

Использование теоретико-графового подхода для решения задачи Roles Mining

С.В. Белим
sbelim@mail.ru

А.Н. Мироненко
mironim84@mail.ru

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

Аннотация

Проблема Roles Mining состоит в построении ролевой политики безопасности для больших информационных систем на основе анализа набора разрешенных полномочий пользователей. В статье разработан алгоритм анализа матрицы разрешенных полномочий пользователей для оптимального формирования ролей. Набор разрешений для отдельного пользователя рассматривается как транзакция. В качестве записи транзакций используются полномочия на действия в системе. После чего применяется алгоритм поиска ассоциативных правил для набора полномочий. Рассматриваются только ассоциативные правила, включающие одно полномочие как предпосылку и одно как следствие. Строится граф взаимосвязи полномочий в данной системе. Данный граф является ориентированным взвешенным. Вес дуги определяется поддержкой соответствующего ассоциативного правила. Для построенного графа выполняется кластеризация с помощью алгоритма поиска сообществ (community). Строится иерархия сообществ. Полномочия объединяются в роли, если они входят в одно сообщество. Иерархия сообществ определяет иерархию ролей. Предложенный алгоритм может быть использован, как для построения ролевой политики безопасности, так и для оптимизации уже существующей иерархии ролей.

Введение

Проблема построения ролевой политики безопасности в компьютерных системах сводится к двум основным задачам: объединение полномочий на работу в системе в роли и сопоставление ролей пользователям. В конечном итоге, в вопросах безопасности системы важную роль играет только сопоставление пользователям требуемых полномочий. Роли являются вспомогательными сущностями. Одних и тех же разрешений для пользователей можно добиться с помощью различного набора ролей. Введение ролей может существенно облегчить администрирование системы. Однако при неверном построении политики ролевого разграничения доступа администрирование может не только усложниться, но и привести к утечке полномочий.

Традиционный подход к построению ролевой политики безопасности базируется на анализе бизнес-процессов организации, в которой функционирует информационная система. Такой подход получил название нисходящего проектирования ролей.

Copyright © by the paper's authors. Copying permitted for private and academic purposes.

In: Sergey V. Belim, Nadezda F. Bogachenko (eds.): Proceedings of the Workshop on Data, Modeling and Security 2018 (DMS-2018), Omsk, Russia, October 2018, published at <http://ceur-ws.org>

После учета всех бизнес процессов часто возникает задача оптимизации структуры ролей. Приведение ролевой иерархии к оптимальному виду может существенно снизить трудоемкость анализа рисков утечки информации. Существует несколько различных постановок оптимизационной задачи для ролевой политики безопасности. Прежде всего оптимизируется дерево ролей. В статьях [1, 2] оптимальность ролевого графа оценивалась с учетом его древовидности, таксономичности, числа узлов или дуг. Оптимальность ролевого графа также может рассматриваться в контексте администрирования подсистемы безопасности [3]. В этом случае может быть выделено три критерия оптимизации: гибкость и масштабируемость, психологическое удобство, экономические механизмы. Особенно актуальны вопросы оптимальности ролей в распределенных системах [4]. В этом случае приходится решать проблему неоднозначности определения ролей. Преобразования ролевого дерева приводят к необходимости разработки алгоритмов сравнения ролевого разграничения доступа, созданного на основе бизнес процессов, с реально существующей политической безопасностью [5]. Одним из подходов к оптимизационной задаче является уменьшение числа ролей в системе [6]. Также при оптимизации ролевого разграничения доступа необходимо выработать механизмы сопоставления ролей пользователям. Один из подходов [7] состоит из нескольких этапов. Сначала каждой роли приписывается вес, для каждого пользователя определяется пороговое значение, для каждого пользователя выбирается наиболее подходящая роль, удовлетворяющая ограничениям.

Однако в современных крупномасштабных информационных системах анализ бизнес-процессов представляется достаточно сложным. Кроме этого бизнес-процессы могут испытывать существенные изменения с течением времени вследствие реорганизации структуры предприятия. В связи с этим получил развитие восходящий принцип проектирования ролевой структуры подсистемы безопасности. В основе этого подхода лежит сопоставление пользователям системы полномочий. Роли формируются таким образом, чтобы удовлетворять некоторым оптимизационным принципам. Данный подход принято называть разработкой ролей (Roles Mining). Основным подходом к разработке ролей стала декомпозиция матрицы полномочий пользователей на произведение двух матриц [8, 9, 10, 11]. В статье [12] предложен подход к формированию дерева ролей, основанный на анализе формальных понятий.

