

Современные методы проверки свойств безопасности в моделях логического разграничения доступа

К. А. Шапченко

Институт проблем информационной безопасности
Московского государственного университета им. М. В. Ломоносова, 119234, Москва, Россия

Представлен краткий обзор современных подходов к проверке свойств безопасности в моделях логического разграничения доступа. Рассматриваются подходы на основе методов теории графов, верификации на модели и автоматического доказательства теорем. Приведены типовые свойства, проверка которых может быть осуществлена с использованием указанных методов.

Ключевые слова: информационная безопасность, контроль доступа, теория графов.

This paper presents a review of modern approaches to verification of security properties in access control models. Reviewed approaches include methods based on graph theory, model checking and automated theorem proving. Several typical security properties are discussed as examples of properties which can be verified by applying these approaches.

Keywords: security, access control, graph theory.

Введение. Механизмы логического разграничения доступа (ЛРД) играют главную роль в обеспечении информационной безопасности современных автоматизированных систем. Такие механизмы реализованы во многих классах программных средств: ядрах операционных систем, системах управления базами данных, сетевых сервисах и многопользовательских веб-приложениях. Задача механизмов ЛРД состоит в проверке, является ли доступ некоторого субъекта к некоторому объекту разрешенным в заданной системе правил разграничения доступа [1-4]. Выделение субъектов, объектов и типов доступа, а также формирование правил разграничения доступа проводится с учетом особенностей используемого механизма разграничения доступа. Например, для ядра многопользовательской операционной системы, могут быть использованы следующие определения основных понятий для механизма ЛРД: субъектом является процесс в операционной системе; объектом доступа является файл в используемой файловой системе; к типам доступа относятся доступ на чтение и доступ на запись; правила формируются как список троек (идентификатор пользователя – владельца процесса, идентификатор файла, тип доступа). Представленный пример значительно упрощен по сравнению с механизмами разграничения доступа, используемыми на практике. Как правило, субъекты и объекты доступа (далее при обозначении таких элементов модели, как субъекты и объекты доступа, для краткости будем использовать термин "сущность") обладают большим количеством атрибутов, в зависимости от которых определяется, разрешен ли запрашиваемый доступ. Кроме того, в зависимости от функциональных возможностей компьютерной системы множество типов доступа может быть расширено [5]. Правила разграничения доступа, которые могут быть заданы в современных механизмах ЛРД, часто поддерживают более сложную логику принятия решения о доступе. Следует отметить, что на практике используется ряд способов сокращения описания таких правил, например с использованием группирования субъектов и объектов доступа, в том числе на основе ролевых моделей [6-8].

Функционирование механизмов ЛРД производится на основе заданной для них конфигурации – набора правил разграничения, представленных в форме входных данных для механизма ЛРД. Продолжая рассмотрение указанного упрощенного примера, заметим, что для механизма ЛРД в ядре операционной системы (ОС) правила могут храниться в специальных атрибутах объектов файловой системы в виде списков прав доступа для каждого отдельного файла. В качестве другого способа хранения таких правил в рассматриваемом примере может использоваться размещение конфигурации механизмов ЛРД в виде таблиц доступа в ядре ОС. Первый метод является традиционным для Unix-подобных ОС, второй используется, например, в механизме ЛРД RSBAC для ядра ОС Linux [5].

Выделим две категории правил разграничения доступа. К первой из них относятся правила, действующие в некотором фиксированном состоянии автоматизированной системы (АС). Такие правила составляют статическую часть правил логического разграничения доступа. Конфигурация механизмов ЛРД задается

перед началом работы автоматизированной системы и, как правило, может изменяться в процессе выполнения административных действий. Реализация подобных действий также должна контролироваться механизмами ЛРД. Ко второй категории относятся правила, описывающие логику принятия решения о выполнении административных действий, в рассматриваемом случае – действий по изменению правил ЛРД. Подобные правила составляют динамическую часть правил ЛРД.

Как отмечено выше, конфигурация механизмов логического разграничения доступа может быть изменена в процессе эксплуатации автоматизированной системы. Такие действия являются штатными в ходе эксплуатации многопользовательских автоматизированных систем при добавлении новых субъектов доступа и их атрибутов (например, ассоциированных с ними пользователей), а также новых объектов доступа (например, файлов, записей в базах данных, веб-ресурсов). Как правило, действия по изменению правил ЛРД производятся с использованием специализированных инструментальных средств. Например, в современных ОС изменение правил доступа к объектам файловых систем производится через предназначенный для этого системный вызов, пользовательский интерфейс к которому предоставляется отдельно функционирующими средствами или интегрирован в программное обеспечение, обладающее функциями управления файлами. Следует отметить, что подобные средства обычно реализуют только "локальное" изменение правил ЛРД – для одного или нескольких объектов доступа. Такая особенность приводит к фрагментарному управлению правилами ЛРД, без оценки всех правил в их объединении. В свою очередь, указанное обстоятельство может привести к ошибкам настройки, в том числе позволяющим получить несанкционированный доступ к контролируемым объектам. Проверка "вручную" всех правил и оценка соответствия их совместной работы заданным (часто неформально) требованиям, как правило, затруднена для администратора безопасности вследствие большого объема конфигурационных данных механизмов ЛРД в современных автоматизированных системах. В случае использования нескольких механизмов ЛРД задача их настройки еще более усложняется, что обусловлено использованием нескольких инструментальных средств, каждое из которых ориентировано на настройку только одного механизма ЛРД. При таком подходе не представляется возможным оценить результаты совместного функционирования нескольких подобных механизмов.

