. В этом случае реализуется доступ субъектов к объектам, но в разграничительной политике объекты отсутствуют — присутствуют только субъекты доступа.

Рассмотрим реализацию контроля доступа к файловым объектам и к буферу обмена как к создаваемым объектам, используемым в системе для хранения данных.

Методы контроля доступа к создаваемым файлам. *Мандатный метод контроля доступа* к создаваемым файлам предполагает задание уровней безопасности (уровней доступа или меток безопасности), мандатов исключительно пользователям (интерактивным пользователям). Назначения меток безопасности объектам доступа не требуется [16].

Рассмотрим работу диспетчера доступа.

Метки безопасности назначаются тем контролируемым пользователям, которые создают файлы, требующие разграничения права доступа. При создании файла любым пользователем (не только тем, которому назначена метка безопасности) файл автоматически размечается диспетчером доступа: в атрибуты создаваемого файла автоматически помещаются учетные данные субъекта (в данном случае его уровень безопасности), создавшего этот файл. Подобным образом будет размечен при модификации и неразмеченный ранее файл.

Размечать все создаваемые файлы требуется для защиты от запуска вредоносных программ. Эффективная защита от такой угрозы реализуется, когда исполнение создаваемых в процессе работы системы файлов, в том числе и с системными правами, запрещено [17].

Запрашиваемый доступ к неразмеченному файлу будет разрешен, при этом в случае модификации файла он будет автоматически размечен. Если запрошен доступ на исполнение к размеченному файлу, запрос доступа отклоняется. При запросе доступа к файлу иного типа определяется, имеет ли метку безопасности пользователь, запросивший доступ к этому файлу. При отсутствии метки запрос доступа отклоняется диспетчером. Если метка безопасности есть, диспетчер анализирует соответствие запроса мандатным правилам доступа, арифметически сравнивая соответствующие метки (мандаты). Для этого диспетчер определяет и сравнивает (по соответствующим учетным записям) мандаты, а именно, числовые значения назначенных уровней доступа пользователя, запросившего доступ к размеченному файлу, и пользователя, создавшего этот файл. Запрошенный доступ либо разрешается (если запрос не противоречит заданным правилам мандатного контроля), либо отклоняется. Таким образом, достигается принципиальное упрощение задачи управления средством защиты.

Рассмотренный метод обеспечивает, кроме того, корректное решение задачи контроля и разграничений прав доступа в общем случае. В процессе работы пользователями создаются и модифицируются файлы в различных каталогах, но существуют папки коллективного доступа, к которым необходимо разрешить полный доступ всем пользователям. Для корректной реализации схемы контроля доступа известными методами эти файлы (либо папки) должны быть выявлены и им не должны присваиваться метки безопасности, иначе не обеспечить коллективный доступ, что противоречит самой идее мандатной схемы контроля доступа [16]. При реализации же контроля доступа к создаваемым файлам любой файл, создаваемый в любой папке, будет принудительно размечаться и права доступа к нему будут разграничиваться.

Корректная разграничительная политика доступа, построенная на основе мандатного метода контроля доступа к категоризованной по уровням конфиденциальности информации, использует иерархические метки безопасности, которые сравниваются исключительно на идентичность [18]. Это используется и при реализации сессионного контроля доступа [19], корректное построение которого определяется условием задания режимов обработки информации различных уровней конфиденциальности для различных учетных записей пользователей. При реализации сессионного контроля доступа упрощение задачи администрирования мандатного контроля доступа приобретает первостепенное значение.

Дискреционный метод контроля доступа. При дискреционном методе контроля задача упрощения управления системой защиты еще более актуальна, поскольку субъект доступа должен идентифицироваться тремя вышеназванными сущностями. Для пояснения реализации метода воспользуемся интерфейсами настройки разработанного средства защиты, реализующего дискреционный метод контроля доступа к создаваемым файлам. Интерфейс создания и отображения заданных субъектов доступа в системе защиты представлен на рис. 1 [20].