Целью данной статьи является построение алгоритма разработки ролей, основанного на поиске ассоциативных правил между полномочиями применительно к конкретной информационной системе. Традиционное использование метода формирования ассоциативных правил из набора данных состоит в исследовании особенностей покупательского спроса на товары в торговых сетях [13, 14]. Информация о взаимосвязи интереса к покупке различных товаров, используется в дальнейшем для их позиционирования в торговых залах. В последнее время наблюдается расширение сферы использования данного метода. В качестве примеров можно привести обработку изображений [15, 16], анализ распространения биологических видов [17], поиск точек интереса горной промышленности по фотографиям поверхности [18] и другие. Достаточно большое число работ посвящено анализу социальных сетей с помощью поиска ассоциативных правил. Так в статье [19] на основе ассоциативных правил выявляется взаимное влияние пользователей социальных сетей.

В данной статье ассоциативные связи используются для выявления взаимосвязи между полномочиями, характерными для конкретной информационной системы. Предполагается, что объединение связанных полномочий в одну роль является целесообразным.

1 Постановка задачи и поиск ассоциативных правил

Будем считать, что задана информация о пользователях системы и полномочиях, которые необходимы каждому пользователю для работы в системе. Обозначим множество пользователей через $U = u_1, \dots, u_n$, множество полномочий - через $P = p_1, \dots, p_m$. Распределение полномочий между пользователями определим отображением $UP : U \rightarrow 2^P$.

Задача построения ролевой политики разграничения доступа состоит в определении множества ролей $R = r_1, \dots, r_s$ и трех отображений:

- $RP : R \rightarrow 2^P$,
- $UR : U \rightarrow 2^R$,
- $RR : R \rightarrow 2^R$.

Отображение RP предоставляет набор полномочий для заданной роли. Отображение UR определяет авторизацию пользователя на некоторый набор ролей. Отображение RR задает авторизацию роли на другие роли.

Пусть множество необходимых полномочий пользователей в компьютерной системе задано в виде таблицы M , строки которой соответствуют пользователям из множества U , а столбцы полномочиям из множества P . В ячейках матрицы M может находиться два значения 0 или 1. $M[i, j] = 1$, если пользователю u_i выделено полномочие p_j , и $M[i, j] = 0$ в противном случае. Задачей ставится задание множества ролей R , распределение полномочий между ними и авторизация пользователей на роли. В качестве ограничения потребуем, чтобы одновременно выполнялись условия минимальности количества ролей и минимальное количество полномочий, выделенных пользователю сверх необходимых.

Проведем анализ таблицы M с целью выявить взаимосвязь полномочий в данной информационной системе. Для этого используем метод выявления ассоциативных правил. В качестве одной транзакции будем рассматривать набор полномочий одного пользователя. Тогда множество транзакций T представляет собой множество строк матрицы M . Соответственно одна транзакция $t_i \in T$ определяется как:

$$t_i = \{M[i, j] \mid j = 1, \dots, |P|, i = 1, \dots, |U|\} \quad (1)$$

Множество ассоциативных правил A формируется из множества транзакций T [13, 14]. Ассоциативным правилом называется импликация $X \rightarrow Y$, где $X \subseteq 2^P, Y \subseteq 2^P$ и $X \cup Y = \emptyset$. В данной работе мы ограничимся только случаем, когда предпосылка и следствие импликации содержат по одному элементу множества P , поэтому можно записать, что $X \in P, Y \in P$ и $X \neq Y$.

На основе множества транзакций можно вычислить параметры ассоциативных правил. Прежде всего, каждое ассоциативное правило характеризуется величиной поддержкой, которая показывает частоту встречаемости данного правила в имеющемся наборе транзакций. Поддержка ассоциативного правила $X \rightarrow Y$ вычисляется как относительное количество транзакций, содержащих множество $X \cup Y$:

$$supp(X \rightarrow Y) = N(X \cup Y) / |T| \cdot 100\%, \quad (2)$$

где $N(X \cup Y)$ — количество транзакций, содержащих множество $X \cup Y$.

