Математические структуры и моделирование 2024. № 1 (69). С. 94–102

УДК 004.056
DOI 10.24147/2222-8772.2024.1.94-102

ИНЖИНИРИНГ ПРИВИЛЕГИЙ В ЗАДАЧЕ ПОСТРОЕНИЯ РОЛЕВОЙ ПОЛИТИКИ РАЗГРАНИЧЕНИЯ ДОСТУПА

Н.Ф. Богаченко
к.ф.-м.н., доцент, e-mail: nfbogachenko@mail.ru

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

Аннотация. Проблема инжиниринга ролей расширяется подзадачей разработки привилегий. Предполагается, что в информационной системе задано дискреционное разграничение доступа. Для построения ролевой политики безопасности предлагается методика, основанная на алгоритмах анализа формальных понятий. По матрице доступов строится решётка Галуа, узлы которой интерпретируются как возможные привилегии. Определяются критерии выбора оптимального набора привилегий и обсуждается эвристический алгоритм решения поставленной задачи.

Ключевые слова: разграничение доступа, роли, привилегии, матрица доступов, анализ формальных понятий.

Одним из основных способов обеспечения информационной безопасности является защита от несанкционированного доступа. При этом задача построения политики разграничения доступа считается не менее важной, чем, например, стойкость используемых криптографических алгоритмов. Практическая реализация классических моделей управления доступом сталкивается с рядом трудностей применительно к крупномасштабным информационным системам (Large-Scale Complex IT Systems). В данной работе будет рассмотрена ролевая модель разграничения доступа и проблемы, возникающие при её построении.

1. Разработка ролей

Рассмотрим математическую постановку задачи построения ролевого разграничения доступа (Role-Based Access Control) [1] в некоторой информационной системе. Этот процесс состоит из трёх этапов:

1. Построение иерархии ролей:
   - ✓ формирование множества ролей: $R = \{r_1, \dots, r_n\}$;
   - ✓ формирование множества привилегий: $P = \{p_1, \dots, p_m\}$;.
   - ✓ назначение привилегий ролям: $RP : R \to 2^P$;
   - ✓ авторизация ролей: $RR : R \to 2^R$.
2. Предоставление ролей пользователям:
   - ✓ формирование множества пользователей: $U = \{u_1, \dots, u_s\}$;
   - ✓ назначение привилегий пользователям: $UP : U \to 2^P$;
   - ✓ авторизация пользователей: $UR : U \to 2^R$.

3. Работа в системе:
   - ✓ формирование множества сеансов работы: $C = \{c_1, \ldots, c_t\}$;
   - ✓ управление доступом: $CU : C \times U \to 2^R$.

При реализации ролевой политики разграничения доступа в небольшой информационной системе, как правило, не возникает сложности в выборе множеств R и P и отображения RP. Другими словами, определение ролей и закрепление за каждой ролью набора привилегий осуществляется администратором безопасности в ручном режиме на основе анализа бизнес-процессов. Ситуация резко усложняется, когда информационная система является крупномасштабной [2, 3]. В этом случае поиск оптимального (корректного, полного) множества ролей и связанных с каждой ролью привилегий является весьма сложной задачей, требующей вычислительной поддержки (автоматизации). Данная задача получила название «инжиниринг ролей» (Role Engineering) [4]. Как известно, для решения сложных проблем, в том числе и для задачи инжиниринга ролей, возможны два подхода, два пути решения: нисходящий (top-down) и восходящий (bottom-up).

Применительно к задаче инжиниринга ролей нисходящий подход (или, как его ещё называют, метод «с чистого листа») заключается в классической последовательности поэтапного построения ролевой политики разграничения доступа. Группа экспертов определяет множество R, задаёт отображения RP и RR. И лишь затем происходит переход ко второму этапу – к авторизации пользователей на роли. Но в крупномасштабных информационных системах как число пользователей, так и число объектов доступа (а значит, и число привилегий) существенно увеличивается. Кроме того, нередко необходим учёт уже имеющихся информационных потоков, разрешений и запретов на доступ и т. п. В связи с этим всё более востребован восходящий подход к решению задачи инжиниринга ролей. В этом случае построение ролевой модели начинается со второго этапа: за основу берутся множество U и отображение UP, т. е. уже существующие правила доступа – потребности пользователей в определённом наборе привилегий. При таком подходе к проектированию ролевого разграничения доступа часто используются методы интеллектуального анализа данных (Data Mining). В связи с чем задачу инжиниринга ролей, для решения которой используется восходящий подход, принято называть «проблемой разработки ролей» (Role Mining Problem) [5]. Постановка этой задачи может быть формализована следующим образом [6]:

