УДК 004.056

© 2018 г. **С.В. Белим,** д-р физ.-мат. наук,
Н.Ф. Богаченко, канд. физ.-мат. наук
(Омский государственный университет им. Ф.М. Достоевского)

ИСПОЛЬЗОВАНИЕ РЕШЕТКИ ФОРМАЛЬНЫХ ПОНЯТИЙ ДЛЯ ПОСТРОЕНИЯ РОЛЕВОЙ ПОЛИТИКИ РАЗГРАНИЧЕНИЯ ДОСТУПА

В статье предложен алгоритм проектирования и создания ролевой политики разграничения доступа на основе использования решетки формальных понятий (решетка концептов). Доказано, что построенная модель удовлетворяет требованиям ролевой политики. Предложенный подход использован для оптимизации существующей ролевой иерархии.
Ключевые слова: ролевое разграничение доступа, анализ формальных понятий, решетка концептов, ролевая иерархия.

DOI: 10.22250/isu.2018.55.16-28

Введение

В связи с ростом объема информации, обрабатываемой информационными системами, возникает проблема администрирования подсистемы безопасности, связанная с необходимостью учета больших массивов данных. Для больших информационных систем эта задача становится ключевой при исследовании возможных каналов утечки информации. Одним из возможных подходов к ее решению является использование мандатной или ролевой политики безопасности. Дискреционная модель безопасности, реализующая произвольное разграничение доступа, позволяет построить практически любую политику безопасности. Однако необходимость определять набор разрешенных и запрещенных доступов для каждой пары «субъект-объект» приводит к квадратичному росту количества настраиваемых параметров при увеличении числа субъектов системы. Ролевая политика безопасности, требующая использования меньшего количества параметров, обладает одновременно меньшей гибкостью в настройках разграничения доступа, в связи с чем актуальной является задача разработки алгоритмов по-

строения ролевой политики безопасности, удовлетворяющей заданному набору требований разграничения доступа.

По своей структуре ролевая политика безопасности опирается на множество ролей, определяемых администратором системы. Внедрение ролевой политики безопасности оправдано в случае небольшой оптимизированной структуры ролевого множества. Множество ролей, с определенным на нем отношением нестрогого порядка, может быть представлено в виде ориентированного графа – ролевого графа. Поэтому задача построения оптимальной ролевой политики безопасности сводится к оптимизации ролевого графа. Оптимизация ролевой иерархии существенно снижает трудоемкость анализа политики безопасности системы. Задача в такой постановке решалась в ряде работ. В статьях [1, 2] рассматривались такие критерии оптимальности ролевого графа как древовидность, таксономичность, минимальное число узлов или дуг. Вопросы преобразования ролевого графа с точки зрения администрирования ролей рассмотрены в статье [3]. Авторы выделяют три критерия оптимизации ролевой иерархии: гибкость и масштабируемость, психологическое удобство, экономические механизмы. В статье [4] исследовано построение и оптимизация ролевой иерархии для пространственных ролей в распределенных системах. Основная цель работы состоит в построении методов устранения неоднозначности определения ролей. В работе [5] разработан дельта-анализ на основе графового подхода, позволяющий сравнивать ролевое разграничение доступа, созданное, согласно предписаниям, с реально существующей политикой безопасности. В статье [6] целью оптимизации ролевой иерархии ставится уменьшение количества ролей. Для решения этой задачи предложено несколько стратегий построения ролевого графа. В статье [7] решается задача оптимального поиска ролей. Предлагается стратегия, состоящая из трех этапов: каждой роли приписывается вес, для каждого пользователя определяется пороговое значение, для каждого пользователя выбирается наиболее подходящая роль, удовлетворяющая ограничениям.

Однако во всех этих работах предполагалось существование уже созданного ролевого графа, который приводился к оптимальному виду с помощью эквивалентных преобразований. В данной статье рассматривается подход, в котором в качестве исходных данных выступает набор требований к обладанию определенных пользователей некоторыми полномочиями. Исходя из этих требований, строится оптимальный ролевой граф. Для построения графа используется анализ формальных понятий (АФП) (Formal Concept Analysis, FCA). Методы АФП впервые были изложены в работах [8 – 10]. Идея АФП основана на построении решетки [11] по произвольному бинарному отношению. Основное направление использования данного метода состоит в поиске взаимосвязей между объектами и их атрибутами. В качестве атрибутов будем рассматривать пользователей, а в качестве

объектов – их полномочия в системе. Наличие взаимосвязей между пользователями и полномочиями позволит построить роли, на которые может быть авторизовано несколько пользователей.

