

УДК 004.415
ББК 32.973.202-018.2

УПРАВЛЕНИЕ ЛОГИЧЕСКИМИ ВЫВОДАМИ В СЕМАНТИЧЕСКИХ БАЗАХ ДАННЫХ

Хоанг Ван Куэт

(Национальный Исследовательский Томский Политехнический Университет)

Предлагается метод контроля доступа пользователей к семантическим базам данных. Выделены две задачи контроля доступа: контроль прямого доступа к данным и контроль логических выводов на получаемых данных. Рассмотрены основные свойства отношений между объектами в семантических данных. Предложен метод контроля логических выводов несанкционированных пользователей.

Ключевые слова: Семантические данные, триплет, управление логическими выводами, несанкционированный доступ.

1. Введение

При повышении возможности взаимодействия между источниками данными, увеличении количества баз данных и упрощении способов использования данных пользователями очень важной становится проблема защиты данных в несанкционированном использовании. В связи с этим задачи контроля доступа к данным и их использования с учетом заданных прав пользователей являются очень актуальными.

В тоже время, в последние годы очень активно развивается такое новое направление информатики, как семантические технологии, которые начинают играть важную роль в разработке информационных систем организаций. Применяемые в этих технологиях семантические данные отличаются от реляционных данных тем, что на их основе может выполняться логический вывод, позволяющий получать новую информацию. Для обеспе-

чения безопасности по работе с такими данными предложены разные подходы [2], но они не предоставляют возможности контроля доступа к частям данных и проверки возможности выполнения пользователями на доступных данных не желательных логических выводов. Для безопасности семантических данных должен быть обеспечен контроль:

- прямого доступа пользователей к различным частям данных, определённых политиками доступа к данным для каждого пользователя (управление прямого доступа к данным);
- доступа к данным, на основе которых могут быть логически выведены данные, не доступные пользователю (управление возможными логическими выводами).

В данной статье рассмотрен метод контроля логических выводов пользователей с помощью алгоритма обнаружения их нарушений для решения второй задачи.

2. Управление логическими выводами

В данном разделе даются основные определения логических правил и видов графов-данных, описываются предлагаемые алгоритмы обнаружения и контроля логических выводов в семантической базе данных.

2.1. ОСНОВНЫЕ ЛОГИЧЕСКИЕ ПРАВИЛА

Обозначим множество всех ресурсов (объект или субъект) как множество $P = \{A, B, C, \dots, Z\}$, а множество всех бинарных отношений как $X = \{X_1, X_2, \dots, X_n\}$, где $n \geq 1$. Для получения логических выводов пользователи используют следующие логические правила:

➤ симметричность: если $X_i \in X$ является симметричным бинарным отношением, то для любых объектов $A, B \in P$ выполняется выражение: $AX_iB \rightarrow BX_iA$;

➤ транзитивность: если $X_i \in X$ является транзитивным бинарным отношением, то для любых объектов $A, B, C \in P$ выполняется выражение: $(AX_iB \text{ и } BX_iC) \rightarrow AX_iC$, где $A \neq B \neq C$;

➤ импликация: если бинарное отношение $X_i \in X$ включает в себе отношение $X_j \in X$, то для любых объектов $A, B \in P$ выполняется выражение: $AX_iB \rightarrow AX_jB$;

➤ корреляция: если $X_i \in X$ является корреляционным бинарным отношением, то для любых объектов $A, B, C, D \in P$ выполняется выражение: $(AX_iB, CX_iD) \rightarrow AX_iC, A \neq C, B \neq C, A \neq D, B \neq D$;

➤ декомпозиция: если $X_i \in X$ является разложимым бинарным отношением, то для любых объектов $A, B \in P$ выполняется выражение: $AX_iB \rightarrow AX_iA, BX_iB$, где $A \neq B$;

➤ левая импликация: если $X_i \in X$ лево включает в себе отношение $X_j \in X$, то любых объектов $A, B \in P$ выполняется выражение: $AX_iB \rightarrow AX_jA, A \neq B$;

➤ правая импликация: если $X_i \in X$ справа включает в себе отношение X_j , то любых объектов $A, B \in P$ выполняется выражение: $AX_iB \rightarrow BX_jB, A \neq B$.

