Математические структуры и моделирование 2026. № 1 (77). С. 63–68

УДК 004.056
DOI 10.24147/2222-8772.2026.1.63-68

О ВОЗМОЖНОСТИ УПРАВЛЕНИЯ РОЛЕВОЙ ПОЛИТИКОЙ БЕЗОПАСНОСТИ ПОЛНОМОЧИЯМИ TAKE И GRANT

Д.М. Арушанян
магистрант, e-mail: arushanyandm@mail.ru
С.В. Усов
к.т.н., доцент, e-mail: raintower@mail.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Предложена синтетическая модель, базирующаяся на ролевой модели безопасности, в качестве инструментов администрирования которой выступают полномочия take и grant, аналогичные эксплуатирующимся в одноимённой дискреционной модели. Выявлены виды ролевой иерархии, допускающие возможность такого синтеза.

Ключевые слова: компьютерная безопасность, ролевые модели разграничения доступа, иерархия ролей, модель Take-Grant, администрирование ролей.

Введение

Существующие модели, такие как дискреционный контроль доступа (DAC), мандатный контроль доступа (MAC) и ролевой контроль доступа (RBAC), получили широкое распространение. Несмотря на свои преимущества, каждая из этих моделей имеет заметные ограничения. DAC может быть ресурсоёмким, требуя детального управления разрешениями для множества субъектов и объектов. MAC, несмотря на высокую степень защиты, не обладает достаточной гибкостью для адаптации к различным операционным потребностям. RBAC, хотя и эффективен для упрощения управления разрешениями, может не соответствовать подробным и специфическим требованиям к доступу.

Чтобы преодолеть эти трудности, существует потребность в унифицированном подходе, который сочетает в себе сильные стороны традиционных моделей и в то же время смягчает их недостатки. Данная работа посвящена синтезу моделей контроля доступа, в частности, интеграции дискреционного подхода с принципами контроля доступа на основе ролей. Полученная в результате работы модель призвана обеспечить более гибкое решение для управления правами доступа.

Цель данной работы заключается в разработке ролевой модели разграничения доступа с механизмами, аналогичными тем, что присутствуют в модели Take-Grant.

1. Синтез гибридной модели разграничения доступа

Будем рассматривать базовую ролевую модель (RBAC) со множеством ролей R и общим множеством полномочий P [1]. Напомним, что каждой роли в RBAC со-

поставляется некоторое подмножество полномочий.

При необходимости, в зависимости от решаемой моделью задачи, на множестве ролей может быть введена иерархия, отражающая принцип наследования полномочий. Напомним три следующих подхода к организации структуры ролей:

1) строго таксономический листовой подход: при данном подходе полномочия назначаются листовым ролям, причём у каждой листовой роли в данном случае полномочия не пересекаются с остальными (каждое сопоставленное листовой роли полномочие уникально), а полномочия ролей-родителей формируются как объединение полномочий их потомков;
2) нетаксономический листовой подход: при данном подходе используются те же принципы, что и при строго таксономическом листовом подходе, но наборы полномочий листовых ролей могут пересекаться;
3) иерархический охватный подход: наследование в данном подходе происходит по тем же принципам, что были описаны выше, с тем только различием, что у родительских ролей допускается наличие полномочий, отсутствующих у их потомков.

Основная идея синтеза новой модели (в дальнейшем будем называть её RTGM – Role-based Take-Grant Model) состоит в добавлении к ролевой модели дополнительных администрирующих полномочий *take* и *grant*, которые будут аналогичны одноименным правам в дискреционной модели Take-Grant [2] для возможности альтернативного администрирования.

Введём для нашей модели следующий список требований, которым она должна соответствовать:

- модель базируется на ролевой модели (RBAC);
- набор полномочий, который принадлежит той или иной роли, модифицируется благодаря управляющим полномочиям *take* и *grant*, источником для которых служат одноимённые права доступа модели Take-Grant.

В дальнейшем, для удобства, будем обозначать представленные выше требования $(*)$.

