РЕАЛИЗАЦИЯ КОНТРОЛЯ И РАЗГРАНИЧЕНИЯ ПРАВ ДОСТУПА ПО СОЗДАНИЮ ФАЙЛОВ И ПРАВ ДОСТУПА К СТАТИЧНЫМ ФАЙЛОВЫМ ОБЪЕКТАМ

Введение.

В работах [1,2] авторами были предложены принципы и методы контроля доступа к создаваемым файловым объектам, основанные на их автоматической разметке и их практическая реализация. Эти методы позволяют реализовать разграничительную политику доступа к данным, обрабатываемым в информационной системе (к создаваемым в процессе работы пользователей файлам). Принципиальным достоинством подобных методов контроля доступа, кардинально меняющим собственно технологию защиты обрабатываемых в информационной системе данных от несанкционированного к ним доступа, является простота администрирования реализующих их средств защиты и корректность реализации разграничительной политики доступа в общем случае [3,4]. Вместе с тем, данные методы позволяют реализовать разграничительную политику доступа в отношении именно создаваемых объектов (создаваемых в процессе работы пользователей файлов, буфера обмена, после размещения в нем данных), но не разграничивать права доступа по созданию новых файлов, равно как и не позволяют разграничивать прав доступа к системным файлам, которые исходно присутствуют в системе (не создаются пользователями в процессе работы).

В данной работе рассмотрим реализацию контроля и разграничения прав доступа к файловым объектам, решающего данные задачи защиты информации, а также средство защиты, реализующее запатентованные авторами технические решения [5,6], использованные и апробированные при разработке комплексной системы защиты информации «Панцирь +" для ОС Microsoft Windows.

1. Требования и подходы к реализации контроля и разграничения прав доступа по созданию файловых объектов и к системным объектам.

Принципиальным при формировании требований к реализации контроля доступа к защищаемым ресурсам, в том числе, к файловым объектам, является ответ на следующий вопрос: каким образом и в каком виде хранятся правила доступа субъектов к объектам (или к объектам субъектов)? Именно способ хранения правил доступа во многом влияет на функциональные возможности контроля доступа, поскольку им определяется то, каким образом идентифицируется в разграничительной политике объект доступа.

Альтернативными способами хранения правил доступа является их хранение либо в качестве атрибутов, соотносимых с объектами доступа, либо в виде отдельной таблицы (матрицы) доступа, хранящейся, например, в отдельном файле. Атрибуты доступа для задания разграничительной политики используются, например, в схеме контроля доступа, в том числе, к файловым объектам, в современных операционных системах. Применительно к любому конкретному файловому объекту (каталогу, файлу) могут быть назначены правила доступа - каким субъектам (в данном случае, пользователям или группам пользователей), какие права доступа разрешены/запрещены. Подобный способ хранения атрибутов позволяет реализовать сущность «Владения» создаваемым файловым объектом, состоящую в том, что пользователь, создавший объект, как его владелец, может назначить права доступа к созданному им объекту другим субъектам - пользователям. Однако данная возможность противоречит требованиям соответствующего нормативного документа к реализации контроля и разграничения прав доступа [7]: «Право изменять правила разграничения доступа должно предоставляться выделенным субъектам (администрации, службе безопасности и т.д.)», что, на взгляд авторов полностью оправдано в современных условиях, когда санкционированный пользователь несет в себе весьма актуальную угрозу хищения обрабатываемой им информации [8].

Хранение правил доступа в отдельной таблице позволяет принципиально иначе задавать объекты доступа - для задания объекта доступа в разграничительной политике могут использоваться маски и переменные среды окружения. Например, маской «*» задается любой файловый объект, C:* - любой файловый объект на диске C, маской «%Windir%*.exe» – любой файл с расширением «exe» из каталога загруженной ОС, маской «*\TEMP\» - любой каталог с соответствующим именем, маской «*.exe» - любой файл с расширением exe. Все это возможно благодаря тому, что правила доступа в виде атрибутов не сопоставляются с конкретным объектом.

Видим, возможности по заданию объекта доступа в разграничительной политике принципиально расширяются. Но главное другое - это качественное расширение функциональных возможностей реализуемого контроля и разграничения прав доступа.

Проиллюстрируем сказанное на примере реализации контроля и разграничения прав доступа по созданию файлов. Появляется возможность разграничивать не только то, в каких объектах можно/нельзя субъекту создавать файловые объекты, в первую очередь, файлы, но и то, какие типы файлов ему разрешено/запрещено создавать. В частности, эту возможность можно использовать для защиты от несанкционированной загрузки на компьютер программ, в том числе, вредоносных, и модификации санкционированно установленных программ (для этого достаточно запретить создавать файлы с расширениями исполняемых объектов). Аналогичным образом при помощи масок могут назначаться объекты файловой системы с расширениями *.js, *.vbs, *.php и др., которые являются скриптовыми файлами. Предотвращение возможности создания на компьютере подобных файлов, загружаемых с веб-сайтов (большинство из этих скриптов предназначено для автозаполнения форм, рекламного баннера, загрузки дополнительной страницы с рекламой и т.д.) не только позволяет защитить соответствующее приложение от наделения его вредоносными свойствами

[9], но и позволяет защитить пользователя от получения им ненужной рекламы и ссылок при работе в сети интернет.