2.2. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ ГРАФОВ

Определение 1 (RDF-граф). Семантическая база данных представляет собой RDF-граф G , который состоит из множества вершин P (множество субъектов и объектов) и множества рёбер E (множество предикатов), обозначается $G = (P, E)$, где: каждая вершина является субъектом или объектом, каждое ребро является предикатом (отношение между субъектом и объектом), существуют некоторые рёбра между двумя вершинами [4].

Определение 2 (видимый граф). Видимый граф для пользователя, имеющего уровень доступа AC_s , является графом G_s , представляющим все триплеты, состоящие из троек: субъект, предикат, объект, к которым пользователь может иметь доступ, где $G_s \subseteq G$.

Определение 3 (логический граф). Логическим графом G_s^1 является результат применения основных логических правил и добавления полученных результатов к графу G_s , где $G_s \subseteq G_s^1$.

Определение 4 (поток информации вершины графа). Поток информации для вершины A является множеством всех рёбер, непосредственно связанных с A . Данной поток обозначается как $C(A)$.

Правило 1 (возможность выполнения логической операции между двумя вершинами). Предположим, что в графе G_s^1 вершины A и B имеют потоки информации $C(A)$ и $C(B)$, где $C(A)$,

$C(B) \subseteq G_s^1$. Если $C(A) \cap C(B) = \emptyset$, то пользователь не может получить логические выводы из этих известных ему вершин и их связей [3]. Из вышесказанного можно составить алгоритм для определения возможности возникновения логических выводов из известных связей и вершин.

Алгоритм 1 (алгоритм определения возможности получения логических выводов между двумя вершинами).

С использованием правила 1 для определения возможности получения логических выводов между двумя вершинами разработан алгоритм, показанный на рис. 2.

В соответствии с данным алгоритмом, если между вершинами A и B обнаружены связи, то пользователь может применить логические правила и получать логические выводы из данных вершин. В противном случае, пользователь не может получить логических выводов из вершин A и B . Входными данными алгоритма 1 являются множества всех вершин и рёбер графа. Выходными данными являются вершины, между которыми имеется возможность выполнять логический вывод.

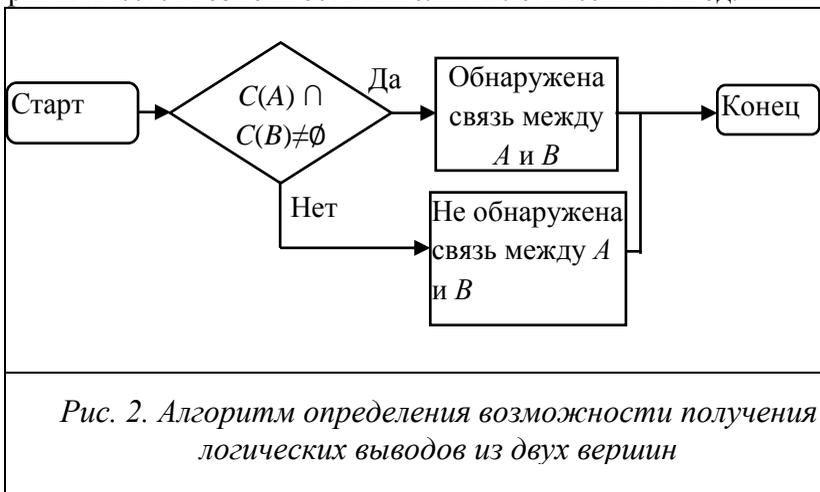


Рис. 2. Алгоритм определения возможности получения логических выводов из двух вершин

Алгоритм 2 (определение возможности получения логических выводов между двумя вершинами A и B , имеющимися разные максимальные уровни безопасности).