1. Постановка задачи и анализ формальных понятий

Будем считать, что задана информация о пользователях системы и полномочиях, которые необходимы каждому пользователю для работы в системе. Обозначим множество пользователей через $U = \{u_1, \ldots, u_n\}$, множество полномочий – через $P = \{p_1, \ldots, p_m\}$. Распределение полномочий между пользователями определим отображением

$UP: U \to 2^P$.

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

$RP: R \to 2^P$; $UR: U \to 2^R$; $RR: R \to 2^R$.

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.

Ведущую роль в этом наборе играет отображение авторизации ролей друг на друга, которое должно определить иерархию ролей, заданную как отношение нестрогого частичного порядка «$\prec$» на множестве R: $r_j \prec r_i \Leftrightarrow r_j \in RR(r_i)$. Рефлективность, антисимметричность и транзитивность отношения «$\prec$» определяют следующие свойства отображения RR:

1) $r_k \in RR(r_k)$;
2) если $r_l \in RR(r_k) \wedge r_k \in RR(r_l)$, то $r_k = r_l$;
3) если $r_l \in RR(r_k) \wedge r_h \in RR(r_l)$, то $r_h \in RR(r_k)$.

Основные требования ролевой политики разграничения доступа заключаются в том, что отображения RR и RP должны гарантировать выполнение условия *наследования полномочий*:

$(r_j \prec r_i) \Rightarrow (RP(r_j) \subseteq RP(r_i))$,

а отображения RR и UR должны обеспечивать выполнение условия *наследования авторизации*:

$(r_j \prec r_i) \wedge (r_i \in UR(u)) \Rightarrow (r_j \in UR(u))$.

Заметим, что так как множества U, P и R конечны, то все рассматриваемые дискретные отображения могут быть представлены булевыми матрицами по правилу: если $F: X \to 2^Y$, то элемент матрицы $\mathbf{F}[i, j] = 1 \Leftrightarrow y_j \in F(x_i)$.

Традиционно для анализа структуры частично упорядоченного множества, каковым является множество ролей R, используется диаграмма Хассе (графовое представление отношения порядка, являющееся транзитивной редукцией ориен-

тированного графа). В рамках ролевой модели такой орграф назовем ролевым графом [12].

Для построения ролевого графа применим АФП [13–17]. В рамках этого метода данные представляются с помощью *формального контекста* – тройки $K = (O, A, \mathcal{R})$, где O – множество *объектов*, A – множество *атрибутов* (*признаков*) и $\mathcal{R} \subseteq O \times A$ – бинарное отношение, задающее атрибуты объектов. Для произвольных подмножеств $X \subseteq O$, $Y \subseteq A$ определяются *операторы Галуа*:

$$X^{\uparrow} = \{a \in A \mid \forall\, o \in X\colon o\, \mathcal{R}\, a\}; \qquad Y^{\downarrow} = \{o \in O \mid \forall\, a \in Y\colon o\, \mathcal{R}\, a\}.$$

Очевидно, что $X^{\uparrow}$ – это множество общих атрибутов для объектов из X, $Y^{\downarrow}$ – множество объектов, имеющих как минимум атрибуты из Y. Двойное применение операторов Галуа называется *операторами замыкания*. Несложно понять, что операторы замыкания являются экстенсивными:

$$X \subseteq O \Rightarrow X \subseteq X^{\uparrow\downarrow}; \qquad Y \subseteq A \Rightarrow Y \subseteq Y^{\downarrow\uparrow}$$

и монотонными:

$$X_1 \subseteq X_2 \subseteq O \Rightarrow X_1{}^{\uparrow\downarrow} \subseteq X_2{}^{\uparrow\downarrow}, \quad Y_1 \subseteq Y_2 \subseteq A \Rightarrow Y_1{}^{\downarrow\uparrow} \subseteq Y_2{}^{\downarrow\uparrow}.$$

Подмножества объектов $X \subseteq O$ и атрибутов $Y \subseteq A$ называются *замкнутыми*, если $X = X^{\uparrow\downarrow}$, $Y = Y^{\downarrow\uparrow}$ соответственно. Пара (X, Y), где $X \subseteq O$ и $Y \subseteq A$, называется *концептом* (*формальным понятием*) контекста K, если

$$(X = Y^{\downarrow}) \wedge (Y = X^{\uparrow}).$$

При этом X – это *объем*, а Y – это *содержание* концепта. Очевидно, объем и содержание являются замкнутыми подмножествами. Кроме того, несложно заметить, что для любых двух концептов (X_1, Y_1) и (X_2, Y_2) справедливо:

$$X_1 \subseteq X_2 \Leftrightarrow Y_1 \supseteq Y_2.$$

На множестве всех концептов вводится отношение предшествования «$\prec$»:

$$(X_1, Y_1) \prec (X_2, Y_2) \Leftrightarrow X_1 \subseteq X_2\ (Y_1 \supseteq Y_2).$$

Отношение «$\prec$» является отношением частичного порядка. Более того, согласно основной теореме АФП, все концепты формального контекста K, упорядоченные согласно отношению предшествования, образуют решетку, называемую *решеткой концептов* (*решеткой Галуа*).