При задании идентификатора пользователя, как исходного (первичного), так и эффективного, можно использовать маску „*“ — „Любой“ (заданные правила распространяются на всех пользователей). Имя процесса может быть задано либо полнопутевым именем исполняемого файла, либо маской (возможно также использование переменных среды окружения). Например, маской C:\ProgramFile* определяются все исполняемые файлы из обозначенного каталога, маской „*“ задается применимость правила к любому процессу. Поскольку один и тот же реальный субъект доступа в разграничительной политике может определяться одновременно несколькими масками, то при анализе запроса доступа диспетчер выбирает из разграничительной политики доступа правило для анализа, наиболее точно соответствующего описанию субъекта доступа.

Правила доступа задаются администратором из интерфейса, представленного на рис. 2 (субъекты доступа здесь отображаются присвоенными им при создании именами, см. рис. 1).

Отметим, что в назначаемые права доступа (см. рис. 2) не внесено право „исполнение“, так как запрет исполнения по умолчанию должен быть установлен для всех создаваемых файлов.

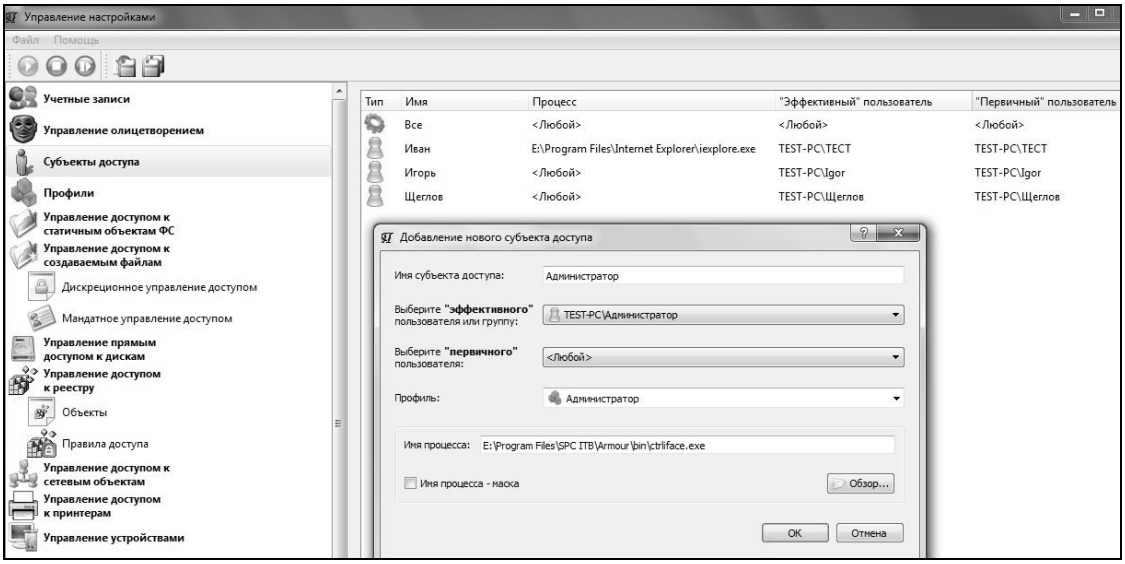

Рис. 1

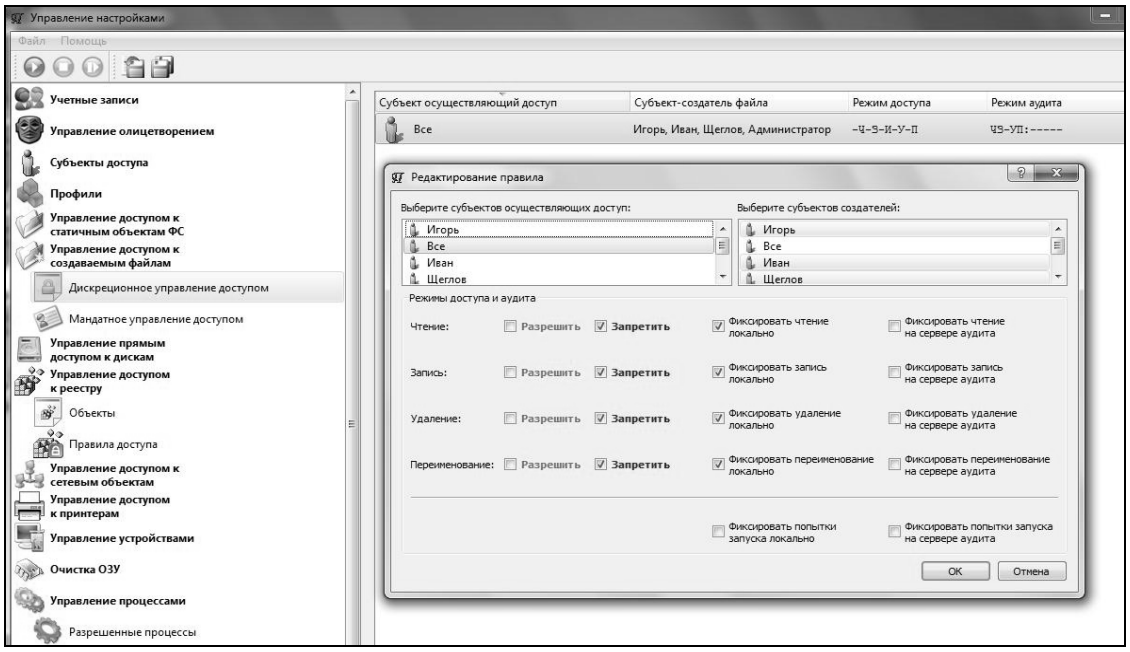

Рис. 2

Задание разграничительной политики доступа осуществляется следующим образом. Из списка заданных субъектов, отображаемого в интерфейсе настройки правил доступа (см. рис. 2), в поле „Выберите субъектов-создателей“ выбираются субъекты доступа, которые создают файлы с ограниченными правами доступа к ним других субъектов. Для выбранного в данном поле контролируемого субъекта-создателя файла назначаются права доступа к создаваемым им файлам других