Предположим, что в графе G_s^1 вершины A и B имеются потоки информации $C(A)$ и $C(B)$, где $C(A), C(B) \subseteq G_s^1$. $C(A)$ имеет

максимальный уровень m , $C(B)$ имеет максимальный уровень n , где $n \geq m$. Пользователь имеет уровень доступа $AC_s = cl$ и не имеет доступа к какой-то части данных. Для определения возможности получения логических выводов между вершинами A и B был построен алгоритм, показанный на рис. 3.

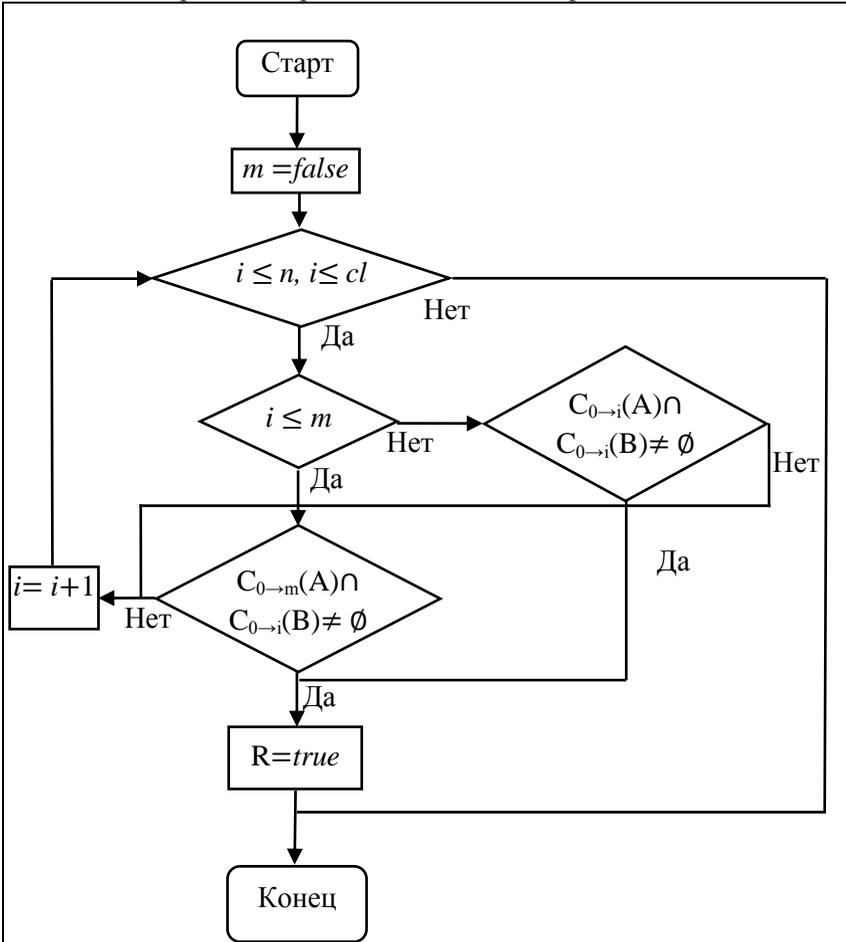


Рис. 3. Алгоритм определения возможности получения логических выводов из двух вершин, имеющих разные уровни права доступа

В соответствие с данным алгоритмом выполняются следующие основные операции:

➤ поиск общего потока информации между A и B на уровнях, которые меньше, чем m . Если между ними обнаружены общий поток информации, то операция прекращается.

➤ поиск общего потока информации между A и B на уровнях, которые меньше, чем cl . Если между ними обнаружены общий поток информации, то операция прекращается.

В данном алгоритме используется переменная R для сохранения получаемого результата: если $R = true$, то можно получить связь между A и B ; если $R = false$, то нельзя получить связь между A и B .