Визуализацией контекста K является двудольный граф G_K с долями O и A и множеством ребер, определяемым отношением $\mathcal{R}$. Каждому концепту соответствует подграф, представляющий собой максимальную двудольную клику (максимальную в том смысле, что она не содержится целиком ни в какой другой двудольной клике графа G_K).

Другая интерпретация контекста K – булева матрица МК, строкам которой соответствуют объекты, столбцам – атрибуты. Если объект i обладает атрибутом j, то МК[i, j] = 1, иначе МК[i, j] = 0. Очевидно, матрица МК является матрицей смежности графа GK (матрицей отношения $\mathcal{R}$), а каждой максимальной двудоль-

ной клике (каждому концепту) соответствует прямоугольник в матрице МК, заполненный единицами (прямоугольник определяется с точностью до перестановки строк и столбцов).

Ряд алгоритмов построения множества концептов и их решеток рассмотрен в работах [15 – 17]. Существуют программные модули [18, 19] и библиотеки алгоритмов [20] для построения решеток концептов и линейных диаграмм (линейная диаграмма – это диаграмма Хассе решетки концептов с так называемой сокращенной пометкой: имя объекта приписывается наименьшему из концептов, в которых встречается данный объект, а имя атрибута – наибольшему из концептов, в которых присутствует этот атрибут).

2. Решетка на множестве ролей

Представим элементы ролевой политики разграничения доступа в терминах АФП. Заметим, что:

1) формальный контекст симметричен относительно множеств объектов и атрибутов;

2) в вершине решетки концептов расположен концепт с максимальным объемом (минимальное содержание).

Исходя из этого, начальные данные о ролевой модели представим формальным контекстом $PU = (P, U, \mathcal{R})$, в котором множество объектов – это множество полномочий P, множество атрибутов – это множество пользователей U, бинарное отношение $\mathcal{R} \subseteq P \times U$ определяется правилом:

$$\forall\, p \in P, \forall\, u \in U: p\, \mathcal{R}\, u \Leftrightarrow p \in UP(u).$$

Матрица **PU** контекста будет иметь размерность $m \times n$. Единичное значение элемента матрицы **PU** соответствует наличию полномочия у пользователя, а нулевое – отсутствию полномочия: если пользователь u_j авторизован на полномочие p_i ($p_i \in UP(u_j)$), то **PU** $[i, j] = 1$, в противном случае **PU** $[i, j] = 0$. На практике при определении формального контекста вместо бинарного отношения $\mathcal{R}$ сразу будем указывать матрицу **PU**: $PU = (P, U, \mathbf{PU})$.

Будем искать концепты для данной системы и отождествлять их с ролями. Каждая роль (концепт) $r_k = (X(r_k), Y(r_k))$ характеризуется двумя множествами:

1) объем $X(r_k) \subseteq P$ – набор полномочий, сопоставленных роли;

2) содержание $Y(r_k) \subseteq U$ – пользователи, которые авторизованы на данную роль.

Пусть $R = \{r_1, \ldots, r_s\}$ – множество всех ролей (концептов). Очевидно, что:

1) объем концепта определяет отображение RP: $RP(r_k) = X(r_k)$;

2) содержание концепта определяет отображение UR: $UR(u_j) = \{r_k \mid u_j \in Y(r_k)\}$;

3) решетка концептов определяет иерархию ролей – частично упорядоченное множество *R* и отображение *RR*. Матрица отображения *RR* представляет собой транзитивное замыкание матрицы решетки концептов.

Выполнение свойств 1) – 3) отображения RR следует из того факта, что отношение предшествования «$\prec$» на множестве концептов является отношением нестрогого порядка.

Утверждение 1. Иерархия ролей, порожденная решеткой концептов, гарантирует выполнение условия наследования полномочий.

Доказательство: $rj \prec ri \Rightarrow (X(rj), Y(rj)) \prec (X(ri), Y(ri)) \Rightarrow X(rj) \subseteq X(ri) \Rightarrow RP(rj) \subseteq RP(ri)$. Что и требовалось доказать.

Утверждение 2. Иерархия ролей, порожденная решеткой концептов, гарантирует выполнение условия наследования авторизации.