субъектов. Субъект, которому назначаются права доступа, выбирается в поле „Выберите субъектов, осуществляющих доступ“. Для выбранной пары субъектов (в левом и правом полях интерфейса) разрешаются либо запрещаются отдельные права доступа (чтение, запись, удаление, переименование). Требования к правилам доступа, выполнение

которых позволяет построить безопасную систему защиты от утечки прав доступа, сформулированы и обоснованы на основе абстрактной модели Харрисона — Руззо — Ульмана [20].

При реализации разграничительной политики доступа к создаваемым файлам мандатный и дискреционный механизмы контроля доступа могут использоваться совместно. При этом запрос доступа будет считаться санкционированным в случае, если он не будет противоречить ни мандатным, ни дискреционным правилам доступа. При этом диспетчер доступа сначала анализирует мандатные правила доступа, затем дискреционные.

Дискреционный метод контроля доступа к создаваемым файлам позволяет решать задачи защиты информации от атак, направленных на использование уязвимостей приложений, — в подобных разграничительных политиках доступа именно приложение рассматривается в качестве наиболее вероятного источника угрозы несанкционированного доступа. Любые приложения и любые их группы можно изолировать друг от друга по используемым данным [21]. Например, используя всего несколько правил в разграничительной политике доступа, можно изолировать работу интернет-браузера (либо иных по каким-либо соображениям критичных приложений), предотвратив его доступ к данным, обрабатываемым иными приложениями.

Проиллюстрируем, насколько меняется модель контроля доступа при реализации рассмотренных методов контроля доступа к создаваемым объектам.