На этом же примере рассмотрим принципиальное изменение задания правил доступа, реализуемых при реализации разграничительной политики доступа к объектам, задаваемыми масками. Например, если установить правило запрета переименования объекта, задаваемого маской «*.exe», то это правило должно действовать, как в части запрета переименования любого файла с расширением exe в любой иной файл (в том числе в файл с иным расширением), так и наоборот, любого иного файла в файл с расширением exe.

Также к важным вопросам, формирующим требования к реализации контроля доступа, следует отнести способ задания (используемый набор сущностей) в разграничительной политике субъекта доступа и способ назначения правил доступа – назначаются ли правила доступа субъекта к объектам, либо, наоборот, к объекту субъектов.

Вопросы задания субъекта доступа при реализации разграничительной политики доступа в современной информационной системе исследованы в [10]. В частности дано обоснование тому, что поскольку процесс (приложение) в современных условиях по ряду причин несет в себе угрозу несанкционированного доступа к информации ничуть не меньшую, если не большую, чем пользователь, субъект доступа должен задаваться в разграничительной политике сущностью «Пользователь, процесс» (какой пользователь каким процессом запрашивает доступ), т.к. все процессы в операционной системе наследуют права доступа запустившего их пользователя, правила доступа для них совпадают. А с целью защиты от обхода заданной разграничительной политики доступа, за счет смены идентификатора пользователя при доступе к ресурсам, например, по средством использования штатной возможности современных операционных систем – сервисов олицетворения, в [10] предложено идентифицировать пользователя следующим образом «Исходный (первичный) идентификатор

пользователя (которым запущен процесс), эффективный идентификатор пользователя (под которым запрашивается процессом доступ к объекту), процесс (полнопутевое имя исполняемого файла процесса)». Заметим, что при этом выполняется и соответствующее требование соответствующего нормативного документа [7]: «Комплекс средств защиты (КСЗ) должен требовать от пользователей идентифицировать себя при запросах на доступ. КСЗ должен подвергать проверке подлинность идентификации - осуществлять аутентификацию. КСЗ должен располагать необходимыми данными для идентификации и аутентификации. КСЗ должен препятствовать доступу к защищаемым ресурсам неидентифицированных пользователей и пользователей, подлинность идентификации которых при аутентификации не подтвердилась».

При контроле и разграничении прав доступа к защищаемым ресурсам с использованием двух сущностей, идентифицирующих пользователя – исходный и эффективный, при запуске процесса запоминается, каким пользователем он запущен (первичный идентификатор), а при запросе этим процессом доступа к ресурсу определяется, от лица какого пользователя запрашивается доступ (эффективный идентификатор). Разграничительной политикой устанавливаются разрешения/запреты смены идентификатора при доступе к защищаемым ресурсам. Аутентификация в данном случае состоит в проверке подлинности идентификации – корректности смены (если она разрешена) первичного идентификатора при запросах доступа к ресурсам.

Замечание. Естественно, что разграничительная политика доступа должна задаваться в отношении, как интерактивных, так и системных пользователей, как прикладных, так и системных процессов.

При задании субъекта доступа, по аналогии с тем, как это делается при задании объекта доступа, опять же могут использоваться маски и переменные среды окружения, что принципиально упрощает задачу администрирования соответствующего средства защиты при включении в субъект доступа сущности «процесс».

Хранение правил доступа в виде отдельного объекта (без привязки в качестве атрибутов доступа к файловым объектам) позволяет принципиально изменить способ назначения прав доступа и отображения заданной разграничительной политики доступа, реализовав назначение прав доступа не к объектам субъектов, а субъектов к объектам. Это крайне принципиальный момент. Реализация данного подхода не только кардинально упрощает реализацию настройки разграничительной политики доступа, но и позволяет наглядно в одном окне интерфейса средства защиты отображать заданные права доступа для выбранного субъекта ко всем объектам, а не к конкретному объекту субъектов, что требуется при реализации разграничений прав доступа между субъектами. Чтобы получить аналогичную информацию о реализованной разграничительной политике доступа к объектам в отношении какого-либо субъекта при альтернативном способе задания прав доступа, потребуется просмотреть атрибуты доступа всех файловых объектов в системе.

Теперь рассмотрим, пожалуй, один из ключевых моментов использования масок при задании субъектов и объектов доступа в разграничительной политике, какие при этом могут возникать противоречия, и как они могут устраняться.

В общем случае под одну и ту же маску будет подпадать (покрываться маской) несколько реальных субъектов, соответственно объектов доступа, с этой целью маски и используются. Но при этом один и тот же реальный субъект, соответственно объект доступа может подпадать одновременно под несколько масок.

Проиллюстрируем сказанное. Пусть с запросом доступа к какому-либо файловому объекту обращается процесс с полнопутевым именем исполняемого файла E:\Program Files\Internet Explorer\iexplore.exe. Разграничительная же политика задана (заданы разные правила в таблице) для следующих субъектов, идентифицируемых масками «*», «E:\Program Files*», «*\iexplore.exe», «*.exe». Всеми представленными масками

покрывается рассматриваемый реальный субъект доступа (процесс, запросивший доступ) E:\Program Files\Internet Explorer\iexplore.exe. Возникает вопрос – какое правило (для какого субъекта) и, исходя из каких соображений, выбрать из разграничительной политики (из таблицы правил) при анализе корректности (непротиворечивости заданной разграничительной политики доступа) запроса доступа. Аналогичные вопросы возникают и в случае, если рассматривать задание масками объектов доступа.