Доказательство: $rj \prec ri \Rightarrow (X(rj), Y(rj)) \prec (X(ri), Y(ri)) \Rightarrow Y(ri) \subseteq Y(rj)$. Пусть $ri \in UR(u)$. Тогда по построению $u \in Y(ri)$. Так как $Y(ri) \subseteq Y(rj)$, то $u \in Y(rj) \Rightarrow rj \in UR(u)$. Что и требовалось доказать.

Таким образом, справедлива следующая теорема.

Теорема 1. Используя АФП, по заданному отображению $UP: U \rightarrow 2P$ возможно построить отображения $RP: R \rightarrow 2P$, $UR: U \rightarrow 2R$, $RR: R \rightarrow 2R$ ролевой модели разграничения *доступа.*

Заметим, что чем выше в построенной иерархии находится роль-концепт, тем больше набор ее полномочий (больше объем) и меньше подмножество пользователей, авторизованных на эту роль (меньше содержание). Может случиться, что содержание вершины иерархии пусто. Этот концепт следует интерпретировать как «псевдороль суперадминистратора», на которую не авторизован ни один пользователь. На практике введение подобных «фиктивных» узлов используется с целью получения заданной структуры иерархии объектов (например, древовидной), что позволяет решать различные проблемы с единой точки зрения [2, 21].

Аналогично самая младшая роль в иерархии может иметь пустой объем. Авторизация на такую роль не имеет смысла, так как пользователи не получат никаких полномочий. Поэтому в дальнейшем концепт с пустым объемом в ролевой иерархии рассматриваться не будет.

Вообще говоря, существование концепта с пустым объемом необходимо для построения решетки. В свою очередь, решетка концептов порождает частично упорядоченное множество ролей, для любых двух элементов которого существуют точная верхняя и точная нижняя грани. Для ролевой иерархии это требование является избыточным. Достаточно задать частичный порядок на множестве ролей. Существует практика построения не всей решетки контекстов, а ее части. Верхняя часть решетки контекстов, включающая только крупные концепты, на-

зывается айсбергом [22] (чем ближе концепт к вершине решетки, тем более крупным он является, чем дальше – тем более мелким).

Применительно к ролевой иерархии разумно исключить из множества ролей наиболее мелкие концепты. Для возможности построения отображения UR требуется выполнить условие покрытия множества пользователей: объединение содержаний концептов айсберга должно покрывать множество пользователей (атрибутов). Заметим, что при построении айсберга будет исключен и концепт с нулевым объемом (максимальным содержанием), о котором говорилось ранее.

Теорема 2. Используя АФП, по заданному отображению UP: U → 2P можно построить семейство ролевых иерархий $\{(R,\prec),(R1,\prec),\ldots,(Rs,\prec)\}$, где $(R,\prec)$ – решетка; $(R1,\prec),\ldots,(Rs,\prec)$ – айсберги решетки $(R,\prec)$.

Усечение решетки ролей до айсберга можно интерпретировать как оптимизацию ролевого графа. При этом критерием оптимизация является мощность множества ролей: |R| → min, а ограничением – условие покрытия множества пользователей. Несложно понять, что такой айсберг можно получить по линейной диаграмме решетки концептов, включив в него все узлы с метками-атрибутами.

Пример 1. Пусть в информационной системе определены множества: пользователи U = {u1, u2, u3, u4, u5} и полномочия P = {p1, p2, p3, p4}. Отображение UP: U → 2P задано матрицей UP (рис. 1а). Рассмотрим формальный контекст PU = (P, U, PU), матрица PU которого определяется по правилу: PU = UPT (рис. 1б). Иерархия ролей, порожденная решеткой концептов контекста PU, представлена на рис. 1в. Роли имеют следующие объемы и содержания:

r1 = ({p1, p2, p3, p4}, {u1}); r2 = ({p1, p2, p3}, {u1, u2});
r3 = ({p2, p3}, {u1, u2, u5}); r4 = ({p2, p4}, {u1, u2, u4});
r5 = ({p3, p4}, {u1, u2, u3}); r6 = ({p2}, {u1, u2, u4, u5});
r7 = ({p3}, {u1, u2, u3, u5}); r8 = ({p4}, {u1, u2, u3, u4}).

Из решетки ролей можно выделить айсберг, определяемый первыми пятью ролями. Действительно, $\bigcup_{i=1}^{5} Y(r_i) = U$ (рис. 1в) концепты, не вошедшие в айсберг, представлены незакрашенными узлами).