- Дано:
  - ✓ множество пользователей: $U = \{u_1, \ldots, u_s\}$;
  - ✓ множество привилегий: $P = \{p_1, \ldots, p_m\}$;
  - ✓ отображение, определяющее назначение привилегий пользователям: $UP : U \to 2^P$.
- Найти:
  - ✓ множество ролей: $R = \{r_1, \ldots, r_n\}$;
  - ✓ отображение, определяющее назначение привилегий ролям: $RP : R \to 2^P$;
  - ✓ отображение, определяющее авторизацию пользователей: $UR : U \to 2^R$;
  - ✓ отображение, определяющее авторизацию ролей: $RR : R \to 2^R$.

Следует заметить, что в рассмотренной, уже ставшей классической, постановке проблемы разработки ролей в числе исходных данных определены привилегии, необходимые каждому пользователю для работы в системе, т. е. весь набор привилегий P и их желаемое распределение между пользователями UP известны заранее.

Возникает вопрос, что делать, если в крупномасштабной информационной системе множество привилегий P и тем более отображение UP ещё не определены? Если подсистема разграничения доступа строится «с нуля», то здесь необходима серьёзная работа экспертов. Но в случае, когда в информационной системе ранее уже была реализована дискреционная политика разграничения доступа (Discretionary Access Control) [7], являющаяся базовой для всех защищённых компьютерных систем, представляется возможным частично автоматизировать процесс выявления множества привилегий P за счёт применения методов интеллектуального анализа данных (используемых для решения проблемы разработки ролей в целом).

2. Вспомогательные модели и методы

Дискреционное разграничение доступа основано на произвольном управлении доступом: разрешение на доступ определяется для каждого субъекта к каждому объекту. Пусть S – множество субъектов, O – множество объектов, A – множество видов доступа. Для каждой пары $(s_i, o_j) \in S \times O$ задаётся набор разрешённых видов доступа $\alpha_{ij} \subseteq A$, т. е. определяется правило доступа (s_i, o_j, α_{ij}). Такие правила организовываются в матрицу доступов $\mathbf{M}$ размерности $|S| \times |O|$, в которой каждому субъекту соответствует своя строка, каждому объекту – свой столбец, а на их пересечении указывается набор разрешённых видов доступа α_{ij}.

Не ограничивая общности, будем считать, что $U = S$, т. е. множества пользователей и субъектов совпадают. В реальных системах $U \subseteq S$, но для решения наших задач достаточно преобразовать матрицу доступов $\mathbf{M}$, удалив из неё строки, которые не соответствуют пользователям.

Далее рассмотрим один из подходов к решению вопроса формирования множества привилегий P на основе имеющейся матрицы доступов $\mathbf{M}$. Предлагается использовать методы анализа формальных понятий (Formal Concept Analysis) [8], успешно применяемые для решения классической проблемы разработки ролей [6].

В основе анализа формальных понятий лежит определение формального контекста – тройки $(I, J, \mathbf{R})$, где I – множество предметов; J – множество признаков; $\mathbf{R}$ – бинарная матрица размерности $|I| \times |J|$, сопоставляющая признаки предметам. На любые подмножества $X \subseteq I$, $Y \subseteq J$ можно подействовать операторами Галуа:

$$X^{\uparrow} = \{j \in J | \forall i \in X : [\mathbf{R}]_{ij} = 1\},$$

$$Y^{\downarrow} = \{i \in I | \forall j \in Y : [\mathbf{R}]_{ij} = 1\}.$$

Несложно заметить, что $X^{\uparrow}$ – это множество признаков, общих для предметов из X, $Y^{\downarrow}$ – множество предметов, гарантированно имеющих признаки из Y. Формальное понятие формального контекста – это пара (X, Y) такая, что