В [5,6] предложено осуществлять выбор правила по наиболее точному описанию, в том числе, маской реального субъекта (объекта) доступа (субъекта, запрашивающего доступ, либо объекта, к которому запрашивается доступ). Преимущество данного подхода состоит в том, что при его реализации не требуется настраивать наследование прав доступа от включающего файлового объекта к включаемым. Опять же проиллюстрируем сказанное примером. Пусть заданы правила доступа к двум объектам, идентифицируемым в разграничительной политике масками – субъекту запрещено исполнение файлов из объекта «*» и разрешено исполнение из объекта «E:\Program Files*». В результате при запросе исполнения субъектом любого файла из подкаталогов каталога E:\Program Files, запрос доступа будет считаться корректным, из любого иного каталога – нет. В том числе, эта возможность может эффективно использоваться для аудита попыток несанкционированного доступа. Можно, например, запретить доступ к папке, но регистрировать попытки доступа только к одному файлу из этой папки (этот файл должен быть задан как отдельный объект, к которому должен быть также запрещен доступ, но в отношении этого объекта должен быть установлен аудит – аудит доступ к папке, включающей данный объект, не устанавливается, при этом к иным объектам в папке попытки доступа регистрироваться не будут). Можно привести массу практических возможностей применения данного решения.

Однако возникает вопрос, а какая, например, из двух масок «E:\Program Files*» или «*\iexplore.exe» более точно описывает реальный субъект (или объект) доступа E:\Program Files\Internet Explorer\iexplore.exe? Естественно, что в общем случае - это маска «*\iexplore.exe». Таким образом, можно задать правила, позволяющие средству защиты автоматически ранжировать между собою маски задаваемых в разграничительной политике субъектов (объектов) доступа по точности описания реального субъекта (объекта). Например, маска «E:\Program Files*» точнее описывает субъект (объект), чем маска «*», а маска «*\iexplore.exe», чем маска «*.exe».

Естественно, что при решении некоторых задач защиты администратору потребуется изменить автоматически устанавливаемые системой ранги субъектов, либо объектов. Например, применительно для двух рассматриваемых ранее масок субъектов «E:\Program Files*» или «*\iexplore.exe» ему важнее то, в какой папке (на каком диске) находится исполняемый файл приложения (субъекта доступа). Для универсальности решения подобная возможность должна быть соответствующим образом реализована.

В [5,6] предложено техническое решение (с различными способами идентификации субъекта доступа), реализующее автоматическое ранжирование системой между собою масок задаваемых в разграничительной политике субъектов (объектов) доступа по точности описания реального субъекта (объекта), с возможностью ручного изменения автоматически заданных рангов вручную администратором.

2. Средство контроля и разграничения прав доступа.

Рассмотрим техническую реализацию предложенного технического решения [6] на примере построения средства защиты «Комплексная система защиты «Панцирь+» для ОС Microsoft Windows (для иллюстрации будем использовать интерфейсы, реализованные в данном средстве защиты).

1. Создание субъектов доступа.

Субъекты доступа, которые, как ранее говорили, идентифицируются в разграничительной политике доступа тремя сущностями, создаются из меню, приведенного на рис.1.

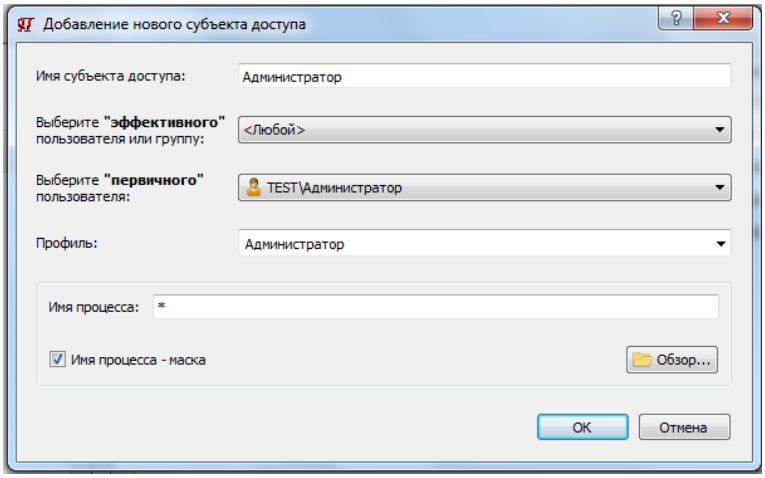

Рис.1. Меню создания субъекта доступа в разграничительной политике

Созданные субъекты автоматически ранжируются (по заданным правилам) по точности описателя и отображаются в интерфейсе в соответствующем порядке (либо сверху вниз, либо снизу вверх), см. рис.2.