Кроме этого, для каждого ассоциативного правила необходимо вычислить величину достоверности, показывающую вероятность, с которой из X следует Y :

$$conf(X \rightarrow Y) = supp(X \cup Y) / supp(X). \quad (3)$$

Будем искать ассоциативные правила между полномочиями на действия пользователей в информационной системе. Наибольшей поддержкой будут характеризоваться ассоциативные правила между полномочиями, наиболее часто встречающимися у одного пользователя. Достоверность будет показывать вероятность того, что наличие одного полномочия у пользователя требует авторизацию его на второе полномочие. Однако при низкой поддержке достоверность будет малоинформативной. Если в системе имеется $|P|$ полномочий, то общее количество ассоциативных правил будет равно $|P|(|P| - 1)$. Вычислим поддержку и достоверность для каждого ассоциативного правила. Построим два графа — G_S и G_C . Для обоих графов множеством вершин будет служить множество полномочий P . Дугами графов будут служить ассоциативные правила с ненулевой поддержкой. В графе G_S вес дуг будет определяться поддержкой ассоциативного правила, а в графе G_C в качестве веса дуг выберем достоверность.

2 Граф связей между полномочиями и формирование ролей

Проведем анализ построенных графов G_S и G_C с целью выявления подмножеств полномочий наиболее тесно связанных друг с другом в данной информационной системе. Для выявления наиболее тесно связанных вершин в обоих графах применим к ним алгоритм поиска сообществ (community) [20]. Для определения состава сообществ получим матрицу весов в приведенном виде $e = E/m$, где $m = \sum E_{ij}$, N — количество вершин. Элемент e_{ij} равен относительный вес заданной дуги в общем весе графа. Очевидно, что выполняется следующее равенство $\sum e_{ij} = 1$. Любой граф может быть разбит на подграфы большим числом способов. Для сравнения правильности разбиения вводится функция модульности (modularity), зависящая от весов вершин и дуг:

$$Q(e) = \sum e_{ii} - \sum p_{ii}, \quad (4)$$

где p_{ii} — «ожидаемая связность», которая может быть определена через исходящую степень вершины a_i , и входящую степень вершины b_i : $p_{ii} = a_i b_i$. Учитывая это равенство функция модульности может быть записана в следующем виде:

$$Q(e) = \sum e_{ii} - \sum a_i b_i, \quad (5)$$

где $a_i = \sum e_{ij}$, $b_i = \sum e_{ji}$.

Поиск сообществ на графах будем осуществлять с помощью алгоритма образования стяжек. Пусть в графе G выделен подграф G' . Заменим подграф G' одной вершиной, оставив без изменения все остальные вершины из $G \setminus G'$. Свяжем новую вершину дугами с вершиной графа $G \setminus G'$ дугой, если существовала хотя бы одна дуга из вершины, входящей в G' к этой вершине из $G \setminus G'$. Если две вершины связывает несколько дуг, то их веса складываются. Изначально все вершины графа G имеют нулевой вес. После образования стяжки новой вершине приписывается ненулевой вес, равный сумме весов вошедших в нее вершин и дуг. Новая стяжка считается выгодной, если она увеличивает функцию модульности графа $Q(e)$. Вершины, вошедшие в выгодную стяжку, относятся к одному сообществу.

Для разработки ролей разобьем на сообщества графы G_S и G_C . Выберем те пары полномочий, объединение которых повышает функцию модульности графа. Следует учесть, что одно полномочие может входить в несколько ролей, поэтому необходимо рассмотреть различные варианты объединений в сообщества с участием одних и тех же вершин графов. Следует учесть, что в графе G_S матрица весов является симметричной относительно главной диагонали, тогда как матрица весов графа G_C таким свойством не обладает. В связи с этим использование этих двух графов может приводить к различным результатам.

При выделении сообществ на графах в первую очередь необходимо определить вершины, из которых будут образовываться стяжки. Для этого определим для каждой вершины исходящую дугу наибольшего веса. Объединим в сообщество вершины, соединяемые данной дугой. В результате будут построены роли первого уровня. Применяя последовательно данный подход можно получить иерархию ролей. Следует отметить, что данная процедура неоднозначно и количество уровней ролевого дерева определяется разработчиком политики безопасности.