Определение 5 (несанкционированный логический вывод). В семантической базе данных для любого пользователя s , имеющего уровень доступа AC_s , несанкционированным логическим выводом является процесс добавления дополнительного ребра e к видимому графу G_s , где $e \in G \setminus G_s$.

Из этого определения следует, что вывод является нарушенным только в случае, когда полученная связь между вершинами находится в невидимой части данных.

Определение 6 (обнаружение нарушения логических выводов). Обнаружение нарушения логических выводов является процессом поиска всех связей, находящихся в невидимой части графа $G \setminus G_s$.

2.3. АЛГОРИТМ ОБНАРУЖЕНИЯ НАРУШЕНИЯ ЛОГИЧЕСКИХ ВЫВОДОВ

Для конкретного уровня права доступа пользователей, после обнаружения выводов, все связи будут разделены на два типа: раскрытые связи (могут являться несанкционированными логическими выводом) и безопасные связи (не могут являться несанкционированными логическими выводом).

Для обнаружения несанкционированных логических выводов был построен алгоритм (алгоритм 3), включающий следующие шаги:

1. Определение видимых частей графа пользователем: Определение связей, у которых уровень меньше, чем уровень доступа пользователя [1]. Определение связей, у которых уро-

вень больше, чем уровень доступа пользователя, эти связи будут отмечены.

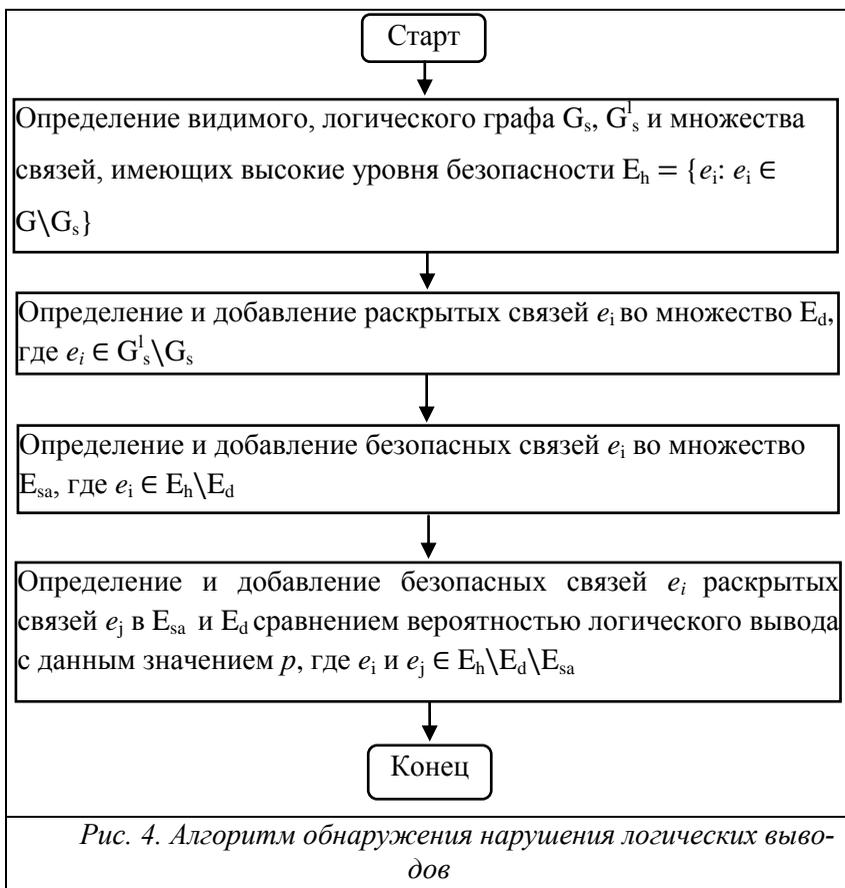
2. Применение основных логических правил для получения логических графов G^l 's. Если логические выводы находятся в невидимых частях графа, то эти связи являются раскрытыми, и они будут отмечены. После этого процесса, остальные связи, имеющие высокой уровень безопасности (не находятся в невидимых частях графа), называются обычными связями.