Перечисленные особенности процесса настройки механизмов логического разграничения доступа приводят к задачам автоматизации комплексного управления настройками механизмов ЛРД, в том числе к задачам автоматизированной проверки выполнения заданных требований в конфигурации таких механизмов. В настоящей работе представлен краткий обзор ряда подходов к спецификации и проверке подобных требований. Все рассматриваемые подходы реализуют статическую проверку требований, без вмешательства в процессе проверки в функционирование исследуемой автоматизированной системы. В контексте рассматриваемой задачи общая схема таких подходов представляет собой следующую последовательность действий:

- сбор данных о конфигурации механизмов ЛРД;
- представление полученных данных в виде формальной модели;
- задание проверяемых свойств на используемом в подходе языке спецификации;
- проведение проверки;
- интерпретация результатов с точки зрения настроек механизмов ЛРД.

Рассмотрим перечисленные действия с точки зрения используемых математических методов и подходов к проверке свойств в моделях ЛРД.

1. Типовые задачи проверки свойств в моделях логического разграничения доступа. Рассмотрим несколько классов типовых задач проверки выполнения заданных свойств в моделях ЛРД. Заметим, что областью исследования является именно процедура проверки выполнения свойств в моделях, а не проверки корректности функционирования механизмов ЛРД по отношению к формальным моделям ЛРД, которые ими реализуются. Задачи проверки соответствия механизма и модели в настоящей работе не рассматриваются.

Заметим также, что указываемые далее типовые задачи проверки свойств моделей ЛРД часто пересекаются. В формальной постановке для конкретной модели ЛРД и заданного класса проверяемых свойств такие задачи могут быть аналогичными. В ряде случаев данное обстоятельство позволяет использовать унифицированные математические методы и инструментальные средства для проверки выполнения свойств в моделях ЛРД.

Поиск конфликтов правил в модели ЛРД. Ключевым свойством, которое, как правило, ожидается от модели логического разграничения доступа, является однозначность определения возможности того или иного доступа согласно заданной статической части модели. Такое свойство естественным образом выполняется в моделях, в которых для каждой пары (субъект доступа, объект доступа) разрешенные типы доступа определяются только одним правилом. Однако во многих распространенных механизмах ЛРД используются логико-языковые средства, позволяющие записывать несколько правил, относящихся к одной и той же паре субъекта и объекта доступа. Часто такое решение обусловлено необходимостью оптимизации размера набора правил и использования более сложной логики принятия решения о доступе. Примером подобной оптимизации является сокращение количества правил за счет группировки субъектов или объектов доступа. При этом возможен конфликт правил для отдельного объекта и для группы, содержащей этот объект. Если эти правила не приводят к одному и тому же результату, то необходимо их скомбинировать таким образом, чтобы избежать неоднозначности. В качестве одного из представительных примеров усложнения логики принятия решения могут быть приведены так называемые немонотонные модели ЛРД. В таких моделях возможна реализация не только системы правил, в которой к общему (как правило, неявному) правилу "запретить все доступы" добавляются правила, разрешающие те или иные доступы, но и система правил, при которой к общему правилу "разрешить все доступы" добавляются правила о запрете доступов. Для указанных двух классов правил возможны конфликты при разрешении определенного действия в одном правиле и запрете того же действия в другом правиле. Распространенным подходом к устранению подобного конфликта в современных механизмах ЛРД является использование приоритетов правил, например применение правил ЛРД в порядке их задания. При этом традиционные подходы к формальному описанию моделей логического разграничения доступа не предполагают зависимости решения о доступе от порядка правил.

Заметим, что в случае использования логических средств устранения правил, например их комбинирования и упорядочения, задача поиска конфликтов не теряет актуальности. В этом случае использование методов поиска конфликтов правил ЛРД позволяет удостовериться в корректности использования средств устранения таких конфликтов.

Поиск конфликтов правил при объединении моделей ЛРД. Обобщением рассмотренного класса задач поиска конфликтов правил в одной модели ЛРД является поиск конфликтов при объединении нескольких таких моделей. Подобная операция объединения является типовой в случае построения распределенной автоматизированной системы на основе нескольких АС, в каждой из которых используются свои механизмы и правила разграничения доступа. Для составной (комбинированной) автоматизированной системы в силу ее распределенной архитектуры характерно наличие нескольких точек принятия решения о доступе. С учетом этого объединение моделей ЛРД происходит, как правило, неявным образом, без фактического построения новой, комбинированной модели для использования в распределенной АС. Основой подобного объединения является набор правил, которые связывают отдельные модели ЛРД. Примером одного из часто используемых правил такого рода является передача прав доступа между субъектами доступа в разных АС. В современных автоматизированных системах реализация рассматриваемого типового правила объединения традиционно организуется следующим образом. Пусть в составе распределенной АС имеется две автоматизированные системы, а именно АС1 и АС2. Субъект доступа из АС1 (например, ассоциированный с одним из пользователей АС1) подключается к АС2 и получает привилегии некоторого субъекта доступа из АС2. Таким образом, часть действий, регулируемых моделью ЛРД в АС2, становится доступной для субъекта доступа из АС1, что свидетельствует об объединении (в рассматриваемом случае – неявном) двух моделей ЛРД. Характерным примером реализации действий, приводящих к объединению моделей ЛРД, в современных АС является осуществление доступа к сетевым сервисам, предоставляемым автоматизированной системой, включая серверы удаленного интерактивного доступа, FTP- и HTTP-серверы, веб-сервисы и построенные на их основе веб-приложения.