В общем случае компьютерная система будет представлена следующей четвёркой $(P^*,\ R,\ \Gamma,\ G)$, где:

- P^* – множество классических полномочий P, дополненное специальными администрирующими полномочиями *take*, *grant*, *create*, *delete* и *destroy* (далее будем обозначать эти полномочия соответственно t, g, c, l и d). Полномочие *take* позволяет заимствовать полномочие у целевой роли, *grant* – вручать полномочие целевой роли, *create* – создавать новую роль, *delete* – отзывать полномочия целевой роли, наконец, *destroy* – удалять целевую роль. Последние три полномочия введены на перспективу, как правило, мы будем работать с усечённой версией модели, в которой множество P дополняется лишь полномочиями *take* и *grant*.

  В силу того, что RTGM берет механизмы администрирования из модели Take-Grant, в которой существуют понятия объектов, субъектов и прав доступа, под элементарным полномочием можно понимать пару $p = [o,\ \alpha]$, где o это объект, а α – набор прав доступа к нему. Администрирующие полномочия в

некоторых случаях могут аналогично определяться парой $t = [t, r]$ в случае с *take* и $g = [g, r]$ в случае с *grant*, где $\{t, g\}$ представляют собой способы взаимодействия, которые распространяются на роль r;

- R – множество ролей, где каждая роль определяется набором полномочий, которые ей доступны;
- $\Gamma(R, E, \{t, g\})$ – tg-граф. Вершинами данного графа являются роли, и если роль r обладает полномочием t на роль r', то в графе существует дуга от r к r', подписанная полномочием t. Аналогичные дуги существуют и для полномочия g. Данный граф позаимствован из модели Take-Grant с тем лишь отличием, что в качестве вершин вместо объектов и субъектов выступают роли, а полномочия (за исключением *take* и *grant*) поставлены в соответствие не дугам, а вершинам графа.
- $G(R, T)$ – ориентированное дерево иерархии ролей, где R – это множество ролей системы, которые являются узлами дерева, T – множество дуг, направленных от родительских ролей к наследникам в рамках иерархической структуры ролей системы.

Множество пользователей и множество сеансов, которые присутствуют в классической ролевой модели, выведены за рамки данной работы, т. к. мы будем рассматривать модификацию иерархии ролей вне привязки к сеансам и пользователям; можно считать, что модификация происходит в рамках одного сеанса.

В силу наличия иерархии и графа иерархии введём отношение частичного порядка на множестве ролей:

$$F_{RR} : R \times R,$$

которое определяет иерархию ролей и задаёт оператор $\geqslant$, такой что если для $r_1, r_2 \in R$, $r_1 \geqslant r_2$, то r_1 находится в иерархии ролей выше чем r_2.

Управление доступом в системе осуществляется в соответствии с отображением

$$F_{RP} : R \longrightarrow P^*,$$

где значением функции $F_{RP}(r)$ является множество полномочий, которыми обладает роль r.

Состояния подсистемы безопасности изменяются под воздействием команд пяти типов.

- Команда «Брать» (Take(p, x, y)) – роль x наделяется полномочием p, относящимся к роли y ($p \in F_{RP}(y)$). Команда осуществима, только если роль x обладает полномочием t;
- Команда «Давать» (Grant(p, x, y)) – роль y наделяется полномочием p, относящимся к роли x ($p \in F_{RP}(x)$). Команда осуществима только если роль x обладает полномочием g;
- Команда «Создать» – Create(α, x, y, β), где $\alpha \subseteq \{t, g\}$ – роль x создаёт роль y, обладающую множеством полномочий β, и получает полномочия α на неё ($\beta \subseteq F_{RP}(x)$). Команда осуществима, только если роль x обладает полномочием c;
- Команда «Удалить» (Delete(β, x, y)) – роль x удаляет подмножество полномочий β роли y ($\beta \subseteq F_{RP}(y)$). Команда осуществима, только если роль x обладает полномочием l;

- Команда «Уничтожить» (Destroy$(x,\ y)$) – роль x удаляет роль y. Команда осуществима, только если роль x обладает полномочием d.