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

В формальном понятии (X, Y) подмножество X принято называть объёмом, а подмножество Y – содержанием. Очевидно, что каждое формальное понятие в матрице **R** определяет заполненный единицами прямоугольник с точностью до перестановки строк и/или столбцов. Для любых двух формальных понятий (X_1, Y_1) и (X_2, Y_2) выполняется правило включения:

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

Это позволяет на множестве всех формальных понятий заданного формального контекста ввести отношение частичного порядка

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

и доказать, что формальные понятия, упорядоченные согласно этому отношению, образуют математическую решётку. Эта решётка называется решёткой Галуа и визуализируется в виде диаграммы Хассе (ориентированного графа), узлами которой являются формальные понятия. Чем выше в диаграмме находится формальное понятие (X, Y), тем больше его объём X и меньше содержание Y.

3. Разработка привилегий

Исходя из подзадач, возникающих на практике при построении ролевой политики разграничения доступа в крупномасштабных информационных системах, очевидна необходимость в формализации и поиске способов решения проблемы выявления полного множества привилегий и закрепления этих привилегий за пользователями – «проблемы инжиниринга привилегий»:

- Дано: множество пользователей U.
- Найти:
  - ✓ множество привилегий P;
  - ✓ отображение, определяющее назначение привилегий пользователям, $UP : U \to 2^P$.

Если для решения поставленной задачи применяются восходящие принципы проектирования, т. е. за основу решения берутся уже существующие в системе предпосылки к разграничению доступа и используются методы интеллектуального анализа данных, то задачу инжиниринга привилегий назовём «проблемой разработки привилегий» [9, 10]. Одна из возможных постановок проблемы разработки привилегий приведена далее.

- Дано:
  - ✓ множество пользователей U;
  - ✓ множество объектов, к которым осуществляется доступ, O;
  - ✓ множество видов доступа A;
  - ✓ матрица доступов **M** размерности $|U| \times |O|$.
- Найти:
  - ✓ множество привилегий P;
  - ✓ отображение, определяющее назначение привилегий пользователям, $UP : U \to 2^P$.

Таким образом в процессе формирования (или выявления) элементов множества привилегий P за основу берутся разрешённые пользователям виды доступа к объектам системы. Важно напомнить, что привилегии в ролевой политике разграничения доступа выдаются пользователям на действия в информационной системе в целом, применительно к целой группе или классу объектов. Виды доступа в дискреционной политике, напротив, задают разрешения на доступ к определённому объекту. Поэтому можно сформулировать следующие требования к процедуре определения привилегий:

1. Привилегия должна быть соотнесена с группой объектов, к которым задан один и тот же вид доступа.
2. Привилегия должна быть выдана группе пользователей, которые имеют один и тот же вид доступа к некоторому объекту.

Эти правила приводят к идее использования методов анализа формальных понятий, которые позволят по матричному представлению исходных данных выделить привилегии с оптимальными наборами объектов и пользователей. Предлагается следующая методика решения проблемы разработки привилегий, состоящая из трёх этапов.

Этап 1. В соответствии с видами доступа, определёнными множеством A, построим бинарные матрицы $\mathbf{M}_1, \mathbf{M}_2, \ldots, \mathbf{M}_{|A|}$ по правилу:

$$[\mathbf{M}_k]_{ij} = 1 \Longleftrightarrow a_k \in [\mathbf{M}]_{ij},$$

где $k = 1, \ldots, |A|$, $a_k \in A$, $\mathbf{M}$ – матрица доступов. Данный процесс назовём декомпозицией матрицы $\mathbf{M}$ по видам доступа. Далее матрицы, полученные в результате декомпозиции, будем рассматривать отдельно.

Этап 2. Следующая подзадача – найти оптимальный набор привилегий P_k для вида доступа a_k. Под оптимальным будем понимать такой набор, который в первую очередь минимизирует число выделенных привилегий. Но если руководствоваться только этим критерием, то достаточно будет создать одну привилегию, которая выдаст доступ a_k ко всем объектам множества O. К сожалению, при таком подходе подавляющее большинство пользователей получат доступ к объектам, к которым в матрице $\mathbf{M}_k$ доступ был запрещён. С точки зрения минимизации рисков утечки информации естественно потребовать, чтобы число таких «лишних» объектов стремилось к нулю. А это приведёт к обратному эффекту – число привилегий придётся увеличивать. Наконец, выделенные привилегии должны обеспечить все доступы, которые разрешались матрицей $\mathbf{M}_k$.