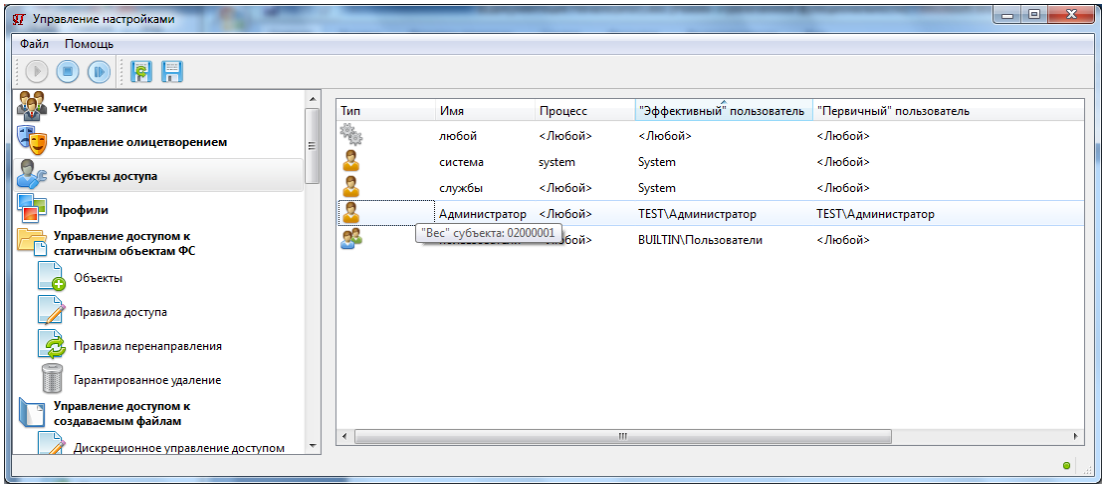

Рис.2. Интерфейс отображения созданных субъектов доступа в разграничительной политике

При необходимости администратор имеет возможность принудительно изменить автоматически заданный системой ранг («вес») субъекта доступа, см. рис.3, при этом в отображении созданных субъектов, субъект, которому изменяется ранг, будет соответствующим образом смещаться в списке заданных субъектов вниз, либо вверх.

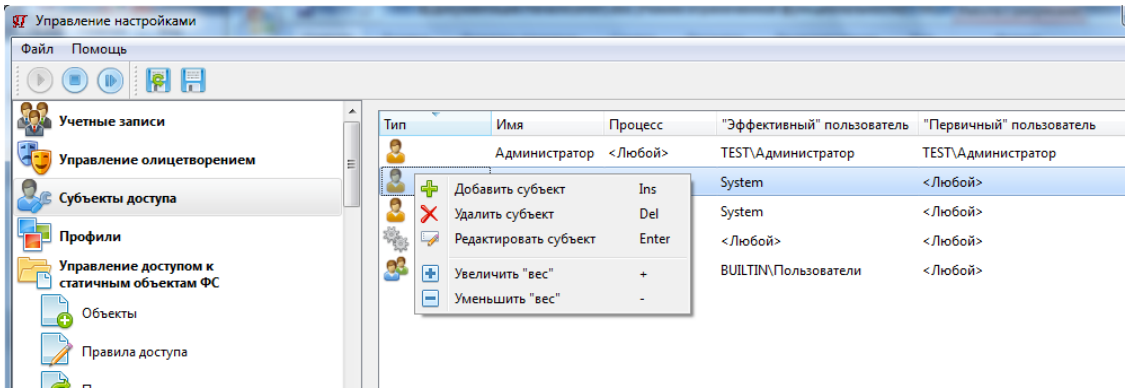

Рис.3. Меню изменения «веса» субъектов доступа в разграничительной политике

В качестве субъектов доступа в разграничительной политике используются профили, что в том числе позволяет реализовать ролевую модель контроля доступа в том случае, если профили создаются под определенные роли.

Субъекты, которые будут обладать одинаковыми правами доступа к объектам, помещаются в один и тот же профиль (для помещения субъекта в профиль, имена соответствующего субъекта и профиля задаются при создании субъекта в окне задания субъекта доступа, см. рис.1). Интерфейс отображения созданных профилей, в котором отображаются созданные администратором профили и включенные в профили субъекты (описываемые своими именами), приведен на рис.4. Для упрощения администрирования реализована возможность создания нового профиля на основе существующего, по средством необходимой модификации существующего профиля.

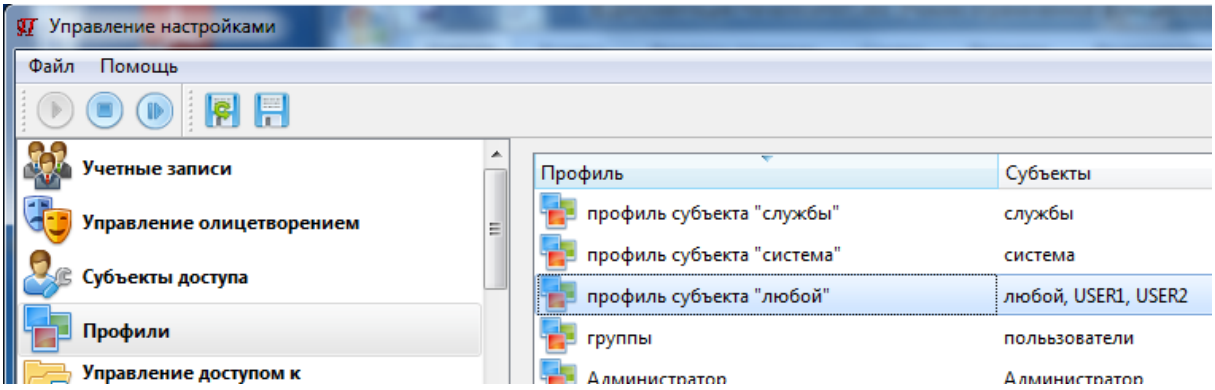

Рис.4. Интерфейс отображения профилей в разграничительной политике

2. Создание объектов доступа.

Файловые объекты доступа создаются из меню, приведенного на рис.4 и отображаются в интерфейсе, представленном на рис.5.