Усечённая версия RTGM содержит только первые две команды, в дальнейшем (если не оговорено обратное) будет рассматриваться именно усечённая модель.

Возможность исполнения команд «Брать» и «Давать» зависит от существующей иерархии на множестве ролей, а также от конкретного способа реализации, что более подробно будет описано в последующих работах, освещающих отдельные случаи реализации модели. Но при наличии иерархии ролей исполнение никакой команды не может приводить к случаю нарушения целостности иерархии с точки зрения полномочий, а именно наличия у роли-потомка полномочия, отсутствующего у роли-родителя. Кроме того стоит отметить, что обе команды привязаны к ролям, т. е. команда «Брать» привязана к роли, у которой происходит похищение полномочия, а у команды «Давать» – к роли, которой мы передаём полномочие.

2. Согласование модели RTGM с иерархией ролей

Модель, представленная в работе, берет за основу ролевую модель, в связи с чем она оперирует полномочиями и ролями. В свою очередь, в ролевых моделях возможна организация ролей в рамках иерархической структуры, в связи с чем возникает необходимость установить, при каких видах иерархии возможна реализация модели RTGM.

Теорема 1. *Реализация RTGM с учётом требований* $(*)$ *осуществима в случае:*

1) отсутствия иерархии;

2) иерархического охватного подхода.

и невозможна в случае:

1) строго таксономического листового подхода;

2) нетаксономического листового подхода.

Доказательство. Заметим, что сама передача полномочий не вызывает противоречий в базовой ролевой модели. Противоречия возникают при внедрении иерархии, а именно при ситуации, когда родительская роль получает полномочие, которого нет ни у одной дочерней роли. Такая ситуация противоречит организации листовых подходов. Из этого следует, что только наличие иерархии накладывает ограничение на внедрение и применение администрирующих полномочий t и g.

В графе иерархии либо существует одна ветвь, либо несколько. Случай с одной ветвью – вырожденный, т. к. в рамках листовых подходов у всех ролей единственной ветви будут одинаковые наборы полномочий, определённые полномочиями листовой вершины этой ветви. Поэтому, без ограничения общности, рассмотрим случай наличия хотя бы двух ветвей в рамках следующего примера.

Пусть дан произвольный граф иерархии G и tg-граф Γ (см. рисунок 1) со следующим распределением полномочий по ролям системы:

$$F_{RP}(r_{111}) = \{p_1\};\ F_{RP}(r_{112}) = \{p_2\};\ F_{RP}(r_{121}) = \{p_3,\ t\};\ F_{RP}(r_{122}) = \{p_4\}.$$

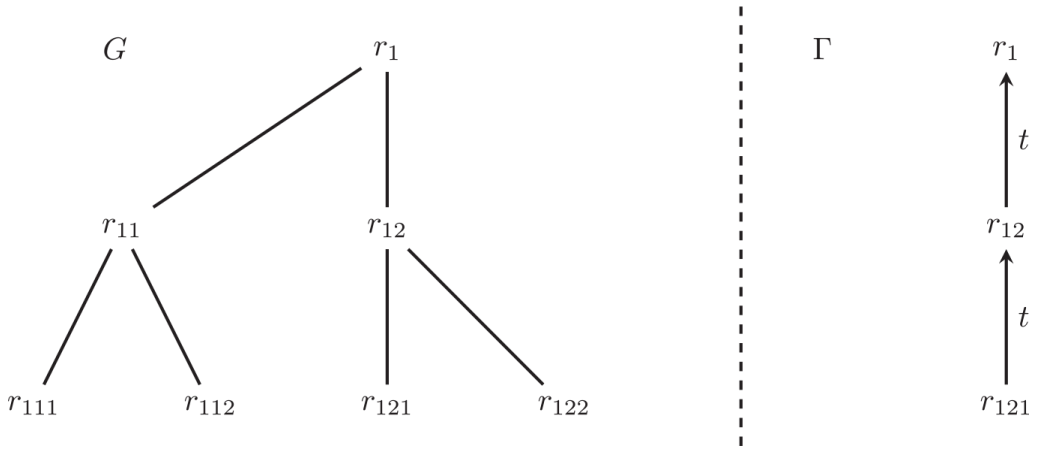

Рис. 1. Граф G_0 и Γ_0

Пусть мы используем строго таксономический или нетаксономический листовые подходы, которые подразумевают, что полномочия старших ролей являются объединением полномочий всех дочерних ролей, принадлежащих этой роли. Из этого следует, что