3. Применение правил 1 для отмеченных вершин логического графа G^l_s , если существует одна или несколько связей между двумя вершинами, то возможно, с некоторой вероятностью, применять логические правила. Если невозможно получить вывод на более высоком уровне безопасности между двумя вершинами, то все связи между ними являются *безопасными*. Если возможно получить вывод на более высоком уровне безопасности между двумя вершинами, то все связи между ними являются *подозреваемыми*.

4. Если существуют *подозреваемые связи*, то необходимо использовать теорию вероятностей для расчёта вероятности их выполнения. Такая возможность называется *вероятностным выводом*. Если вероятность вывода больше или равна *заданной вероятности* p (пороговое значение), то эти подозреваемые связи помечены как раскрытые связи, в противном случае они помечаются как безопасные связи.

На рис. 4 показан алгоритм обнаружения нарушения логических выводов. В данном алгоритме множество всех связей, имеющих более высокие уровни безопасности, чем уровень доступа пользователя AC_s , обозначено, как E_h , где $E_h = \{e_i: e_i \in G \setminus G_s, \text{ где } e_i \text{ является ребром}\}$, а множества всех раскрытых и безопасных связей, имеющих высокие уровни безопасности, обозначаются, как E_d и E_{sa} .

С помощью данного алгоритма могут быть определены все безопасные и раскрытые связи. Пользователь не может получить логической вывод только в случае, когда все вершины, влияющие на данные связи, отмечены как невидимые.



3. Пример применения предложенного алгоритма для решения задачи

Имеется семантическая база данных, которая содержит множество триплетов со своими индексами безопасности $R = \{R_{AoB}^1(0100), R_A^5(0110), R_{AoB}^9(1100), R_{ABo}^8(0000), R_{AoE}^9(1111), R_{BoBE}^1(0000), R_{BCo}^2(1110), R_{CoB}^6(0100), R_{CoB}^3(1100), R_{DoC}^7(0110), R_D^{10}(1100), R_{DEo}^4(1110)\}$, где $R_{AoB}^y(S, P, PS, C)$ означает, что R является утверждением о том, что элемент A связан с элементом B использованием бинарного отношения y , R имеет индекс

безопасности $AC = (S, P, PS, C)$, включающий следующие 4 показателя:

- *чувствительность S* – определяет уровень значимости или важности связи (предиката), а также её уязвимости перед несанкционированным лицом;
- *приватность P* – определяет права владельца на возможность передачи данной информации другим пользователям;
- *персональная безопасность PS* – определяет уровень защиты персональной информации человека или организации;
- *конфиденциальность C* – задаёт возможность совместного использования данной информации с другими ресурсами.

Каждый показатель S, P, PS, C может принимать значения из множества меток безопасности $L = \{\text{неклассифицированный } (L_U), \text{конфиденциальный } (L_C), \text{секретный } (L_S) \text{ и сверхсекретный } (L_{TS})\}$, где $L_U < L_C < L_S < L_{TS}$. В данной задаче предлагается, что эти показатели могут принимать значение 0 (unclassified), 1 (confidential). Право доступа пользователей $AC_s = (S_s, P_s, PS_s, C_s)$ к данным так же включает 4 показателя: чувствительность, приватность, персональная безопасность и конфиденциальность. В данной задаче предлагается, что пользователь имеет уровень доступа $AC_s = (1100)$.