В рассматриваемом случае остаются актуальными все задачи поиска конфликтов в модели ЛРД в применении к комбинированной модели ЛРД, заданной, как правило, неявным образом. Отличие подобных задач от поиска конфликтов в одной модели ЛРД заключается в необходимости учета правил объединения моделей. Такие правила можно представить в виде формальной модели с помощью способа, отличающегося от формализации отдельных объединяемых моделей ЛРД. Данное обстоятельство свидетельствует о необходимости доработки методов проверки свойств в моделях ЛРД для применения в случае их неявного объединения.

Среди многообразия задач проверки свойств в комбинированной модели следует выделить задачи определения эквивалентности принятия решения о локальных доступах – доступах в рамках одной из объединяемых моделей ЛРД – до и после объединения моделей. Каждому такому доступу в рамках одной модели обычно соответствует одно правило. В этом контексте проверка такого свойства отдельной модели ЛРД, как допустимость одного доступа, является тривиальной. Для выполнения этой проверки достаточно установить наличие соответствующего правила в заданной модели. В случае объединения нескольких моделей ЛРД задача проверки подобного свойства усложняется вследствие наличия связей между отдельными моделями ЛРД. Например, если некоторый доступ запрещен в локальной модели ЛРД, то требуемый от доступа эффект в ряде случаев может быть достигнут в результате последовательности доступов, регулируемых несколькими из объединяемых моделей.

Задачи рассматриваемого класса усложняются, если отсутствует информация о некоторых из объединяемых моделей. В этом случае целесообразно заменить такие модели на их внешние, неполные спецификации.

Сравнение двух моделей ЛРД. В процессе управления настройками механизмов ЛРД часто возникает необходимость сравнить два варианта одной модели ЛРД [9-11]. При этом один из вариантов может соответствовать модели до внесения изменений, другой – модели после внесения изменений. В этом случае ставится задача об определении вносимых изменений и сопоставлении их с некоторой спецификацией. Подобная спецификация изменений может ограничивать, например, область внесения изменений – набор сущностей в модели, в отношении которых меняются правила разграничения доступа. В случае внесения значительных изменений, например расширения множеств субъектов и объектов доступа и добавления новых правил, задача сравнения двух моделей ЛРД может рассматриваться как одна из указанных ранее задач поиска конфликтов при объединении моделей. Для обозначения задач такого класса часто используется термин "анализ влияния изменений" (change-impact analysis). На практике методы анализа влияния изменений используются, как правило, в области программной инженерии, а также при исследовании изменяющихся декларативных спецификаций программ или правил. Их примером в полной мере являются модели ЛРД [9].

Другой задачей сравнения двух моделей является определение соответствия оригинальной, "грубой" модели и ее уточнения. В общем случае подобная задача обозначается как "задача о вложении моделей" [10, 11]. Выполнение уточнения модели ЛРД представляется целесообразным при проектировании таких моделей "сверху вниз", когда в качестве проверяемого свойства выступает непротиворечивость оригинальной и уточненной моделей ЛРД. В конкретной постановке семантика свойства непротиворечивости может изменяться. Однако под таким свойством, как правило, понимается, что уточненная модель ЛРД не добавляет новых разрешенных доступов по сравнению с "грубой" моделью. Заметим, что в указанной постановке задача сравнения двух моделей ЛРД может быть интерпретирована как задача поиска конфликтов при объединении нескольких моделей. Действительно, пусть объединяются две модели ЛРД – оригинальная модель и ее уточнение, причем в качестве правил объединения рассматривается только согласование объектов и субъектов доступа, без определения порядка применения правил доступа из различных моделей. В этом случае возможно возникновение конфликтов правил разграничения доступа, при которых один и тот же доступ может быть одновременно разрешен (согласно правилам уточненной модели) и запрещен (по правилам оригинальной модели ЛРД). Задача поиска конфликтов указанного вида аналогична первоначально поставленной задаче о сравнении моделей.

Исследование свойств информационных потоков. В области проверки свойств в моделях логического разграничения доступа можно выделить класс задач, связанных с так называемыми разрешенными (или допустимыми) составными информационными потоками (далее для краткости информационными потоками) [1, 12, 13]. Понятие информационного потока определяется для моделей ЛРД, в которых ряд типов доступа подразумевает передачу информации между сущностями в модели следующим образом. Пусть в указанной модели ЛРД рассматривается субъект доступа S и объект доступа O . Будем считать, что элементарный информационный поток от S к O разрешен, если правилами ЛРД допускается доступ "на запись" от S к O . Аналогично элементарный информационный поток от O к S разрешен, если правилами допускается доступ "на чтение" от S к O . Под типами доступа "на чтение" и "на запись" понимаются такие, что при реализации соответствующего доступа информация передается от объекта к субъекту (для типа доступа "на чтение") и от субъекта к объекту (для типа доступа "на запись"). Заметим, что таких типов доступа каждого вида может быть несколько. Рассматриваемые типы доступа реализуются во многих компонентах современных автоматизированных систем. Важным примером являются механизмы операционных систем для логического раз-