Представим нашу проблему в терминах анализа формальных понятий. Для каждой бинарной матрицы $\mathbf{M}_k$ определим формальный контекст $M_k = (U, O, \mathbf{M}_k)$, в котором множество предметов – это множество пользователей U, множество признаков – это множество объектов O, бинарная матрица, сопоставляющая предметы и признаки, – это матрица $\mathbf{M}_k$.

Далее найдём все формальные понятия (X_k^i, Y_k^i) формального контекста M_k и построим решётку Галуа Γ_k. Существуют различные алгоритмы решения этой задачи. Их классификацию и описание можно найти, например, в работе [11]. Узлы полученной решётки (формальные понятия (X_k^i, Y_k^i)) будем интерпретировать как возможные привилегии, выделенные для вида доступа a_k. Среди узлов решётки Γ_k

необходимо выбрать набор привилегий $P_k = \{(X_k^1, Y_k^1), \ldots, (X_k^{m_k}, Y_k^{m_k})\}$, удовлетворяющий следующим критериям и ограничениям:

1. Число выделенных привилегий минимально:

$$F = |P_k| \longrightarrow \min. \tag{1}$$

2. Число «лишних» объектов, образующихся при переходе от матрицы доступов к распределению привилегий между пользователями, минимально:

$$G = \sum_{u_i \in U} \mathfrak{L}(\mathbf{M}, P_k, UP) \longrightarrow \min, \tag{2}$$

здесь $\mathfrak{L}$ – оператор, вычисляющий количество «лишних» объектов[1].

3. Объединение содержаний выделенных привилегий покрывает множество объектов:

$$\bigcup_{(X_k^i, Y_k^i) \in P_k} Y_k^i = O. \tag{3}$$

Очевидно, что поиск точного решения сформулированной подзадачи выбора оптимального набора привилегий (1) – (3) весьма трудоёмкий. Он сводится к полному перебору возможных подмножеств привилегий, удовлетворяющих требованиям матрицы доступов $\mathbf{M}_k$ в разрезе разрешений на доступ (требованиям ограничения (3)). С практической точки зрения достаточно найти приближённое решение. Рассмотрим эвристический алгоритм выбора набора P_k по матрице $\mathbf{M}_k$ и по решётке Γ_k.

Несложно понять, что для каждой j-й строки матрицы $\mathbf{M}_k$, отвечающей пользователю u_j, найдётся узел (X_k^i, Y_k^i) в решётке Γ_k, в котором:

- объём $X_k^i = \{u_j\}$ состоит из одного пользователя;
- содержание Y_k^i включает те объекты, которым отвечают единичные клетки выбранной строки матрицы $\mathbf{M}_k$.

На первом шаге отбора привилегий необходимо найти минимальное число именно таких узлов решётки Γ_k, объёмы которых содержат по одному пользователю, а все содержания в объединении покрывают множество объектов O. Тем самым будут удовлетворены все разрешения на доступ, выдаваемые матрицей $\mathbf{M}_k$ (ограничение (3)). Но этот процесс эквивалентен поиску минимального числа строк матрицы $\mathbf{M}_k$, покрывающих единицами все её столбцы. Эта подзадача представляет собой интерпретацию хорошо известной задачи минимизации булевой формулы методом Квайна–мак-Класки в пункте работы с импликантной матрицей, алгоритмы решения которой изучены и реализованы [12]. От выбранных строк опять перейдём к узлам решётки Γ_k.

На втором шаге, с целью минимизации числа «лишних» объектов при выдаче привилегий пользователям, для каждого выбранного на предыдущем шаге узла (X_k^i, Y_k^i) рассмотрим узлы, из которых ведут в него дуги, т. е. рассмотрим «родителей» этого узла. Если среди узлов-«родителей» можно выбрать набор узлов, содержания которых совместно покрывают содержание узла (X_k^i, Y_k^i), то фиксируем

[1]Вопросы построения отображения UP и оператора $\mathfrak{L}$ по матрице доступов $\mathbf{M}$ и набору привилегий P_k требуют отдельного обсуждения и выходят за рамки данной статьи.