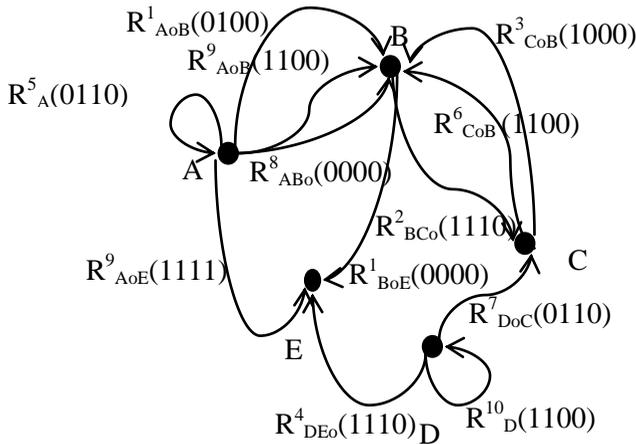


Рис. 5. Граф связи между элементами триплетов в базе данных

На рис. 5 показан данный фрагмент RDF-графа. Элементы множества отношений $\{X_1, X_2, \dots, X_{10}\}$ базы данных обладают следующими свойствами: X_1 является симметричной связью, X_3 является транзитивной связью, $\langle X_6, X_8 \rangle$ является доминирующим правилом, (в данной задаче $R^6_{BC} \rightarrow R^8_{BC}$). $\langle X_9, X_5 \rangle$ левое доминирующее правило, т. е., если R_i левое доминирующее R_j , то $xR_i y \rightarrow xR_j x$, $x \neq y$; (в данной задаче, $R^9_{AoB} \rightarrow R^5_A$).

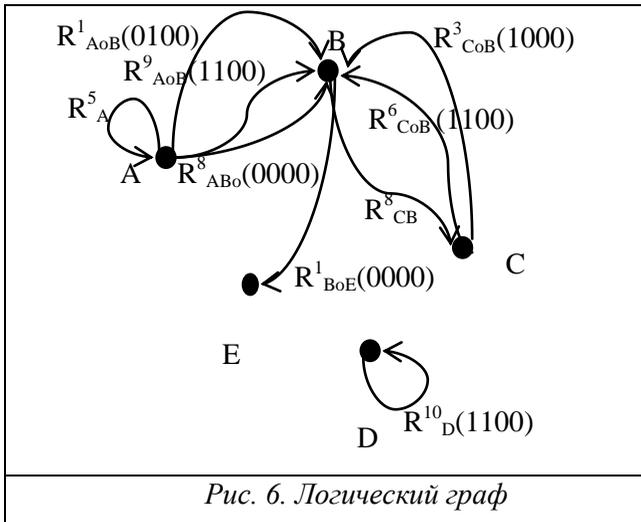
Требуется применить алгоритм 3 для решения задачи безопасности семантической базы данных.

Решение

Шаг 1: определение видимых триплетов (граф G_s).

При прямой проверки права доступа пользователя, нельзя увидеть триплеты $R^5_A(0110)$, $R^9_{AoB}(1100)$, $R^4_{DEo}(1110)$, $R^7_{DoC}(0110)$, $R^2_{BCo}(1110)$, потому что их уровень доступа меньше, чем уровень безопасности триплетов.

Шаг 2: определение логических и видимых триплетов (граф G^1_s).



По заданным правилам A_{AB}^9 доминирует над R_{A}^5 и R_{CB}^6 доминирует над R_{CB}^8 , получим логический граф, представленный на рис 6.

Шаг 3: определение раскрытых, безопасных и подозреваемых триплетов.

Один из двух логических результатов R_{A}^5 является нарушенным триплетом, потому что $R_{A}^5 \in G \setminus G_s$. Тогда получаются: раскрытая связь R_{A}^5 и неясные связи $R_{AoE}^9(1111)$, $R_{DEo}^4(1110)$, $R_{DoC}^7(0110)$, $R_{BC}^2(1110)$. Для контроля неясных связей, необходимо проверить такие пары вершин, как: $\{<A, E>, <D, E>, <D, C>, <C, B>\}$.

Так как известно, что:

- $C(A) \cap C(E) \neq \emptyset$ (между вершинами A и E существует общая вершина B), значит из них могут следовать другой триплет;
- $C(D) \cap C(E) = \emptyset$, значит из них не могут прямо следовать другой триплет;
- $C(D) \cap C(C) = \emptyset$, значит из них не могут следовать другой триплет;
- $C(B) \cap C(C) \neq \emptyset$, значит из них могут следовать другой триплет.