граничения доступа к объектам файловых систем. В подобном случае типы доступа "на чтение" и "на запись" реализуются в функциях чтения и записи (в том числе продолжающей записи) файлов, чтения и изменения каталогов, чтения метаданных объектов файловой системы. Согласно обсуждаемому определению в рассматриваемом примере разрешены элементарные информационные потоки между процессами в ОС (субъектами доступа) и объектами файловых систем (объектами доступа). Из нескольких таких элементарных информационных потоков складываются составные допустимые информационные потоки. Будем считать, что между сущностями A1 и A2, каждая из которых может быть субъектом или объектом доступа в модели ЛРД, разрешен информационный поток, если существует конечная последовательность элементарных информационных потоков, такая что

- первый элементарный поток начинается в A1;
- последний элементарный поток заканчивается в A2;
- конец каждого элементарного потока, кроме последнего, совпадает с началом следующего элементарного потока.

Интерпретировать подобную конструкцию на примере операционной системы и доступа к объектам файловых систем можно следующим образом. Пусть процессу P1 в ОС не разрешен доступ на чтение файла Ф1, однако разрешен доступ на чтение файла Ф2. Одновременно процессу P2 в ОС разрешены доступы на чтение Ф1 и на запись Ф2. Следовательно, согласно приведенным определениям разрешен информационный поток $\Phi1 \rightarrow P2 \rightarrow \Phi2 \rightarrow P1$. Такое следствие указывает на то, что при определенной координации действий процессов P1 и P2 в операционной системе процесс P1 может получить информацию из файла Ф1, доступ на чтение к которому запрещен для P1 в соответствии с заданными правилами разграничения доступа. Подобное взаимодействие может противоречить общим принципам правил ЛРД, которые устанавливаются для автоматизированной системы. Это свидетельствует о некорректной настройке используемых механизмов логического разграничения доступа, например в том случае, если процессы, аналогичные P1, могут создаваться только пользователем с низким уровнем привилегий в АС, а файл Ф1 содержит данные, предназначенные для пользователей с высоким уровнем привилегий.

Вследствие важности ограничения составных информационных потоков в практически значимых механизмах логического разграничения доступа, ставятся задачи проверки свойств таких потоков в моделях ЛРД. Общий вид одного из типовых свойств информационных потоков формулируется следующим образом. Пусть задано ограничение на информационный поток, в частности указаны требования к начальной и конечной сущностям в потоке, а также к последовательности сущностей и типов доступа в потоке, например, что одна сущность A1 всегда следует после сущности A2 и между ними всегда имеется сущность A3. Необходимо проверить, выполняется ли это свойство для всех допустимых информационных потоков в заданной модели ЛРД. Заметим, что, как правило, кроме непосредственно проверки выполнения такого свойства появляется необходимость в построении контрпримера для случая, если данное свойство не выполняется. Дополнительная информация, получаемая в процессе интерпретации подобного контрпримера, позволяет упростить поиск некорректно заданных правил разграничения доступа.

Следует отметить, что задание свойств, которыми должны обладать информационные потоки, часто составляет основу для формализации других задач проверки свойств моделей ЛРД. Например, ряд задач поиска конфликтов в модели ЛРД или в объединении таких моделей можно сформулировать в терминах информационных потоков. Подобная формулировка может иметь вид ограничения на поток между двумя сущностями: если поток разрешен, то он обязательно имеет некоторый заданный вид. Примером ограничения на поток может являться спецификация его длины или требование "локальности" по отношению к одной из объединяемых моделей ЛРД. Проверка подобных свойств позволяет определить ряд дополнительных потоков и доступов, ставших допустимыми вследствие объединения моделей.

2. Подходы к представлению моделей логического разграничения доступа. Для проведения процедуры проверки свойств в моделях ЛРД необходимо формализованное описание (для краткости – спецификация) модели и проверяемых свойств. Целесообразно выделить следующие компоненты спецификации:

- описание автоматизированной системы и ее компонентов (в той части, которая важна для формирования модели ЛРД);
- описание исследуемой модели логического разграничения доступа;
- описание проверяемых свойств.

Следует отметить, что первое из указанных описаний, как правило, опускается, и в спецификации остаются только описания модели и проверяемых свойств. Подобный подход возможен, однако следует указать на одну из существенных ошибок, которая может быть допущена при его использовании. Часто принимается модель ЛРД, необоснованно упрощенная по сравнению с функциональными возможностями автоматизированной системы, в частности связанными с используемыми в этой системе механизмами разграничения доступа. При исследовании такое упрощение может привести к потере важных компонентов модели вплоть до отдельных субъектов и объектов доступа. Выделение описания автоматизированной системы в качестве самостоятельной части спецификации необходимо для дополнительного пояснения упрощений, производимых при описании модели ЛРД. Поясним данную особенность на примере. Пусть рассматривается задача проверки свойств в моделях ЛРД, которые используют механизмы ОС при разграничении доступа к объектам файловых систем. С одной стороны, в качестве объектов доступа можно выделить множество всех файлов, расположенных в файловых системах. В рассматриваемом случае такое упрощение представляется естественным, поскольку именно файлы содержат информацию, доступ к которой необходимо ограничивать. Однако в представленном упрощении теряется информация о других элементах файловой системы, которые могут быть использованы как объекты доступа, в том числе для организации информационных потоков. Примерами таких элементов являются древовидная система каталогов в файловой системе и метаданные хранимых в ней объектов. Перечисленные элементы тесно связаны с функциональными возможностями автоматизированной системы или отдельного ее программного компонента. Отмеченное обстоятельство и указанная связь с моделью логического разграничения доступа свидетельствуют о необходимости проведения исследования функциональных возможностей, предоставляемых автоматизированной системой, для уточнения описания модели ЛРД.