3 Пример разработки ролей

Рассмотрим пример использования разработанного алгоритма. Пусть матрица M , определяющая сопоставление полномочий пользователям имеет следующий вид табл. 1:

Таблица 1: Матрица полномочий

$U \setminus P$	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
u_1	1	1	1	0	0	0	0	0	1	0
u_2	0	0	0	1	1	1	0	1	1	0
u_3	0	1	1	1	1	0	0	0	1	1
u_4	0	0	0	0	0	1	1	0	0	0
u_5	1	0	1	0	0	1	0	0	1	1
u_6	0	0	0	1	1	1	0	1	0	0
u_7	0	0	0	0	0	1	1	0	0	0
u_8	1	1	1	0	0	1	1	0	1	0
u_9	0	1	1	1	1	0	1	0	1	1
u_{10}	1	1	0	0	0	1	0	1	0	1

Данное отношение может быть представлено в виде двудольного графа, изображенного на рис. 1.

Найдем для всех ассоциативных правил поддержку и запишем их в виде таблицы, в которой строкам соответствуют предпосылки, а столбцам следствия (см. табл. 2). Эта таблица будет играть роль матрицы весов для графа G_S .

Аналогично вычислим достоверность для всех ассоциативных правил и запишем их в виде таблицы (см. табл. 3), которая одновременно будет матрицей весов для графа G_C .

Выделение сообществ на графах G_S и G_C приводит в данном случае к одинаковой структуре ролей: $R_1 = p_2, p_{10}$, $R_2 = p_3, p_9$, $R_3 = p_4, p_5$, $R_4 = p_6, p_7$, $R_5 = p_1, p_2$, $R_6 = p_6, p_8$.

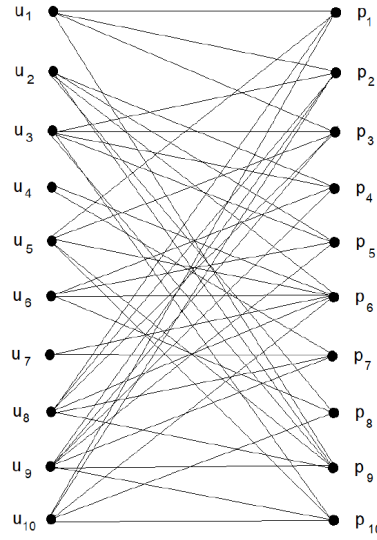


Рис. 1: Представление отображения UP в виде двудольного графа.

Таблица 2: Таблица значений поддержки ассоциативных правил

$X \rightarrow Y$	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
p_1	0	0,3	0,2	0	0	0,3	0,1	0,2	1	0,2
p_2	0,3	0	0,4	0,2	0,1	0,2	0,2	0,1	0,4	0,3
p_3	0,2	0,4	0	0,2	0,1	0,2	0,2	0	0,5	0,3
p_4	0	0,2	0,2	0	0,3	0,2	0,1	0,2	0,3	0,2
p_5	0	0,1	0,1	0,3	0	0,2	0,1	0,2	0,2	0,1
p_6	0,3	0,2	0,2	0,2	0,2	0	0,3	0,3	0,3	0,2
p_7	0,1	0,2	0,2	0,1	0,1	0,3	0	0	0,2	0,1
p_8	0,1	0,1	0	0,2	0,2	0,3	0	0	0,1	0,1
p_9	0,2	0,4	0,5	0,3	0,2	0,3	0,2	0,1	0	0,3
p_{10}	0,2	0,3	0,3	0,2	0,1	0,2	0,1	0,1	0,3	0

Права пользователей определяются через аутентификацию на роли следующим образом: $u_1 = R_2, R_5, u_2 = R_2, R_3, R_6, u_3 = R_1, R_2, R_3, u_4 = R_4, u_5 = R_1, R_2, R_5, R_6, u_6 = R_3, R_6, u_7 = R_4, u_8 = R_2, R_4, R_5, u_9 = R_1, R_2, R_3, R_4, u_{10} = R_1, R_5, R_6$.

Таким образом, предложенный подход позволяет распределить десять полномочий по шести ролям, сохраняя при этом заданное для пользователей распределение полномочий.

Рольевое дерево при этом будет иметь вид, представленный на рис. 2.

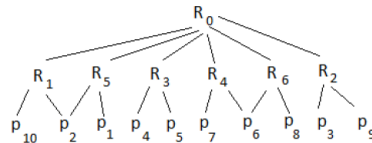


Рис. 2: Одноуровневая иерархия ролей.

Однако возможна двухуровневая иерархия ролей, изображенная на рис. 3.