После этого шага получается множество нарушенной связи $\{R_{A}^5\}$, множество безопасных связей $\{R_{DEo}^4(1110), R_{DoC}^7(0110)\}$ и множество подозреваемых связей $\{R_{AoE}^9(1111), R_{BC}^2(1110)\}$.

Шаг 4: контроль наследованных триплетов.

Контроль первой подозреваемой связи $R_{AoE}^9(1111)$ между вершинами A и E:

Основными триплетами, не влияющими на подозреваемую связь $R_{AoE}^9(1111)$, являются $\{R_{CB}^3, R_{CB}^6, R_{CB}^8\}$. Основными триплетами, влияющими на подозреваемую связь $R_{AoE}^9(1111)$, являются $\{R_{AB}^1, R_{AB}^9, R_{AB}^8, R_{BE}^8\}$.

Тогда можно определить пары наследования $\{R_{AB}^1, R_{BoE}^1\}$, $\{R_{AB}^9, R_{BoE}^1\}$, $\{R_{ABo}^8, R_{BoE}^1\}$. Из этих пар пользователь может наследовать R_{AoE}^9 , с некоторой вероятностью.

Для того, чтобы пользователь абсолютно не смог наследовать R_{AoE}^9 , в этих парах не должен присутствовать один из

элементов. В этом случае, пользователю нельзя видеть триплет R^1_{BoE} , или триплеты $\{R^1_{\text{AB}}, R^9_{\text{AB}}, R^8_{\text{ABo}}\}$.

Контроль второй подозреваемой R^2_{BC} :

Связи $R^1_{\text{AB}}, R^8_{\text{AB}}, R^9_{\text{AB}}, R^1_{\text{BE}}$ являются не связанными с подозреваемой связью R^2_{BC} . Связи $R^3_{\text{CB}}, R^6_{\text{CB}}, R^8_{\text{CB}}$ являются триплетами, связанными с подозреваемой связью R^2_{BC} . Из этих триплетов пользователь может наследовать R^2_{BC} с какой-то вероятностью. Для того чтобы пользователь абсолютно не мог наследовать R^2_{BC} , для него должны быть невидимыми триплеты $R^3_{\text{CB}}, R^6_{\text{CB}}, R^8_{\text{CB}}$.

Выводы

Сложность работы с семантическими базами данных заключается в том, что на основе полученных результатов можно делать логические выводы, позволяющие пользователям получать не разрешенные для него данные. В связи с этим при поддержке работы пользователей с такими базами данных необходимо выполнять управление логическими выводами на получаемых результатах.

Решение данной задачи возможно путем использования предложенного в статье алгоритма определения возможности получения логических выводов из двух вершин, имеющих разные уровни права доступа. Корректное использование разработанного алгоритма гарантирует защиту информации семантических баз данных

Литература

1. ДЕВЯНИН П. Н. *Анализ безопасности управления доступом и информационными потоками в компьютерных системах*. — М.: Радио и связь, 2006. — 176 с.
2. ДЕВЯНИН П. Н. *Модели безопасности компьютерных систем*. — М.: Академия, 2005. — 144 с.
3. MORGENSTERN, M. A. Controlling logical inference in multilevel database systems - On Security and privacy, 1998. — 245 p.

4. THURASINGHAM, B. H. *Building trustworthy semantic webs, Technical Report, University of Texas – Department of Computer Science, 2008. – 434 p.*

CONTROL INFERENCE IN SEMANTIC DATABASES

Hoang Van Quyet: National Research Tomsk Polytechnic University, postgraduate (student8050@sibmail.com).

Abstract: This paper proposes method for controlling access to semantic database. Two problems of access control were identified: direct access control and inference control. The basic definitions of relations between objects in the semantic data are concerned. A method of controlling unauthorized inferences is created.

Keywords: Semantic data, triple, inference control, unauthorized access.