Модель контроля доступа. Рассмотрим абстрактную модель Харрисона — Руззо — Ульмана [3]. Если считать, что $C=\{C_1, \ldots, C_l\}$ и $O=\{O_1, \ldots, O_k\}$ — линейно упорядоченные множества субъектов и объектов доступа соответственно, а $R=\{w, r, x, d\}$ — конечное множество прав доступа (запись, чтение, исполнение, удаление), то разграничительная политика доступа субъектов к объектам описывается матрицей доступа M; $M[C, O]$ — ячейка матрицы, которая содержит набор прав доступа субъекта из множества $C=\{C_1, \ldots, C_l\}$ к объекту из множества $O=\{O_1, \ldots, O_k\}$. В любой момент времени система описывается своим текущим состоянием $Q=(C, O, M)$. Разграничительная политика доступа субъектов к объектам отображается матрицей M:

$$M = \begin{array}{c} \\ C_1 \\ C_2 \\ \vdots \\ C_{l-1} \\ C_l \end{array} \begin{array}{c} \begin{array}{ccc} O_1 & O_2 & O_k \end{array} \\ \begin{bmatrix} r,w,d & w & 0 \\ r & r,w,d & 0 \\ \ldots & \ldots & \ldots \\ 0 & 0 & r \\ 0 & w & r,w,d \end{bmatrix} \end{array}.$$

В этой модели требование к построению безопасной системы формулируется следующим образом: для заданной системы состояние $Q_0 = (C_0, O_0, M_0)$ следует считать безопасным относительно некоторого права R, если не существует применимой к Q_0 последовательности действий, в результате выполнения которых субъектом C_0 приобретается право R доступа к объекту O_0, исходно отсутствующее в ячейке матрицы $M_0[C_0, O_0]$. Если же право R, отсутствующее в ячейке матрицы $M_0[C_0, O_0]$, приобретается субъектом C_0, то считается, что произошла утечка права R, а система небезопасна относительно права R.

При реализации принципов контроля доступа к создаваемым объектам, как установлено выше, объект доступа исключается из разграничительной политики и, как следствие, из матрицы доступа M. Абстрактная модель контроля доступа в этом случае принимает иной вид.

Если считать, что $C=\{C_1, \ldots, C_l\}$ — линейно упорядоченное множество субъектов доступа, а $R = \{w, r, x, d\}$ — конечное множество прав доступа (запись, чтение, исполнение, удаление) субъекта C_i к объекту, созданному субъектом C_j, i=1, …, l, j=1, …, l, то матрица доступа M имеет следующий вид (условимся в строках матрицы указывать учетную информацию

субъектов, запрашивающих доступ к объектам, а в столбцах — учетную информацию субъектов, унаследованную созданными объектами, к которым запрашивается доступ):

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

Как видим, в любой момент времени система описывается своим текущим состоянием $Q = (C, C, M)$, где $M[C, C]$ — ячейка матрицы, содержащая набор прав доступа.

Реализация предложенных решений позволяет принципиально пересмотреть подходы к построению систем обнаружения (и предотвращения) вторжений уровня хоста, основанных на мониторинге (по журналам аудита) действий приложений. Вторжение обнаруживается и может быть предотвращено, предотвращаются атаки, следующие за вторжением (более полно эти вопросы рассмотрены в работе [21]), причем все это осуществляется в реальном времени.

Метод контроля доступа к буферу обмена. Буфер обмена предназначен для временного хранения данных, используемых для обмена между приложениями. Поскольку на момент задания администратором разграничительной политики доступа эти данные еще не созданы, то можно говорить о контроле и разграничении прав доступа к создаваемым объектам (к данным, временно сохраняемым в буфере обмена). Поэтому в данном случае имеет смысл применить предложенные принципы контроля и разграничения прав доступа. С учетом же назначения буфера обмена основным субъектом доступа в разграничительной политике должен выступать процесс.

Субъекты доступа задаются из интерфейса, представленного на рис. 1, а правила доступа субъектов к буферу обмена — из интерфейса, аналогичного приведенному на рис. 2, но используется лишь одно правило — разрешение/запрет субъекту на получение информации из буфера обмена, помещенной в буфер обмена другим субъектом.