а)

	p_1	p_2	p_3	p_4
u_1	1	1	1	1
u_2	0	1	1	1
u_3	0	0	1	1
u_4	0	1	0	1
u_5	0	1	1	0

= **UP**

б)

		Пользователи (атрибуты)				
		u_1	u_2	u_3	u_4	u_5
Полномочия (объекты)	p_1	1	0	0	0	0
	p_2	1	1	0	1	1
	p_3	1	1	1	0	1
	p_4	1	1	1	1	0

в)

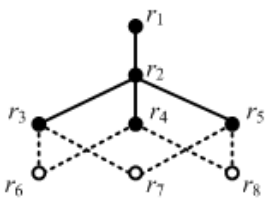

Рис. 1. Процесс построения ролевого графа: а) отображение *UP*; б) формальный контекст; в) иерархия ролей (ориентация дуг: «сверху – вниз»).

Таким образом, основные отображения RP: $R \to 2^P$, UR: $U \to 2^R$ и RR: $R \to 2^R$ будут описаны матрицами **RP**, **UR** и **RR** соответственно:

$$\begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \end{array}\!\!\begin{array}{c} \begin{array}{cccc} p_1 & p_2 & p_3 & p_4 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \end{array} = \mathbf{RP}, \quad \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{array}\!\!\begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & r_3 & r_4 & r_5 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array} = \mathbf{UR}, \quad \begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \end{array}\!\!\begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & r_3 & r_4 & r_5 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} = \mathbf{RR}.$$

3. Алгоритм построения ролевой модели разграничения доступа

Резюмируя вышеизложенные рассуждения, сформулируем основные шаги алгоритма построения основных отображений ролевой модели разграничения доступа использующего АФП.

Вход. Булева матрица **UP** отображения UP: $U \to 2^P$.

Выход. Булевы матрицы **RP**, **UR** и **RR** отображений RP: $R \to 2^P$, UR: $U \to 2^R$ и RR: $R \to 2^R$ соответственно.

Шаг 1. Построить формальный контекст $PU = (P, U, \mathbf{PU})$, в котором $\mathbf{PU} = \mathbf{UP}^T$.

Шаг 2. Найти все концепты $r_k = (X(r_k), Y(r_k))$ контекста PU (алгоритмы построения множества концептов можно найти в [15 – 17]). $X(r_k)$ – объем концепта r_k, $Y(r_k)$ – его содержание. Пусть s – число полученных концептов. Сформировать множество ролей $R = \{r_1, \ldots, r_s\}$.

Шаг 3. На множестве ролей R задать отношение порядка «$\prec$» по правилу: $r_j \prec r_i \Leftrightarrow X(r_j) \subseteq X(r_i)$ ($Y(r_j) \supseteq Y(r_i)$). Построить решетку ролей $(R, \prec)$ (алгоритмы построения решетки концептов представлены, например, в работах [15 – 17]). Диаграмму Хассе этой решетки интерпретировать как ролевой граф. Пусть **R** – матрица смежности ролевого графа.

Шаг 4. При необходимости провести усечение решетки концептов до айсберга с множеством ролей R_I: $\bigcup\limits_{r_i \in R_I} Y(r_i) = U$. Для получения айсберга с минимальным числом ролей необходимо:

1) для каждого пользователя найти наибольший концепт из тех, в содержании которых присутствует этот пользователь;

2) из отобранных концептов и вершины решетки сформировать айсберг.

Далее будем считать, что R и **R** – это множество узлов и матрица смежности результирующего ролевого графа.

Шаг 5. Построить матрицу **RP** по правилу: $\mathbf{RP}[i, j] = 1 \Leftrightarrow p_j \in X(r_i)$.

Шаг 6. Построить матрицу **UR** по правилу: $\mathbf{UR}[i, j] = 1 \Leftrightarrow u_i \in Y(r_j)$.

Шаг 7. Построить матрицу **RR** как транзитивное замыкание матрицы **R** (для построения транзитивного замыкания матрицы используется алгоритм Уоршалла [23]).

4. Оптимизация ролевого графа

Предложенный алгоритм построения ролевой модели разграничения доступа может быть использован для оптимизации уже имеющейся в системе ролевой иерархии. Продемонстрируем этот подход на примере.

Пример 2. Будем считать, что в системе заданы правила авторизации пользователей на роли и ролей друг на друга (матрицы **UR** и **RR**), кроме того, в системе определены полномочия для ролей (матрица **RP**):

$$\begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \end{array}\!\!\begin{array}{c} \begin{array}{cccc} p_1 & p_2 & p_3 & p_4 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array} = \mathbf{RP}, \quad \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \end{array}\!\!\begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & r_3 & r_4 & r_5 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} = \mathbf{UR}, \quad \begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \end{array}\!\!\begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & r_3 & r_4 & r_5 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} = \mathbf{RR}.$$

Этим данным соответствует ролевой граф, представленный на рис. 2а.