В этом случае отображения, сопоставляющие ролям полномочия и пользователям роли, будут иметь вид: $RR(R_7) = R_2, R_5, RR(R_8) = R_3, R_6, RR(R_9) = R_2, R_3, RR(R_{10}) = R_1, R_4, UR(u_1) = R_7, UR(u_2) = R_8, R_9, UR(u_3) = R_9, R_{10}, UR(u_4) = R_{10}, UR(u_5) = R_7, R_{10}, UR(u_6) = R_8, UR(u_7) = R_{10}, UR(u_8) = R_7, R_{10}, UR(u_9) = R_9, R_{10}, UR(u_{10}) = R_7, R_8, R_{10}$.

Данный вариант характеризуется меньшим количеством ролей, используемых при авторизации пользователей. Однако при этом возрастает количество избыточных полномочий у отдельных пользователей.

Таблица 3: Таблица значений достоверности ассоциативных правил

$X \rightarrow Y$	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
p_1	0	0,66	0,40	0	0	1	0,33	0,33	0,66	0,66
p_2	0,40	0	0,80	0,40	0,20	0,40	0,40	0,20	0,80	0,60
p_3	0,40	0,80	0	0,40	0,20	0,40	0,40	0	1	0,60
p_4	0	0,50	0,50	0	0,75	0,50	0,25	0,50	0,75	0,50
p_5	0	0,33	0,33	1	0	0,66	0,33	0,66	0,66	0,33
p_6	0,42	0,28	0,28	0,28	0,28	0	0,42	0,42	0,42	0,28
p_7	0,25	0,50	0,50	0,25	0,25	0,75	0	0	0,50	0,25
p_8	0,33	0,33	0	0,66	0,66	1	0	0	0,33	0,33
p_9	0,33	0,66	0,83	0,50	0,33	0,50	0,33	0,16	0	0,50
p_{10}	0,50	0,75	0,75	0,50	0,25	0,50	0,25	0,25	0,75	0

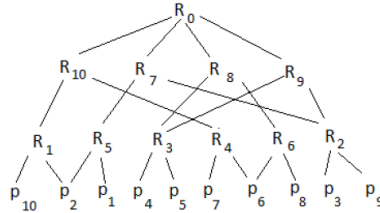


Рис. 3: Двухуровневая иерархия ролей.

Представим построенное отображение UR также в виде двудольного графа (см. рис. 4).

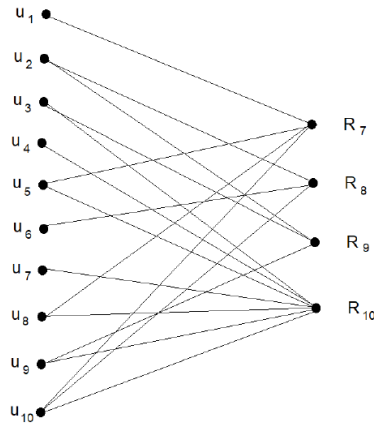


Рис. 4: Представление отображение UR также в виде двудольного графа.

Из сравнения рисунков 4 и 1 можно сделать вывод о снижении трудоемкости анализа системы на наличие утечек полномочий.

4 Заключение

Таким образом, подход, основанный на поиске ассоциативных правил, построении графа взаимосвязи полномочий в данной информационной системе позволяет построить адекватную ролевую политику безопасности. Предложенное решение задачи Roles Mining имеет квадратичную трудоемкость по числу полномочий и пользователей, что позволяет использовать ее для больших информационных систем. В данной работе не рассмотрен вопрос построения иерархии ролей. Он может быть решен в рамках предложенного подхода с помощью итерационной процедуры.