Такой механизм защиты, совместно с механизмом контроля доступа к создаваемым файлам, позволяет реализовать полностью изолированную обработку данных отдельными приложениями (группами приложений) в информационной системе.

Изменение технологии защиты данных в информационной системе. Рассмотрим, как практическое использование методов контроля и разграничения прав доступа к создаваемым объектам сказывается на технологии защиты данных в информационной системе на примере решения задач защиты данных — задач гарантированного удаления и шифрования файлов.

Далеко не все данные, записанные на жестком диске или на внешнем накопителе, образуют файлы. Как правило, на диске присутствует так называемая остаточная информация. Дело в том, что при удалении файла штатными средствами ОС собственно данные не удаляются, а осуществляется переразметка MFT-таблицы (Windows). Эти данные невозможно прочитать, обратившись к файлу (они не образуют файла), но достаточно просто получить к ним доступ с использованием сторонних программ прямого доступа к диску.

Задача предотвращения появления на накопителе обрабатываемых данных в виде остаточной информации — это отдельная важная задача защиты информации, решаемая механизмом гарантированного удаления файлов. Запрос на удаление файла перехватывается средством защиты, и в удаляемый файл записывается заданное число раз исходно заданная администратором информация, после чего управление передается системе для „удаления“ штатными средствами ОС. В результате в качестве остаточной информации на накопителе будут

скапливаться данные, которые средством защиты принудительно записываются в файл перед его удалением.

При реализации методов контроля доступа к файлам правила гарантированного удаления должны устанавливаться в отношении конкретных объектов доступа — файловых объектов, в которых предполагается сохранение пользователями конфиденциальной информации.

Использование подобного решения связано, опять же, с вопросами корректности и сложности администрирования. Дело в том, что гарантированно удалять необходимо файлы не только из папок, предназначенных для хранения файлов с конфиденциальными данными, но и из всех временных файлов, которые создаются большинством приложений. Однако при удалении временного файла системой данные также будут храниться в виде остаточной информации на диске. Это многократно усложняет настройку подобного механизма защиты и ставит под сомнение возможность корректного решения задачи такой защиты в принципе.

Рассмотрим, как изменится реализация механизма защиты, в случае если система защиты основана на методе контроля доступа к создаваемым файлам. Описанное выше гарантированное удаление файлов здесь не может быть применимо, так как любой файл любым субъектом может быть создан в любом объекте (в любой папке), что априори не позволяет исходно задать правила гарантированного удаления путем задания объектов.

Однако созданный файл однозначно описывается разметкой. Это позволяет реализовать метод гарантированного удаления, основанный на автоматической разметке файлов [22]. Настройкой средства защиты создаются правила гарантированного удаления — задаются субъекты, идентифицируемые своими именами либо уровнями доступа (при мандатном контроле), и файлы, созданные ими, должны гарантированно удаляться. При запросе на удаление любого файла средством защиты считывается разметка файла, анализируются заданные правила и принимается решение о необходимости гарантированного удаления этого файла. Как видим, решение задачи реализуется простейшими настройками соответствующего механизма защиты и корректно в общем случае — где бы контролируемым субъектом не был создан файл, он будет гарантированно удален.

Все сказанное относится и к решению задачи автоматического шифрования создаваемых файлов. Именно эти файлы, используемые для хранения обрабатываемых данных, и требуется хранить в зашифрованном виде.

Очевидно, что при использовании в системе методов, реализующих разграничительную политику доступа субъектов к объектам, администратору необходимо задавать файловые объекты, включая файловые накопители данных, сохраняемых субъектом, в которых эти данные будут автоматически зашифровываться. Это трудоемкая задача, а ошибка управления здесь крайне критична, так как может привести к утечке конфиденциальной информации. Чтобы предотвратить подобные потенциальные „каналы“ утечки, администратору необходимо выявить все подобные файлы и задать применительно к ним режим записи с шифрованием.