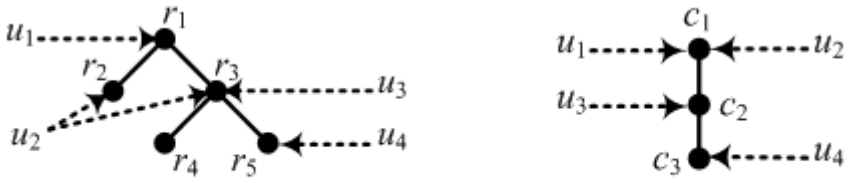

Рис. 2. Исходный (а) и оптимизированный (б) ролевые графы, где дуги ролевых графов изображены непрерывными линиями (ориентация дуг «сверху-вниз»), авторизация пользователей представлена пунктирными линиями.

Построим матрицу распределения полномочий между пользователями с учетом наследования при авторизации ролей – матрицу **UP** и рассмотрим формальный контекст $PU = (P, U, \mathbf{PU})$:

$$\begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \end{array}\!\!\begin{array}{c} \begin{array}{cccc} p_1 & p_2 & p_3 & p_4 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array} = \mathbf{UP}, \quad \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \\ p_4 \end{array}\!\!\begin{array}{c} \begin{array}{cccc} u_1 & u_2 & u_3 & u_4 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \end{array} = \mathbf{PU}.$$

Множество концептов формального контекста PU состоит из трех элементов:

$c_1 = (\{p_1, p_2, p_3, p_4\}, \{u_1, u_2\})$;

$c_2 = (\{p_3, p_4\}, \{u_1, u_2, u_3\})$;

$c_3 = (\{p_4\}, \{u_1, u_2, u_3, u_4\})$.

Диаграмма Хассе решетки, образованной этими концептами, интерпретируется как оптимизированный ролевой граф (рис. 2б). Результирующие отображения распределения полномочий и авторизации представляются матрицами $\mathbf{RP}^*$, $\mathbf{UR}^*$ и $\mathbf{RR}^*$:

$$\begin{array}{c} \\ c_1 \\ c_2 \\ c_3 \end{array}\!\!\begin{array}{c} \begin{array}{cccc} p_1 & p_2 & p_3 & p_4 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array} = \mathbf{RP}^*, \quad \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \end{array} = \mathbf{UR}^*, \quad \begin{array}{c} \\ c_1 \\ c_2 \\ c_3 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \end{array} = \mathbf{RR}^*.$$

5. Анализ результатов и выводы

В рамках предложенного в данной статье алгоритма применение АФП позволяет по заданному распределению полномочий между пользователями получить в общем случае несколько ролевых иерархий, одна из которых представляет собой решетку ролей, а остальные являются айсбергами этой решетки. Отображение *UP*, лежащее в основе формального контекста, можно расценивать как «набор типовых пользователей», по которому строится ролевой граф. Чем больше вероятность того, что новый пользователь, определяемый в системе, по запросу на набор полномочий совпадет с каким-либо «типовым» пользователем, тем эффективнее построенная модель ролевой иерархии.

Необходимо отметить, что существуют два различных подхода в построении ролевой политики безопасности: «сверху-вниз» и «снизу-вверх» [24, 25]. Подход «сверху-вниз» реализуется путем анализа бизнес-процессов и определения потенциальных ролей, необходимых для реализации некоторых функций в системе. Этот подход подразумевает разработку системы разграничения доступа на этапе проектирования системы до ввода его в эксплуатацию. Теоретически он позволяет получить оптимальное решение на начальном этапе функционирования системы, однако вызывает значительные трудности с течением времени, так как требует больших затрат на адаптацию к изменениям в компьютерной системе. Один из возможных подходов к построению ролевой политики безопасности "сверху-вниз" представлен в работе [26]. Подход "снизу-вверх" основывается на данных о распределении полномочий между пользователями и на их основе строит множество ролей, удовлетворяющих определенным ограничениям. При этом игнорируются существующие бизнес-процессы, в результате чего полученные роли могут не совпадать с определенными функциями работников организации [27]. Подход «снизу-вверх» получил название "разработка ролей" (Role Mining, RM). Наибольшими перспективами "разработка ролей" обладает для больших информационных систем, для которых анализ бизнес-процессов составляет зна-

чительную трудность,в силу распределенности в пространстве и большого количества пользователей.