Представление статической части правил логического разграничения доступа. Для описания правил, по которым определяется, разрешен ли доступ в модели ЛРД, традиционно используется следующий подход, как правило, называемый теоретико-множественным. Определяется набор базовых множеств, описывающих основные сущности, используемые в модели. Для этих множеств описывается ряд связывающих их отношений, набор предикатов-ограничений (если требуется в модели) и формулируется предикат предоставления доступа. Аргументами предиката предоставления доступа являются субъект, объект и тип доступа. Значение предиката определяет, разрешен ли такой доступ по заданной статической части правил ЛРД.

Теоретико-множественные описания построены для широкого класса моделей ЛРД. В качестве наиболее распространенных отметим подходы к формализации статических правил в моделях дискреционного разграничения доступа [3, 14], в моделях многоуровневого мандатного разграничения доступа [15, 16], а также в ролевых моделях ЛРД [6-8]. Например, базовым подходом является использование так называемой матрицы доступа – отображения из всех пар (субъект доступа, объект доступа) в подмножества множества всех типов доступа. Использование данного подхода, в том числе без каких-либо дополнительных усложнений, позволяет получить упрощенное описание модели ЛРД [3].

Традиционным способом сокращения описания статической части правил ЛРД является использование группирования субъектов и (или) объектов доступа. Такой подход наглядно демонстрируется при использовании групп в дискреционных моделях ЛРД, в том числе в механизмах операционных систем, а также в ролевых моделях ЛРД. Применение подобных подходов показательны с позиции оптимизации методов проверки свойств в моделях ЛРД. При этом главной особенностью является возможность выполнения проверки ряда свойств одновременно для набора сгруппированных субъектов или объектов доступа вместо выполнения каждой проверки в отдельности.

Описание динамической части правил логического разграничения доступа. При рассмотрении подходов к описанию правил выполнения административных действий для механизмов ЛРД следует отметить необходимость описания как самих правил, так и математической модели, согласно которой выполняются административные действия. Поясним указанное обстоятельство на примере. Пусть дана модель ЛРД, статическая часть правил в которой описывается отмеченным ранее упрощенным способом с использованием матрицы доступа. В качестве административных могут рассматриваться действия по изменению матрицы доступа. Правила, по которым определяется допустимость таких действий, могут быть описаны с помощью дополнительной матрицы доступа. В этой матрице типам доступа соответствуют возможности по изменению правил доступа к некоторому объекту, т. е. по изменению элементов основной матрицы доступа, описывающих доступ каждого из субъектов к заданному объекту. Для интерпретации таких правил динамиче-

ского изменения статической конфигурации может быть использована система переходов, состояниями которой являются описания статической части правил (основная матрица доступа), а переходы между ними определяются дополнительной матрицей доступа. Как и при описании статической части правил, может быть применен способ группирования сущностей в модели для уменьшения размера матрицы доступа. Схеме динамического изменения модели можно усложнить путем введения действий по изменению не только основной матрицы доступа, но и дополнительной. Такие изменения содержательно соответствуют передаче прав на управление доступом к заданным объектам. В рассматриваемой упрощенной модели это означает передачу права "владения" объектом по аналогии с дискреционным разграничением доступа к объектам файловых систем, используемым в распространенных ОС. Подобные изменения часто рассматриваются в качестве операций над некоторым математическим объектом, например графом, предоставляющим доступы в модели ЛРД, разрешенные в рамках принятого набора правил разграничения доступа [17]. Следует отметить развитие подходов к описанию действий по изменению в ролевых моделях ЛРД. Для ряда ролевых моделей разработаны расширения, позволяющие описывать допустимые административные действия и правила по управлению ими [18].

Универсальные языки для описания моделей ЛРД. Необходимо отметить ряд результатов в области стандартизации языков описания моделей ЛРД. В настоящее время известны два языка, в рамках которых могут быть специфицированы правила разграничения доступа: OASIS eXtensible Access Control Markup Language (XACML) и OASIS Security Assertion Markup Language [19, 20]. Языки XACML и SAML основаны на использовании XML-контейнеров для хранения предикатов, с помощью которых для каждого проверяемого доступа определяется, разрешен ли он в заданной модели. Заметим, что язык XACML позволяет описывать ряд подходов к объединению нескольких моделей ЛРД на основе задания порядка применения правил разграничения доступа. Для описаний на языке XACML предложен подход к их формализации в виде логической модели, позволяющий проводить проверку нескольких видов свойств с помощью автоматизированных средств для доказательства теорем [21].

Представление модели ЛРД в виде программы. Рассмотрим такой класс подходов к описанию моделей ЛРД, как представление их в виде программы на некотором языке программирования. Один из вариантов подобного представления можно построить естественным образом – достаточно взять исходный код программного механизма разграничения доступа, реализующего требуемую модель. Преимуществом такого варианта представления модели ЛРД является его точное соответствие исследуемому механизму разграничения доступа. Тем не менее данный подход обладает рядом существенных недостатков. Во-первых, в применении к рассматриваемым задачам проверки свойств следует отметить, что без выделения такой абстракции, как модель ЛРД, свойства формулируются как свойства алгоритма вычислений, задача доказательства которых в общей постановке является алгоритмически неразрешимой. Во-вторых, выделение алгоритма принятия решения о доступе из исходного кода компонентов автоматизированной системы часто затруднительно.