В случае использования метода (мандатного или дискреционного либо обоих одновременно) контроля доступа к создаваемым файлам в системе защиты может быть решена задача принудительного для субъекта доступа хранения информации в зашифрованном виде. При этом для настройки правил шифрования файлов потребуется задавать не объекты доступа — папки, в которых сохраняемые данные будут зашифровываться, а субъекты доступа (при мандатном контроле — уровни доступа или метки безопасности), созданные которыми данные, где бы они не сохранялись, будут автоматически зашифрованы. Учетной же информации субъекта, сохраняемой в качестве атрибута создаваемого (модифицируемого) файла в незашифрованном виде (она не является секретной информацией), вполне достаточно, чтобы выбрать ключ для расшифровывания файла, где бы (в какой бы папке) он не был создан [23]. Таким образом, получено принципиальное упрощение задачи администрирования и корректное решение задачи защиты в общем случае.

Заключение. Практическое применение предложенных методов контроля доступа к создаваемым объектам позволяет защиту данных и системных объектов представить как совершенно различные задачи защиты, что принципиально меняет требования к их решению в дополнение к реализации разграничительной политики доступа к файловым объектам. Это позволяет говорить именно о новой технологии (как о совокупности взаимосвязанных методов) защиты данных, обрабатываемых в информационной системе.

СПИСОК ЛИТЕРАТУРЫ

1. *Девянин П. Н.* Модели безопасности компьютерных систем. М.: Изд. центр „Академия“, 2005.
2. *Цирлов В. Л.* Основы информационной безопасности автоматизированных систем. Ростов-на-Дону: Феникс, 2008.
3. *Harrison M., Ruzzo W., Ullman J.* Protection in operating systems // Communication of ACM. 1976.
4. *Щеглов А. Ю.* Защита компьютерной информации от несанкционированного доступа. СПб: Наука и техника, 2004.
5. *Bell D. E., LaPadula L. J.* Secure computer systems: unified exposition and multics interpretation // Tech. Rep. MTR-2997, Vol. 4 — Mitri Corp., Bedford, MA, 1976.
6. Гостехкомиссия России. Руководящий документ. Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от НСД к информации. М., 1992.
7. *Щеглов К. А., Щеглов А. Ю.* Защита от атак со стороны приложений, наделяемых вредоносными функциями. Модели контроля доступа // Вопросы защиты информации. 2012. Вып. 99, № 4. С. 31—36.
8. *Щеглов К. А., Щеглов А. Ю.* Защита от атак на уязвимости приложений. Модели контроля доступа // Вопросы защиты информации. 2013. Вып. 101, № 2. С. 36—43.
9. *Щеглов К. А., Щеглов А. Ю.* Методы идентификации и аутентификации пользователя при доступе к файловым объектам // Вестник компьютерных и информационных технологий. 2012. № 10. С. 47—51.
10. *Щеглов К. А., Щеглов А. Ю.* Контроль доступа к статичным файловым объектам // Вопросы защиты информации. 2012. Вып. 97, № 2. С. 12—20.
11. *Маркина Т. А., Щеглов А. Ю.* Метод защиты от атак типа drive-by загрузка // Изв. вузов. Приборостроение. 2014. Т. 57, № 4. С. 15—20.
12. Пат. 2534599 РФ. Система контроля доступа к ресурсам компьютерной системы с субъектом доступа „пользователь, процесс“ / *А. Ю. Щеглов, К. А. Щеглов.* Приор. 30.04.2013.
13. Пат. 2534488 РФ. Система контроля доступа к ресурсам компьютерной системы с субъектом „исходный пользователь, эффективный пользователь, процесс“ / *А. Ю. Щеглов, К. А. Щеглов.* Приор. 18.06.2013.
14. *Щеглов К. А., Щеглов А. Ю.* Принцип и методы контроля доступа к создаваемым файловым объектам // Вестник компьютерных и информационных технологий. 2012. № 7. С. 43—47.
15. Пат. 2524566 РФ. Система контроля доступа к файлам на основе их автоматической разметки / *А. Ю. Щеглов, К. А. Щеглов.* Приор. 18.03.2013.
16. *Щеглов К. А., Щеглов А. Ю.* Реализация метода мандатного доступа к создаваемым файловым объектам // Вопросы защиты информации. 2013. Вып. 103, № 4. С. 16—20.
17. *Щеглов К. А., Щеглов А. Ю.* Защита от вредоносных программ методом контроля доступа к создаваемым файловым объектам // Вестник компьютерных и информационных технологий. 2012. № 8. С. 46—51.
18. *Щеглов К. А., Щеглов А. Ю.* Модели и правила мандатного контроля доступа // Вестник компьютерных и информационных технологий. 2014. № 5. С. 44—49.
19. *Щеглов К. А., Щеглов А. Ю.* Метод сессионного контроля доступа к файловым объектам. Вопросы практической реализации // Вестник компьютерных и информационных технологий. 2014. № 8. С. 54—60.
20. *Щеглов К. А., Щеглов А. Ю.* Практическая реализация дискреционного метода контроля доступа к создаваемым файловым объектам // Вестник компьютерных и информационных технологий. 2013. № 4. С. 43—49.