Список литературы

- [1] S.V. Belim, N.F. Bogachenko. Postroenie rolevoj politiki bezopasnosti na proizvol'nom orientirovannom grafe. *Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy*, 3:7–17, 2009.
- [2] S.V. Belim, N.F. Bogachenko, E. Ilushechkin. An analysis of graphs that represent a role-based security policy hierarchy. *Journal of Computer Security*, 23(5):641–657, 2015.
- [3] N. Li, Z. Mao. Administration in Role-Based Access Control. *Proceedings of the 2nd ACM symposium on Information, computer and communications security*, 2007, pp. 127–138.
- [4] M.N. Tahir. Hierarchies in Contextual Role-Based Access Control Model (C-RBAC). *International Journal of Computer Science and Security (IJCSS)*, 2(4):28–42, 2008.
- [5] M. Leitner. Delta Analysis of Role-Based Access Control Models. *Lecture Notes in Computer Science*, 2013, pp. 507–514.
- [6] D. Zhang, K. Ramamohanara, S. Versteeg, R. Zhang. Graph Based Strategies to Role Engineering. *Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research*, article 25, 2010.
- [7] A. Colantonio, R.D. Pietro, A. Ocello, N.V. Verde. Taming role mining complexity in RBAC. *Computers & Security*, 29(5):548–564, 2010.
- [8] H. Lu, J. Vaidya, V. Atluri. Optimal Boolean matrix decomposition: Application to role engineering. *Proceedings of International Conference on Data Engineering (ICDE)*, 2008, pp. 297–306.
- [9] I. Molloy, H. Chen, T. Li, Q. Wang, N. Li, E. Bertino, S. Calo, J. Lobo. Mining roles with semantic meanings. *Proceedings of the 13th ACM symposium on Access control models and technologies*, 2008, pp. 21–30.
- [10] J. Schlegelmilch, U. Steffens. Role mining with ORCA. *Proceeding of Symposium on Access Control Models and Technologies (SACMAT)*, 2005, pp. 168–176.
- [11] J. Vaidya, V. Atluri, Q. Guo. The role mining problem: Finding a minimal descriptive set of roles. *Proceeding of Symposium on Access Control Models and Technologies (SACMAT)*, 2007, pp. 175–184.
- [12] S.V. Belim, N.F. Bogachenko. Ispolzovanie reshetki formalnyh ponyatij dlya postroeniya rolevoj politiki razgranicheniya dostupa. *Informatika i sistemy upravleniya* 1:16–28, 2018.
- [13] R. Agrawal, T. Imielinski, A. Swami. Mining Association Rules between Sets of Items in Large Databases. *Proc. of the 1993 ACM SIGMOD International Conference on Management of Data*, Washington DC, USA, 1993, pp. 207–216.
- [14] M. Shaheen, M. Shahbaz, A. Guergachi. Context Based Positive and Negative Spatio Temporal Association Rule Mining. *Elsevier Knowledge-Based Systems*, 2013, pp. 261–273.
- [15] S.V. Belim, A.O. Mayorov-Zilbernagel. Image Restoration With Static Gaps On The Basis Of Association Rules. *Herald of computer and information technologies*, 12:18–23, 2014.
- [16] S.V. Belim, A.O. Mayorov-Zilbernagel. Algorithm for Searching the Broken Pixels and Eliminating Impulse Noise in Images Using a Method of Association Rules. *Science and Education of the Bauman MSTU*, 12:716–737, 2014. URL: <http://technomag.bmstu.ru/doc/744983.html>.
- [17] M.S. Atepalikhin, B.Yu. Kassal, S.V. Belim. The identification of interrelation of habitation of species by association rules. *Herald of Omsk university*, 2(72):25–29, 2014.
- [18] I. Lee, G. Cai, K. Lee. Mining Points-of-Interest Association Rules from Geo-tagged Photos. *46th Hawaii International Conference on System Sciences*, 2013, pp. 1580–1588.
- [19] F. Erlandsson, P. Brodka, A. Borg, H. Johnson. *Finding Influential Users in Social Media Using Association Rule Learning*, arXiv:1604.08075v2, 2016.

Solution of the Roles Mining Problem with the Help of Graph Theory

Sergey V. Belim, Anton N. Mironenko

For large information systems, there is a problem with Roles Mining. The essence of the problems is that it is necessary to develop a role-based security policy for the system based on an analysis of the authorized powers of users. In this paper, an algorithm is proposed that allows solving this problem. Permissions for each user are delivered in the form of a transaction. A transaction record is the authority to perform actions in the system. The association rules for the set of authorities are being run. Associative rules including one-sidedness as a prerequisite and one as a consequence are considered. The linkage of powers is represented in the form of an oriented weighted graph. The support of the corresponding associative rule determines the weight of the arc. In the received graph, community searches are performed and their hierarchy is built. If the plenipotentiaries enter into one entity, they are united in a role. The hierarchy of roles is determined by the previously constructed hierarchy of communities. Using the proposed algorithm, you can implement both the creation of a role-based security policy and the optimization of an existing hierarchy of roles.