$$\{t\} \subseteq F_{RP}(r_{12});\ \{t\} \subseteq F_{RP}(r_1);$$

Роль r_{12} может реализовать команду Take$(p_1,\ r_{12},\ r_1)$, тем самым присвоить полномочие p_1, которое принадлежит её родительской роли r_1, тогда:

$$F_{RP}(r_{12}) = \{p_3,\ p_4,\ p_1,\ t\}.$$

Из этого следует, что

$$F_{RP}(r_{121}) \cup F_{RP}(r_{122}) \neq F_{RP}(r_{12}).$$

После применения Take$(p_1,\ r_{12},\ r_1)$ роль r_{12} обладает полномочием p_1, которое отсутствует у её потомков-листьев, что нарушает нетаксономический листовой подход и строго таксономический листовой подход, т. к. в данных подходах полномочия ролей-родителей являются объединением полномочий ролей-потомков, следовательно, множество полномочий всех листовых ролей должно совпадать с множеством полномочий роли-предка.

Приведённый пример демонстрирует ситуацию, когда в графе иерархии есть роль, которая может применить команду Take к роли, обладающей полномочием, которого нет у её потомков-листьев. В примере Take применяется ролью-похитителем к своему родителю, обладающему вторым наследником, но точно такой же эффект вызовет попытка применения команды Take к роли из другой ветви, если такое допустимо в рамках модели.

В результате исполнения команды нарушается строго таксономический листовой подход и нетаксономический листовой подход, но при этом подобная передача прав не создаёт противоречий при иерархически охватном подходе, т. к. в нем допускается наличие у родительской роли полномочия, которого нет у роли-потомка. ■

В последующих работах предполагается рассмотрение трёх возможных вариантов организации администрирования с помощью полномочий t и g:

1) команды Take и Grant используются в отношении всех полномочий, причём Take может быть применена только в отношении родительской роли, а Grant – только в отношении дочерней роли;
2) каждое полномочие t (или g) строго привязано к роли, по отношению к которой оно допускает применение соответствующей команды Take (или Grant), и согласовано с иерархией ролей;
3) внедрение команд Take и Grant рассматривается без использования иерархической структуры в рамках модели.

Литература

1. Ferraiolo D.F., Kuhn D.R. Role-Based Access Control // 15th National Computer Security Conference. October 1992. P. 554–563.
2. Jones A., Lipton R., Snyder L. A Linear Time Algorithm for Deciding Security // Proceedings of the 17th Annual Symposium on the Foundations of Computer Science. 1976.

ON THE POSSIBILITY OF MANAGING A ROLE-BASED SECURITY POLICY USING TAKE AND GRANT PERMISSIONS

D.M. Arushanyan
Master's Degree Student, e-mail: arushanyandm@mail.ru
S.V. Usov
Ph.D. (Techn.), Associate Professor, e-mail: raintower@mail.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. In this paper, the possibility of synthesizing a hybrid access control model by integrating the role-based access control model with the discretionary Take-Grant model is considered. Role-based access control models augmented with administrative take and grant rights are examined. The resulting Role-based Take-Grant Model (RTGM) is formalized. The compatibility of the proposed model with various types of role hierarchies is analyzed. The feasibility of implementing the RTGM in systems without a role hierarchy and in those with a hierarchical coverage approach is proven. The contradictions arising from the application of the proposed model to strictly taxonomic and non-taxonomic leaf-based approaches are demonstrated.

Keywords: Role-based access control model, discretionary access control model, Take-Grant model, role hierarchy, administration rights.

Дата поступления в редакцию: 02.03.2026