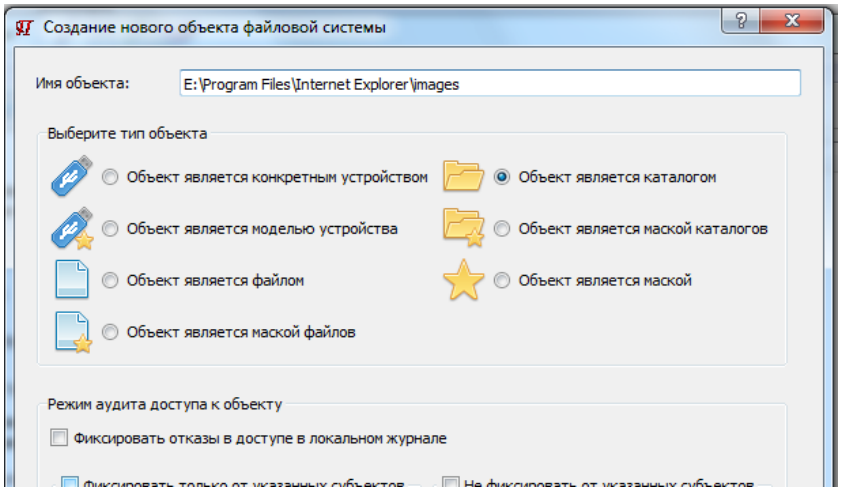

Рис.5. Меню создания объекта доступа в разграничительной политике

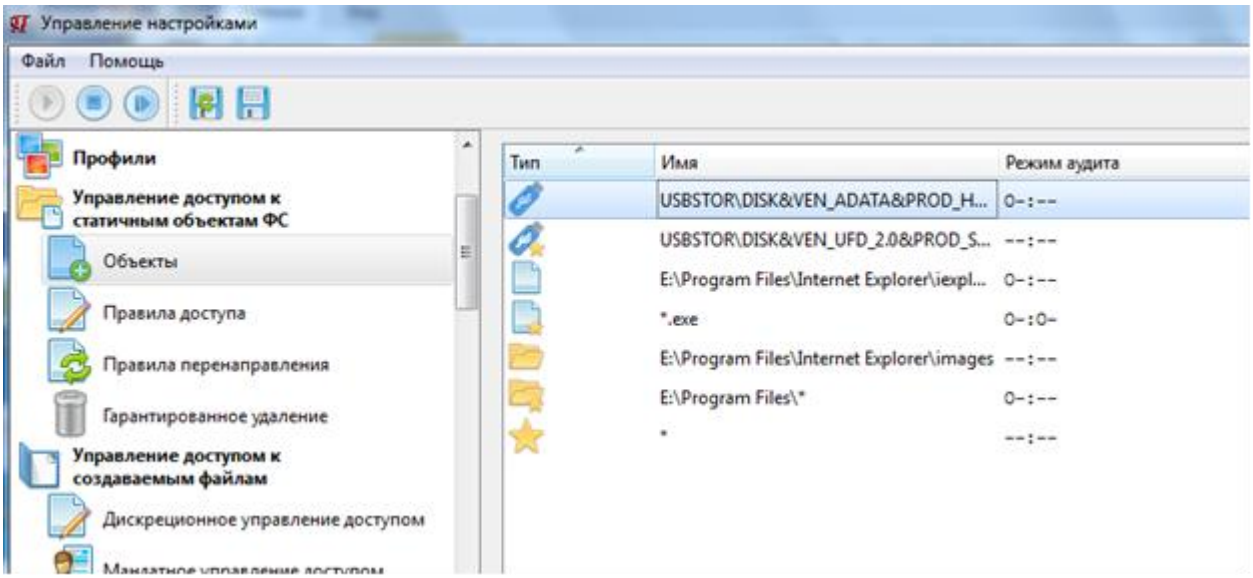

Рис.6. Интерфейс отображения созданных объектов доступа в разграничительной политике

Как отмечали ранее, при задании объектов доступа могут использоваться маски и переменные среды окружения. С учетом возможности использования масок, одновременно несколько заведенных объектов, могут соответствовать реальному объекту, определяемому его

полнопутевым именем, к которому запрашивается доступ. Для выбора правила доступа реализован следующий приоритет точности описания объектов в разграничительной политике, исходя из их типа: файл, маска файла, каталог, маска каталога, маска. Заданные в разграничительной политике объекты сравниваются с реальным объектом и запроса доступа, с целью определения наиболее точного описателя, в заданном порядке обработки - файл, если не подходит, то маска файла, соответственно, каталог, маска каталога, маска.

Тип объекта (файл, маска файла, каталог, маска каталога, маска) при его заведении в системе задается автоматически, но при реализации конкретной разграничительной политики, при этом администратору предоставляется возможность смены типа файлового объекта (назначения вручную) при его создании. пользователю может понадобиться установить тип объекта вручную.

В качестве файловых объектов доступа в разграничительной политике могут выступать локальные и разделенные в сети файловые объекты (при разграничении прав доступа к разделенным в сети объектам, объект доступа задается в разграничительной политике доступа следующим образом: «//имя машины/ имя файлового объекта», при этом также используются маски и переменные среды окружения.

Обособленно при задании объектов доступа находятся файловые устройства (файловые накопители). Для реализации корректной разграничительной политики доступа (предотвращения возможности ее обхода), объект доступа файловое устройство в разграничительной политике доступа необходимо идентифицировать не буквой диска, к которой устройство было примонтировано (она может быть изменена), а непосредственно идентификатором устройства. При этом объект доступа можно задавать, как моделью (типом) устройств, так и конкретным устройством, идентифицируемым его серийным номером, см. рис.7.