Еще одним вариантом рассматриваемого подхода является представление модели ЛРД в качестве вычислительного алгоритма, задаваемого на декларативном языке программирования (обычно на одном из языков логического программирования). Известны примеры такого подхода с использованием языков логического программирования [10, 22] и языков переписывания термов [23]. В рамках указанных примеров демонстрируется формализация нескольких типовых моделей ЛРД, в том числе ряда дискреционных и ролевых моделей, на основе языков декларативного программирования. В качестве методов анализа моделей ЛРД в подобной формализации авторы указанных подходов предлагают использовать аппарат доказательства свойств программ на языках логического программирования.

3. Способы задания и проверки свойств безопасности в моделях логического разграничения доступа. Представим ряд способов спецификации и проверки свойств в моделях ЛРД. Для каждого из рассматриваемых способов кратко указываются подход к формальному описанию проверяемых свойств и метод их проверки. Способы сгруппированы по применяемым математическим методам доказательства свойств.

Методы поиска и проведения преобразований в графах. В случае использования конструкций из теории графов для описания моделей ЛРД целесообразно к исследованию таких моделей применить методы поиска и преобразования графов. Как правило, задаваемые свойства связаны с наличием пути между некоторыми вершинами графа. Выделим две типовые формы спецификации свойств на графах в применении к исследо-

ванию моделей ЛРД: 1) наличие пути из одной заданной вершины в графе в другую; 2) возможность преобразования графа таким образом, что в итоговом графе будет существовать путь между двумя заданными вершинами. Указанные свойства, как правило, дополняются ограничениями на искомый путь в графе, например за счет раскраски вершин и наличия пометок на ребрах графа. В случае использования для описания модели ЛРД графа, вершины которого соответствуют сущностям модели – субъектам и объектам доступа, а ребра – доступам или элементарным информационным потокам, такой подход к формулировке свойств позволяет задавать ряд ограничений на информационные потоки в исследуемой модели ЛРД. Спецификация допустимых преобразований графа позволяет выделить операции по изменению статической конфигурации модели ЛРД. Для указания таких свойств используется аппарат графовых грамматик [24, 25].

Проверка свойств в моделях ЛРД, заданных некоторым графовым описанием, как правило, проводится с помощью традиционных алгоритмов на графах [25].

Верификация на модели. Методы верификации на модели (model checking) основаны на задании модели как системы переходов между состояниями ("машины состояний"), для каждого из которых задан набор истинных в нем атомарных высказываний пропозициональной логики. Проверяемое свойство специфицируется в виде формулы временной логики, налагающей ограничения на состояния в путях в заданной системе переходов. В зависимости от используемого языка временной логики некоторым образом ограничивается последовательность состояний в возможных путях [26].

В применении к анализу свойств в моделях ЛРД следует отметить ряд подходов с использованием методов верификации на модели [12, 13, 27]. Как правило, подобные подходы ориентированы на исследование свойств информационных потоков в моделях ЛРД.

Выделяются два подхода к формированию системы переходов. Первый из них основан на использовании субъектов и объектов доступа в модели ЛРД в качестве состояний в системе переходов [12, 13]. Такие состояния прямо соответствуют либо сущностям в модели ЛРД, либо их группам. Переходы между состояниями соответствуют доступам в модели ЛРД или элементарным информационным потокам. При использовании такого подхода спецификация свойств и их проверка являются расширениями методов, основанных на поиске путей в графе. В этом случае использование методов верификации на модели позволяет систематизировать формулировку свойств как формул временной логики и предоставляет ряд оптимизаций по их проверке по сравнению с традиционными алгоритмами поиска пути в графе.

В работе [27] представлен подход к заданию системы переходов, в котором в качестве состояния рассматривается набор атрибутов для всех сущностей в модели. Переходам соответствует допустимое изменение таких атрибутов. Дополнительно определяется предикат, который определяет, разрешен ли некоторый доступ в модели ЛРД в зависимости от атрибутов субъектов и объектов доступа. В такой модели проверяемые свойства содержательно соответствуют утверждениям о том, можно ли при заданных способах изменения атрибутов (т. е. при заданных переходах) сделать предикат предоставления доступа истинным для некоторого заданного доступа. Таким образом, рассматриваемый подход позволяет исследовать модели ЛРД с изменяющимися свойствами субъектов и объектов доступа, в том числе когда такие изменения являются результатом изменения конфигурации модели ЛРД. В качестве усложнения данного подхода можно предложить использование правил разграничения доступа как части состояния системы переходов. В этом случае становится возможной проверка свойств в моделях ЛРД с изменением правил доступа. Однако следует отметить, что вследствие увеличения количества состояний применимость данного подхода на практике может быть ограничена.

Применение методов верификации на модели в контексте исследования моделей ЛРД на практике продемонстрировано в задачах анализа настроек механизмов разграничения доступа в операционных системах [12, 13]. Наглядным примером в этой области является задача проверки свойств в модели Type Enforcement, используемой в механизме разграничения доступа Security-Enhanced Linux. Для спецификации модели ЛРД и ее свойств использовался первый из рассмотренных выше подходов, при котором состояния системы переходов задавались на основе субъектов и объектов доступа в модели. Программные эксперименты показали возможность проведения эффективной по времени проверки типовых моделей ЛРД с количеством субъектов и объектов доступа до 100 тысяч, что соответствует распространенным конфигурациям механизмов ЛРД в современных ОС.