В статье [27] предложена формальная формулировка проблемы разработки ролей (Role Mining Problem, RMP): при заданных наборе пользователей, наборе полномочий и распределении полномочий между пользователями необходимо найти минимальный набор ролей, назначение пользователей на роли и распределение полномочий между ролями. Эта задача является NP-трудной. Для решения проблемы разработки ролей предложено несколько алгоритмов, из которых прежде всего, необходимо отметить FastMiner [28] и ORCA [29], служащих основой для целого ряда других алгоритмов [30]. Одним из важных расширений данной задачи является проблема Edge RMP, в которой минимизируется сумма количества авторизаций пользователей на роли пользователя и количество ролей [31].

В статьях [32, 33] в качестве целевой использована взвешенная сумма количества ролей и количества авторизаций пользователей на роли. Также в этих работах предложен алгоритм CompleteMiner построения ролевой политики безопасности на основе АФП. Основная идея состоит в построении решетки формальных понятий с последующим ее усечением. Усечение представляет собой алгоритм постобработки решетки формальных понятий и применение некоторого набора правил. Однако предложенная процедура усечения не гарантирует оптимального решения проблемы разработки ролей. В работе [34] также исследуется применение АФП к задаче построения и сопровождения ролевой политики разграничения доступа. При этом атрибутами контекста являются полномочия, а объектами – декартово произведение множества ролей на множество данных. Цель построения решетки концептов заключается в моделировании наборов полномочий ролей. Эта идея нашла развитие в статье [35], где авторы работают уже с трехмерным контекстом, соответствующим ролевой модели.

В данной работе предложен алгоритм решения проблемы разработки ролей, основанный на АФП и не требующий дополнительной постобработки. Как показано в статье [2], задача построения ролевого дерева не является однозначной. Существуют эквивалентные преобразования, позволяющие изменять набор ролей и назначенных им полномочий, не изменяющие полномочия пользователя.

Таким образом, алгоритм построения ролевого дерева, предложенный в статье, является базой для поиска ролевой модели, удовлетворяющей той или иной оптимизационной задаче.

ЛИТЕРАТУРА

1. *Белим С.В., Богаченко Н.Ф.* Построение ролевой политики безопасности на произвольном ориентированном графе // Проблемы информационной безопасности. Компьютерные системы. – 2009. – № 3. – С. 7–17.

2. *Belim S., Bogachenko N., Ilushechkin E.* An analysis of graphs that represent a role-based security policy hierarchy // Journal of Computer Security. – 2015. – Vol. 23, No. 5. – P. 641-657.
3. *Li N., Mao Z.* Administration in Role-Based Access Control // Proceedings of the 2nd ACM symposium on Information, computer and communications security. – 2007. – P. 127-138.
4. *Tahir M.N.* Hierarchies in Contextual Role-Based Access Control Model (C-RBAC) // International Journal of Computer Science and Security (IJCSS). – 2008. – No 2(4). – P. 28-42.
5. *Leitner M.* Delta Analysis of Role-Based Access Control Models // Lecture Notes in Computer Science. – 2013. – P. 507-514.
6. *Zhang D., Ramamohanarao K., Versteeg S., Zhang R.* Graph Based Strategies to Role Engineering // Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research. – 2010. – Article 25.
7. *Colantonio A., Pietro R.D., Ocello A., Verde N.V.* Taming role mining complexity in RBAC // Computers & Security. – 2010. – No 29(5). – P. 548-564.
8. *Wille R.* Restructuring Lattice Theory: an approach based on hierarchies of concept / Ordered sets (I. Rival, ed.). – Reidel, Dordrecht-Boston, 1982.
9. *Wille R.* Lattices in data analysis: how to draw them with computer / Algorithms and Order (I. Rival, ed.). – Kluwer, Dordrecht-Boston, 1989.
10. *Wille R.* Knowledge acquisition by methods of formal concept analysis / Data Analysis, learning symbolic and numeric knowledge (E. Diday, ed.). – New York, Budapest, Nova Science Publisher, 1989.
11. *Биркгоф Г.* Теория решеток. – М.: Наука, 1984.
12. *Белим С.В., Богаченко Н.Ф., Фридман И.А.* Исследование утечки привилегий в ролевой политике безопасности // Проблемы информационной безопасности. Компьютерные системы. – 2012. – № 3. – С. 7-13.
13. *Игнатов Д.И.* Анализ формальных понятий: от теории к практике / Анализ изображений, сетей и текстов // Доклады Всероссийской научной конференции АИСТ'12. Модели, алгоритмы и инструменты анализа данных; результаты и возможности для анализа изображений, сетей и текстов. Екатеринбург, 16 – 18 марта 2012 г. – Вып. 1. – М.: Национальный открытый университет «ИНТУИТ», 2012. – С. 3-15.
14. *Кашницкий Ю.С., Игнатов Д.И.* Ансамблевый метод машинного обучения, основанный на рекомендации классификаторов // Интеллектуальные системы. Теория и приложения. – 2015. – Т. 19, № 4. – С. 37-55.
15. *Choi V.* Faster Algorithms for Constructing a Concept (Galois) Lattice. – 2006. – arXiv:cs/0602069v2.
16. *Krajca P., Outrata J., Vychodil V.* Parallel Recursive Algorithm for FCA / Proceedings of the Sixth International Conference on Concept Lattices and Their Applications // CEUR Workshop Proceedings. – 2008. – Vol. 433. – P. 71-82.
17. *Kuznetsov S.O., Obiedkov S.A.* Comparing performance of algorithms for generating concept lattices // Journal of Experimental & Theoretical Artificial Intelligence. – 2002. – No 14(2–3). – P. 189-216.
18. *Евтушенко С.А.* Система анализа данных «Concept Explorer» // Труды 7-й Национальной конференции по искусственному интеллекту. – 2000. – С.127-134.
19. URL: http://conexp.sourceforge.net/.
20. URL: https://pypi.python.org/pypi/concepts.