Созданные объекты доступа файловые устройства соответствующим образом отображаются интерфейсе, представленном на рис.6.

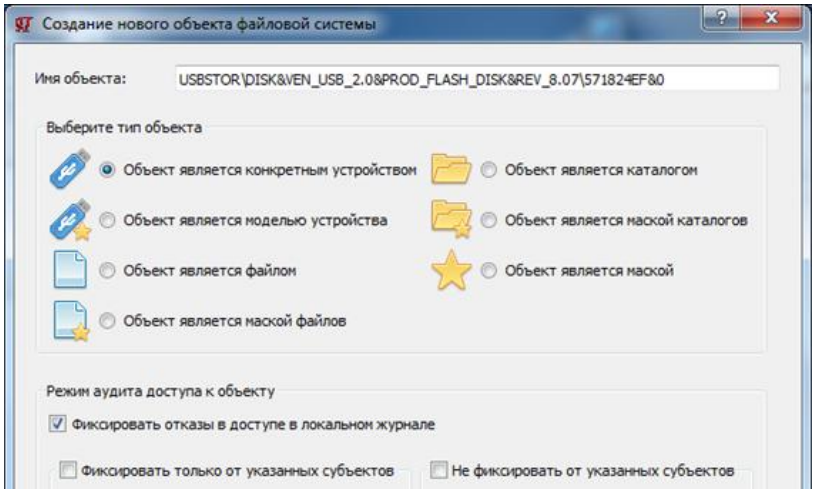

Рис.7. Задание объекта доступа устройство в разграничительной политике

Замечание. В строке «Имя объекта», см. рис.7, в результате использования обзора устройств, появляется идентификатор устройства. В этой строке можно дополнить имя файлового объекта на устройстве вручную, в том числе, используя маски, или скопировав имя каталога или файла, расположенных на устройстве, из строки «Имя объекта», полученного путем использования обзора каталогов или файлов на этом устройстве.

3. Создание правил доступа и правил аудита доступа.

Правила доступа и правила аудита для заданных субъектов к заданным объектам создаются из меню, приведенного на рис.8 и отображаются в интерфейсе, применительно к выбранному субъекту (профилю), представленном на рис.9.

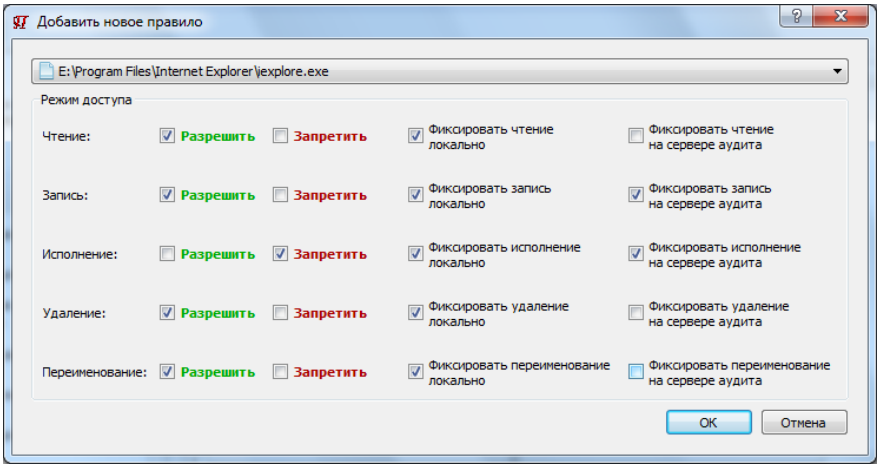

Рис.8. Меню задания правил доступа в разграничительной политике

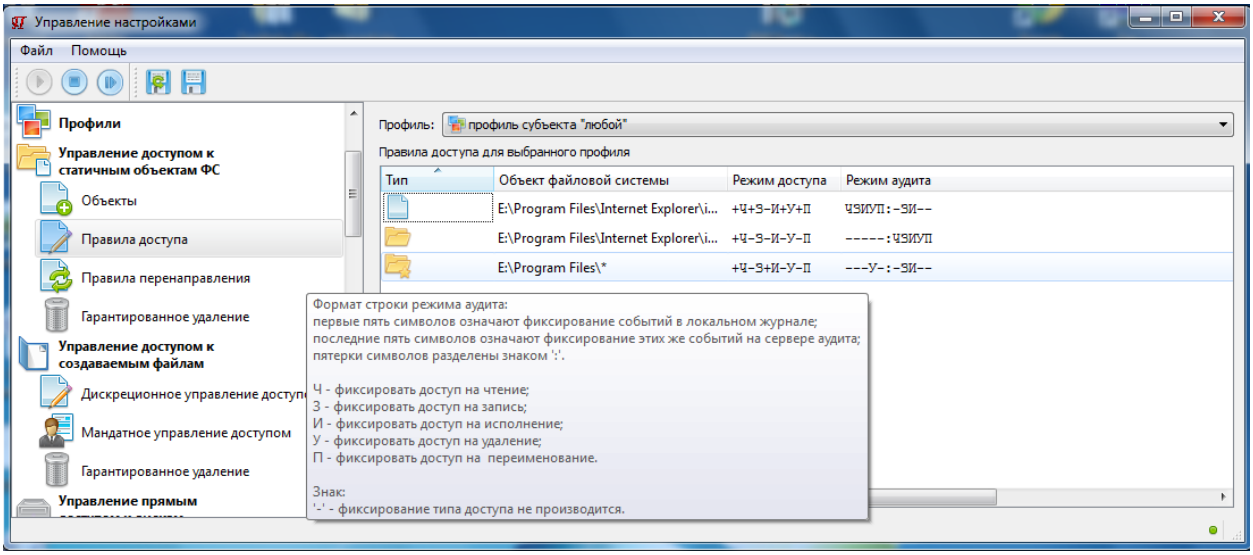

Рис.9. Интерфейс отображения созданных правил доступа и аудита доступа для выбранного субъекта в разграничительной политике