Автоматическое доказательство теорем. Рассмотренный выше способ проверки свойств моделей ЛРД с использованием методов верификации на модели является одним из примеров применения методов авто-

математического доказательства теорем. По сравнению с другими такими методами верификация на модели подразумевает использование значительно упрощенного языка спецификации исследуемых моделей и проверяемых в них свойств. С одной стороны, используемые в методах верификации на модели логические средства позволяют разработать унифицированный и достаточно эффективный алгоритм проверки свойств. С другой стороны, выразительные свойства как языка спецификации модели, так и языка, на котором задаются проверяемые свойства, могут оказаться недостаточными, например в случае исследования динамических свойств модели ЛРД или анализа моделей ЛРД со сложными правилами принятия решения о доступе. В ряде подобных случаев удается создать альтернативную таким методам логическую модель, исследование свойств которой может быть осуществлено средствами автоматического доказательства теорем. В качестве примеров использования таких средств отметим подход к проверке свойств вложения моделей ЛРД (в том числе решение отмеченной ранее задачи об уточнении "грубой" модели) на основе сведения к задаче о выполнимости булевой формулы [21] и подходы к созданию логического аппарата для спецификации и анализа моделей ЛРД, заданных как программы на языках логического программирования [10, 11, 28]. Следует отметить, что подходы с использованием средств автоматического доказательства теорем демонстрируются, как правило, на небольших учебных примерах, а подходы к их эффективному применению для анализа достаточно объемных моделей ЛРД, которые используются, например, в механизмах операционных систем, предстоит разработать.

Роль и место проверки свойств в автоматизации процесса управления настройками механизмов безопасности. Перечисленные ранее подходы к статической проверке выполнения требований, предъявляемых к конфигурации механизмов логического разграничения доступа, в рамках некоторых ограничений позволяют проводить оценку корректности заданных правил ЛРД. Следует отметить, что такая оценка может производиться как в автономном режиме – вне процесса эксплуатации исследуемой автоматизированной системы, так и непосредственно в рамках ее использования. Первый случай характерен для проведения экспертной оценки состояния защищенности АС по собранным данным о ее конфигурации. Во втором случае примером использования методов проверки свойств в моделях ЛРД является оценка изменений, вносимых в конфигурацию механизмов логического разграничения доступа. В обоих случаях главной особенностью применяемых методов является возможность проведения оценки конфигурации механизмов ЛРД в целом, а не в отношении отдельных правил, заданных в конфигурации.

Следует отметить, что эффективность рассмотренных методов проверки свойств в моделях ЛРД по времени, затраченному на проверку, продемонстрирована только для методов верификации на модели и для методов поиска пути в графе в случае более простых свойств. В задачах проверки свойств моделей ЛРД, используемых в компонентах современных автоматизированных систем, следует ожидать большого объема исследуемых моделей, особенно в случае распределенных систем, в которых объединяются несколько моделей ЛРД. Указанные обстоятельства свидетельствуют о необходимости решения задач оптимизации описаний моделей ЛРД и алгоритмов проверки свойств в них с целью применения к анализу крупных автоматизированных систем.

Кроме задачи непосредственно проверки свойств в моделях ЛРД в процессе исследования конфигурации механизмов логического разграничения доступа следует отметить ряд смежных задач, а именно задачи формирования проверяемых свойств и интерпретации результатов проверки. Таким задачам, как правило, уделяется меньше внимания, тем не менее в подходах проверки свойств моделей ЛРД, ориентированных на практическое применение, важность данных задач очевидна. Эффективность задания типовых проверяемых свойств и возможности по использованию результатов проверки в корректировке настроек и в вынесении экспертной оценки об уровне защищенности исследуемой автоматизированной системы играют важную роль в процессе управления настройками механизмов защиты.

Заключение. Представленный обзор подходов, используемых при описании моделей логического разграничения доступа, выделения и проверки свойств в таких моделях, свидетельствует о наличии в этой области значительного числа результатов теоретического характера. Актуальность проведения подобных исследований по разработке подходов к анализу корректности конфигурации средств защиты, в том числе механизмов логического разграничения доступа, подтверждается необходимостью достижения высоких уровней защищенности в некоторых классах автоматизированных систем. Следует отметить, что получаемые результаты редко затрагивают практические аспекты проведения проверки корректности настроек, а именно

учет всех функциональных возможностей механизмов ЛРД и эффективность алгоритмов проведения проверки в случае анализа больших объемов настроек. Тем не менее некоторые из рассмотренных подходов содержат ряд перспективных способов проверки, обладающих потенциалом в области исследования дискретных структур, соответствующих моделям ЛРД. К таким способам в первую очередь следует отнести верификацию на модели. С развитием подобных методов проверки свойств, а также подходов к их практическому применению следует ожидать их внедрение в процессы администрирования автоматизированных систем, а именно в процесс управления настройками механизмов защиты включая средства разграничения доступа.