21. *Belim S.V., Bogachenko N.F.* Distribution of Cryptographic Keys in Systems with a Hierarchy of Objects // Automatic Control and Computer Sciences. – 2016. – Vol. 50, No. 8. – P. 777-786.
22. *Stumme, G., Taouil, R., Bastide, Y., Lakhal, L.* Conceptual Clustering with Iceberg Concept Lattices // Proceedings of GI-Fachgruppentreffen Maschinelles Lernen. – University of Dortmund, 2001.
23. *Aho A.V., Hopcroft J.E., Ullman J.D.* Data Structures and Algorithms. – Addison-Wesley, 1987.
24. *Ene A., Horne W., Milosavljevic N., Rao P., Schreiber R., Tarjan R.E.* Fast exact and heuristic methods for role minimization problems // Proceedings of Symposium on Access Control Models and Technologies (SACMAT). – 2008. – P. 1-10.
25. *Kuhlmann M., Shohat D., Schimpf G.* Role mining – revealing business roles for security administration using data mining technology // Symposium on Access Control Models and Technologies (SACMAT). – 2003.
26. *Schaad A., Moffett J., Jacob J.* The role-based access control system of a European bank: A case study and discussion // Proceedings of ACM Symposium on Access Control Models and Technologies. – 2001. – P. 3-9.
27. *Vaidya J., Atluri V., Guo Q.* The role mining problem: Finding a minimal descriptive set of roles // Proceedings of Symposium on Access Control Models and Technologies (SACMAT). – 2007. – P. 175-184.
28. *Vaidya J., Atluri V., Warner J.* RoleMiner: mining roles using subset enumeration // Proceedings of the ACM conference on Computer and Communications security. – 2006. – P. 144-153.
29. *Schlegelmilch J., Steffens U.* Role mining with ORCA // Symposium on Access Control Models and Technologies (SACMAT). – 2005.
30. *Molloy I., Li N., Li T., Mao Z., Wang Q., Lobo J.* Evaluating Role Mining Algorithms // Proceedings of ACM Symposium on Access Control Models and Technologies (SACMAT). – 2009.
31. *Lu H., Vaidya J., Atluri V.* Optimal boolean matrix decomposition: Application to role engineering // Proceedings of International Conference on Data Engineering (ICDE). – 2008. – P. 297-306.
32. *Li N., Li T., Molloy I., Wang Q., Bertino E., Calo S., Lobo J.* Role Mining for Engineering and Optimizing Role Based Access Control Systems. – Purdue University, IBM T.J. Watson Research Center, 2007.
33. *Molloy I., Chen H., Li T., Wang Q., Li N., Bertino E., Calo S., Lobo J.* Mining roles with semantic meanings // Proceedings of the 13th ACM symposium on Access control models and technologies. – 2008. – P. 21-30.
34. *Kumar C.A.* Modeling Access Permissions in Role Based Access Control Using Formal Concept Analysis / Venugopal K.R., Patnaik L.M. (eds) Wireless Networks and Computational Intelligence // Communications in Computer and Information Science. – 2012. – Vol. 292. – Springer, Berlin, Heidelberg. – P. 578-583.
35. Kumar C.A., Mouliswaran S.C., Jin-hai L., Chandrasekar C. Role based access control design using Triadic concept analysis // Journal of Central South University. – 2016. – Vol. 23, No. 12. – P. 3183-3191.

Статья представлена к публикации членом редколлегии А.А. Шелупановым.

E-mail:
Белим Сергей Викторович – sbelim@mail.ru;
Богаченко Надежда Федоровна – nfbogachenko@mail.ru.