Отметим, что как для отдельно взятого субъекта, так и для системы в целом может быть реализована, как запретительная «Все, что явно указано, то запрещено», так и разрешительная «Все, что явно не указано, то запрещено» политики доступа. Для задания разрешительной политики доступа необходимо для соответствующего субъекта, либо для всех субъектов (субъект задается масками «*,*,*» запретить все права доступа к объекту, задаваемого маской «*», после чего вводить требуемые разрешения доступа, для которых объекты и/или субъекты доступа в разграничительной политике будут задаваться более точными описателями.

В качестве же объектов доступа в создаваемых разграничительных политиках, в том числе по созданию новых файловых объектов, могут использоваться файловые объекты, в том числе на внешних накопителях, присутствующие на момент задания правил доступа администратором (классифицируемые нами в [1], как статичные), в том числе системные файловые объекты.

Как видим, рассмотренные решения имеют, что имеют общего, с используемыми сегодня на практике реализациями методов контроля и

разграничения прав доступа к файловым объектам, и могут эффективно применяться при реализации сложных разграничительных политик доступа.

Отметим, что технические решения [5,6], практическая реализация которых рассмотрена в работе, могут использоваться при реализации контроля и разграничения прав доступа к любым защищаемым ресурсам – к объектам реестра ОС, к принтерам, к сетевым объектам и т.д., с учетом соответствующих особенностей задания объектов и правил доступа, что, к слову сказать, заявлено в [5,6]. Поскольку речь в работе идет о контроле и разграничении прав доступа к системным объектам, то сказанное можно проиллюстрировать на примере решения соответствующей задачи, применительно к объектам реестра ОС (естественно, Microsoft Windows), см. рис.10 - рис.12.

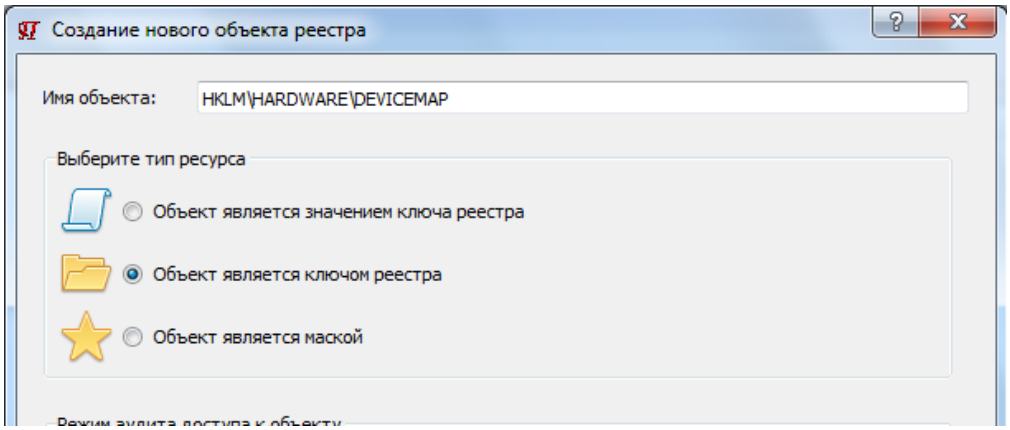

Рис.10. Задание объекта доступа объект реестра ОС в разграничительной политике

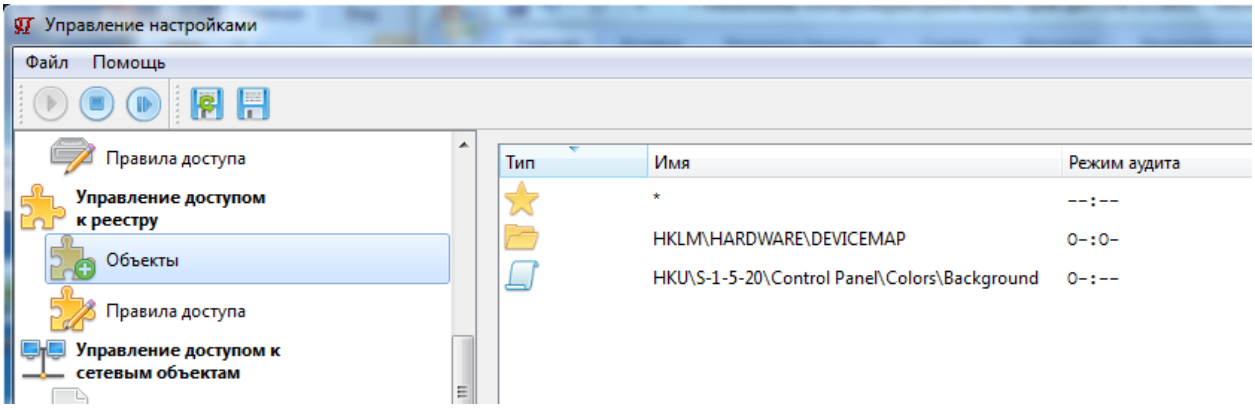

Рис.11. Интерфейс отображения созданных объектов доступа – объектов реестра ОС в разграничительной политике