Список литературы

1. ВАСЕНИН В. А., ШАПЧЕНКО К. А., АНДРЕЕВ О. О. Математические модели и механизмы логического разграничения доступа в операционной системе Linux: текущее состояние и перспективы развития // Материалы II Междунар. науч. конф. по проблемам безопасности и противодействия терроризму. Пятая общероссийская научная конф. "Математика и безопасность информационных технологий" (МаБИТ-06). Москва, 25-26 октября 2006 г. М.: МЦНМО, 2007. С. 159-171.
2. ГАЛАТЕНКО В. А. Основы информационной безопасности: Курс лекций. М.: Интуит, 2008.
3. SANDHU R., SAMARATI P. Access control: principles and practice // IEEE Communications. 1994. V. 32, N 9.
4. SCHUMACHER M. Security patterns: integrating security and systems engineering / M. Schumacher, E. Fernandez-Buglioni, D. Hybertson, F. Buschmann, P. Sommerlad. John Wiley and Sons, 2006.
5. OTT A. Mandatory rule set based access control in Linux. A multi-policy security framework and role model solution for access control in networked Linux systems. Shaker Publ., 2007.
6. NYANCHAMA M., OSBORN S. Access rights administration in role-based security systems // Database Security VIII: Status and Prospects. North-Holland, 1994. P. 37-56.
7. SANDHU R. Role-based access control // Advances in Computers. V. 46. Acad. Press, 1998.
8. SANDHU R., COYNE E. J., FEINSTEIN H. L., YOUAN C. E. Role-based access control models // IEEE Computer. 1996. V. 29, N 2. P. 38-47.
9. FISLER K., KRISHNAMURTHI S., MEYEROVICH L. A., TSCHANTZ M. C. Verification and change-impact analysis of access-control policies // Proc. of the Intern. conf. on software engng. May 2005. P. 196-205.
10. BERTINO E., CATANIA B., FERRARI E., PERLASCA P. A logical framework for reasoning about access control models // ACM Trans. Inf. Syst. Secur., 2003. V. 6. P. 71-127.
11. DOUGHERTY D. J., FISLER K., KRISHNAMURTHI S. Specifying and reasoning about dynamic access-control policies // Lecture Notes in Computer Sci. Springer-Verlag, 2006. P. 632-646.
12. ШАПЧЕНКО К. А. К вопросу о средствах ОС Linux для управления доступом при использовании ролевых политик безопасности // Материалы конф. "Математика и безопасность информационных технологий", Москва, 2-3 ноября 2005 г. М.: МЦНМО, 2006. С. 257-281.
13. GUTTMAN J. D., HERZOG A. L., RAMSDELL J. D., SKORUPKA C. W. Verifying information flow goals in security-enhanced Linux // J. Comput. Security. 2004. V. 13.
14. LI N., TRIPUNITARA M. On safety in discretionary access control // Proc. of the IEEE symp. on security and privacy, May 2005.
15. BELL D., LAPADULA L. J. Secure computer systems: Mathematical foundations and model // Techn. Rep. M74-244. The Mitre Corporation, 1976.
16. SANDHU R. Lattice-based access models // IEEE Computer. 1993. V. 26, N 11. P. 9-19.
17. JONES A., LIPTON R., SNYDER L. A linear time algorithm for deciding security // Proc. of the 17th Annual symp. on the foundations of computer science, Oct. 1976. P. 33-41.
18. SANDHU R., BHAMIDIPATI V., MUNAWER Q. The ARBAC97 model for role-based administration of roles // ACM Transactions on Information and Systems Security (TISSEC). 1999. V. 2.
19. MOSES T. eXtensible Access Control Markup Language (XACML) version 1.0 // Techn. Rep. OASIS, 2003.
20. OASIS Security Services (SAML) TC Public Documents [Electron. resource] / http://www.oasis-open.org/committees/documents.php?wg_abbrev=security.
21. HUGHES G., BULTAN T. Automated verification of access control policies // Techn. Rep. 2004-22. Santa Barbara: University of California, 2004.
22. BARKER S., STUCKEY P. J. Flexible access control policy specification with constraint logic programming // ACM Trans. Inf. Syst. Secur. 2003. V. 6.

23. BARKER S., FERNÁNDEZ M. Term rewriting for access control // Proc. of the DBSec2006. V. 4127 in LNCS. Springer-Verlag, 2006. P. 179-193.
24. KOCH M., MANCINI L. V., PARISI-PRESICCE F. Decidability of safety in graph-based models for access control // Proc. of the Europ. symp. on research in computer security. 2002. P. 243-299.
25. Handbook of graph grammar and computing by graph transformation. V. 1. Foundations / Ed. by G. Rozenberg. World Scientific, 1997.
26. CLARKE E. M., GRUMBERG O., LONG D. E. Model checking and abstraction // ACM Trans. on Program. Languages and Systems. 1994. V. 16, N 5. P. 1512-1542.
27. GUELEV D. P., RYAN M. D., SCHOBENS P.-Y. Model-checking access control policies // In Information Security Conf. N 3225 in Lecture Notes in Computer Sci. Springer-Verlag, 2004.
28. SARNA-STAROSTA B, STOLLER S. D. Policy analysis for security-enhanced Linux // Proc. of the Workshop on Issues in the Theory of Security, April 2004. P 1-12.
29. KOCH M., PARISI-PRESICCE F. Describing policies with graph constraints and rules // Proc. of the ICGT02. Springer-Verlag, 2002. P. 223-238.

*Кирилл Александрович Шапченко – ст. науч. сотр. Ин-та проблем информационной безопасности
МГУ им. М. В. Ломоносова, тел.: (903) 641-37-10; e-mail: shapchenko@iisi.msu.ru*