такой минимальный по числу узлов набор и привилегию (X_k^i, Y_k^i) заменяем на привилегии, соответствующие отобранным узлам. Этот процесс, назовём его расщеплением привилегий, ведёт к увеличению числа привилегий и в зависимости от структуры решётки Γ_k может итеративно продолжаться вплоть до её верхней грани. Критерием остановки следует выбрать условие достижения некоторого баланса между числом привилегий и числом «лишних» объектов при авторизации пользователей на привилегии.

Этап 3. Пусть для каждого вида доступа $a_k \in A$ определён набор привилегий P_k. Сформируем множество привилегий $P = \bigcup P_k$ $(k = 1, \ldots, |A|)$ и определим отображение $UP : U \to 2^P$ так, чтобы обеспечить все разрешённые матрицей $\mathbf{M}$ доступы[2].

Пример 1. Разберём второй этап решения проблемы разработки привилегий на примере. Пусть построен формальный контекст M_k с матрицей $\mathbf{M}_k$.

$$\mathbf{M}_k = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{array} \begin{array}{c} \begin{array}{ccccccc} o_1 & o_2 & o_3 & o_4 & o_5 & o_6 & o_7 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array}.$$

Иерархия возможных привилегий, порождённая решёткой формальных понятий Γ_k формального контекста M_k, представлена на рис. 1. Возможные привилегии имеют следующие объёмы и содержания:

$p_1 = (\{u_1, u_2, u_3, u_4, u_5\}, \varnothing)$; $p_2 = (\{u_1, u_2, u_3\}, \{o_1, o_6\})$;
$p_3 = (\{u_2, u_4\}, \{o_3, o_5\})$; $p_4 = (\{u_4, u_5\}, \{o_2\})$;
$p_5 = (\{u_2, u_3\}, \{o_1, o_4, o_6\})$; $p_6 = (\{u_2\}, \{o_1, o_3, o_4, o_5, o_6\})$;
$p_7 = (\{u_4\}, \{o_2, o_3, o_5\})$; $p_8 = (\{u_5\}, \{o_2, o_7\})$;
$p_9 = (\varnothing, \{o_1, o_2, o_3, o_4, o_5, o_6, o_7\})$.

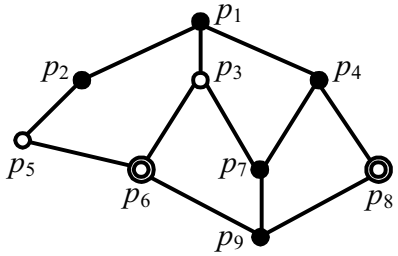

Рис. 1. Диаграмма Хассе решётки Галуа Γ_k

На первом шаге отбора привилегий будут выбраны привилегии p_6 и p_8, так как их объёмы состоят из одного пользователя, а объединение содержаний покрывает всё множество объектов. На втором шаге обнаруживается возможность заменить привилегию p_6 её «родителями» – привилегиями p_3 и p_5, так как содержания этих двух

[2]Более подробно алгоритм построения отображения UP будет рассмотрен в последующих публикациях.

привилегий в объединении покрывают содержание привилегии p_6. Таким образом, имеются два варианта формирования набора P_k: $P_k^1 = \{p_6, p_8\}$ и $P_k^2 = \{p_3, p_5, p_8\}$. Оценим число выделенных привилегий и число «лишних» объектов для каждого варианта (см. табл. 1, 2). Для P_k^1: $F = 2, G = 9$; для P_k^2: $F = 3, G = 2$. Окончательный выбор, какое из подмножеств P_k^1 или P_k^2 принять за искомый набор P_k, осуществляется экспертами, исходя из того, какой критерий – F или G – более значим.