21. *Щеглов К. А., Щеглов А. Ю.* Защита от атак на уязвимости приложений // Информационные технологии. 2014. № 9. С. 34—39.

22. *Щеглов К. А., Щеглов А. Ю.* Принципы реализации дополнительной защиты информации при контроле доступа к создаваемым файловым объектам на основе их автоматической разметки // Вопросы защиты информации. 2014. Вып. 104, № 1. С. 29—34.

23. Пат. 2533061 РФ. Система контроля доступа к шифруемым создаваемым файлам / *А. Ю. Щеглов, К. А. Щеглов.* Приор.26.06.2013.

Сведения об авторах

Константин Андреевич Щеглов — аспирант; Университет ИТМО; кафедра вычислительной техники; E-mail: info@npp-itb.spb.ru

Андрей Юрьевич Щеглов — д-р техн. наук, профессор; Университет ИТМО; кафедра вычислительной техники; E-mail: info@npp-itb.spb.ru

Рекомендована кафедрой вычислительной техники

Поступила в редакцию 07.10.14 г.

Ссылка для цитирования: *Щеглов К. А., Щеглов А. Ю.* Новый подход к защите данных в информационной системе // Изв. вузов. Приборостроение. 2015. Т. 58, № 3. С. 157—166.

NEW APPROACH TO DATA SECURING IN INFORMATION SYSTEM

K. A. Shcheglov, A. Yu. Shcheglov

ITMO University, 197101, Saint Petersburg, Russia
E-mail: info@npp-itb.spb.ru

A new method of securing data processed in information system is considered. The approach is based on realization of control of access to newly created object (file, clipboard) in order to exclude the object from the access policy due to automatic marking on newly created objects. Practical realization of the method is illustrated by an example of patented, implemented, and approved technical solution. Application of the proposed approach is shown to change all requirements to other security mechanisms designed to solve problems other than access rights control and restriction and therefore generates a new information system data securing technology.

Keywords: information system, data security, unauthorized access, access rights control and restriction, access policy, newly created object.

Data on authors

Konstantin A. Shcheglov — Post-Graduate Student; ITMO University; Department of Computer Science, E-mail: info@npp-itb.spb.ru

Andrey Yu. Shcheglov — Dr. Sci., Professor; ITMO University; Department of Computer Science, E-mail: info@npp-itb.spb.ru

Reference for citation: *Shcheglov K. A., Shcheglov A. Yu.* New approach to data securing in information system // Izvestiya Vysshikh Uchebnykh Zavedeniy. Priborostroenie. 2015. Vol. 58, N 3. P. 157—166 (in Russian).

DOI: 10.17586/0021-3454-2015-58-3-157-166