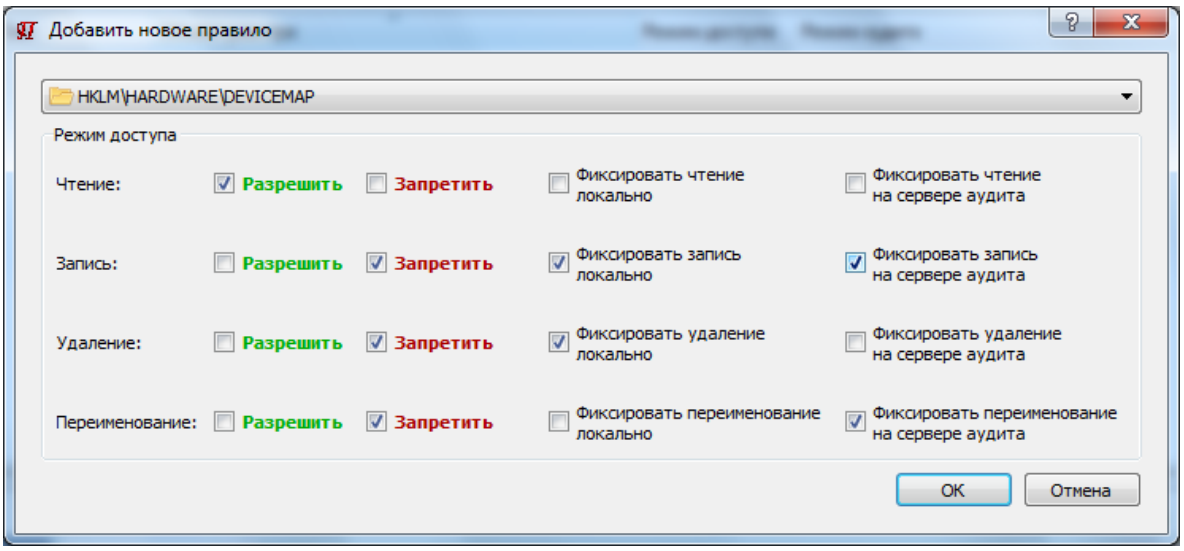

Рис.12. Меню задания правил доступа к объектам реестра ОС в разграничительной политике

Заключение. В заключении отметим следующее. Одна из ключевых задач защиты информации, формулируемая в предположении, что именно процесс несет в себе доминирующую угрозу несанкционированного доступа к информации по причинам, рассмотренным в [11,12], решается сегодня, так называемыми, системами обнаружения (обнаружения и предотвращения) вторжений, основу которых (для систем уровня хоста) составляет анализ поведения процессов (приложений) по средством периодического анализа журналов ОС и приложений. У подобной технологии защиты множество недостатков, ключевым из которых является практическая невозможность в общем случае предотвращения в реальном времени атаки процесса на защищаемый ресурс, даже в случае обнаружения вторжения. Очевидно, что альтернативным эффективным решением рассматриваемой задачи защиты может послужить реализация разграничительной политики доступа к ресурсам для субъекта, включающего в себя сущность «процесс», в том числе, с применением рассмотренного в работе решения и решения, реализующего контроль доступа к создаваемым объектам, описанного, например, в [1,2], поскольку данный подход к защите позволяет не только обнаружить вторжение, но и предотвратить в реальном времени соответствующую атаку.

Однако рассмотрение данного подхода к защите выходит за рамки настоящей работы.

Литература.

1. К.А. Щеглов, А.Ю. Щеглов. Принцип и методы контроля доступа к создаваемым файловым объектам // Вестник компьютерных и информационных технологий. - 2012. - № 7. - С. 43-47.

2. Щеглов А.Ю., Щеглов К.А. Система контроля доступа к файлам на основе их автоматической разметки. Патент на изобретение № 2524566. Приоритет изобретения 18.03.2013.

3. 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.

4. Щеглов К.А., Щеглов А.Ю. Реализация метода мандатного доступа к создаваемым файловым объектам // Вопросы защиты информации. - 2013. - Вып. 103. - № 4. - С. 16-20.

5. Щеглов А.Ю., Щеглов К.А. Система контроля доступа к ресурсам компьютерной системы с субъектом доступа «пользователь», «процесс». Положительное решение на выдачу патента на изобретение по заявке № 201320208/08(030001) от 30.04.2013.

6. Щеглов А.Ю., Щеглов К.А. Система контроля доступа к ресурсам компьютерной системы с субъектом «исходный пользователь», «эффективный пользователь», «процесс». Положительное решение на выдачу патента на изобретение по заявке № 2013128215/08(041992) от 18.06.2013.

7. Гостехкомиссия России. Руководящий документ. Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от НСД к информации. - Москва, 1992.

8. Опрос "Кода Безопасности" выявил наиболее актуальные ИБ угрозы [Электронный ресурс]// URL: //http://www.securitycode.ru/company/news/SC-analytic-2011.

9. Маркина Т.А., Щеглов А.Ю. Метод защиты от атак типа drive-by загрузка. - Известия ВУЗзов. Приборостроение, 2014. - № 4. - С. 15-20.

10. 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.

11. 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.

12. 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.