Таблица 1. Отображение UP и «лишние» объекты для набора привилегий P_k^1

Пользователь	Привилегии	«Лишние» объекты
u_1	p_6	$\{o_3, o_4, o_5\}$
u_2	p_6	$\oslash$
u_3	p_6	$\{o_3, o_5\}$
u_4	$p_6 \cup p_8$	$\{o_1, o_4, o_6, o_7\}$
u_5	p_8	$\oslash$

Таблица 2. Отображение UP и «лишние» объекты для набора привилегий P_k^2

Пользователь	Привилегии	«Лишние» объекты
u_1	p_5	$\{o_4\}$
u_2	$p_3 \cup p_5$	$\oslash$
u_3	p_5	$\oslash$
u_4	$p_3 \cup p_8$	$\{o_7\}$
u_5	p_8	$\oslash$

Заключение

Результаты, представленные в статье, носят теоретический характер и затрагивают математическую постановку обсуждаемых проблем, алгоритмическое описание возможных путей решения. Практическая реализация предлагаемых алгоритмов и анализ результатов вычислительного эксперимента – это задачи дальнейших исследований.

Литература

1. Sandhu R.S., Coyne E.J., Feinstein H.L., Youman C.E. Role-Based Access Control Models // IEEE Computer. 1996. No. 29(2). P. 38–47.
2. Богаченко Н.Ф. Анализ проблем управления разграничением доступа в крупномасштабных информационных системах // Математические структуры и моделирование. 2018. № 2 (46). С. 135–152.

3. Богаченко Н.Ф. О сложности подсистем разграничения доступа крупномасштабных информационных систем // Математические структуры и моделирование. 2018. № 4 (48). С. 92–98.
4. Coyne E.J. Role engineering // RBAC '95: Proceedings of the first ACM Workshop on Role-based access control. New York: ACM Press, 1995. P. 4–5.
5. Kuhlmann M., Shohat D., Schimpf G. Role mining – revealing business roles for security administration using data mining technology // SACMAT '03: Proceedings of the eighth ACM symposium on Access control models and technologies. 2003. P. 179–186.
6. Белим С.В., Богаченко Н.Ф. Использование решётки формальных понятий для построения ролевой политики разграничения доступа // Информатика и системы управления. 2018. № 1 (55). С. 16–28.
7. Harrison M.A., Ruzzo W.L., Ullman J.D. On Protection in Operating Systems // Communications of the ACM. 1975. P. 14–25.
8. Wille R. Restructuring Lattice Theory: an approach based on hierarchies of concept // Ordered sets / Ed. I. Rival. Dordrecht; Boston: Reidel, 1982.
9. Богаченко Н.Ф. Интеллектуальный анализ политик разграничения доступа больших информационных систем // Математическое и компьютерное моделирование: сб. материалов V Междунар. науч. конф. Омск: Изд-во Ом. гос. ун-та, 2017. С. 142–145.
10. Богаченко Н.Ф. Проблема разработки полномочий // Омские научные чтения – 2020: материалы Четвёртой Всерос. нау. конф. Омск: Изд-во Ом. гос. ун-та, 2020. С. 367–369.
11. Семенова Д.В., Катаева А.В., Монгуш Ч.М. Метод декомпозиции формального контекста и неизбыточное представление закономерностей в многомерных данных: моногр. Красноярск: Сиб. федер. ун-т, 2020.
12. Дмитриев Г.А., Комиссарчик В.Ф., Марголис Б.И. Программа минимизации функций алгебры логики методом Мак-Класки // Программные продукты и системы. 1997. № 2. URL: `https://swsys.ru/index.php?page=article&id=1032` (дата обращения: 21.02.2024).

PERMISSIONS ENGINEERING IN THE TASK OF CONSTRUCTING A ROLE-BASED ACCESS CONTROL POLICY

N.F. Bogachenko
Ph.D. (Phys.-Math.), Associate Professor, e-mail: nfbogachenko@mail.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. The problem of role engineering is extended by the subproblem of permissions engineering. It is assumed that discretionary access control is specified in the information system. To build a role-based security policy, a technique based on algorithms for formal concept analysis is proposed. Based on the access matrix, a Galois lattice is constructed, the nodes of which are interpreted as possible permissions. Criteria for choosing the optimal set of permissions are determined and a heuristic algorithm for solving the problem is discussed.

Keywords: access control, roles, permissions, access matrix, formal concept analysis.

Дата поступления в редакцию: 26.02.2024