УДК 681.3.06

К.В. Хоанг, А.Ф. Тузовский

Решения основных задач в разработке программы поддержки безопасности работы с семантическими базами данных

Рассматриваются вопросы обеспечения безопасности информационных систем. Поставлена задача поддержки безопасности работы с семантическими базами данных. Описаны предлагаемые алгоритмы их решения.

Ключевые слова: семантические данные, онтология, алгоритм, программа.

Информационные базы данных широко применяются в самых разных областях человеческой деятельности, таких как военная, коммерческая, банковская и научная. Очевидно, что информация, хранящаяся в этих базах данных (БД), должна быть надёжно защищена от доступа лиц, не имеющих права на работу с ней.

Одним из современных направлений развития информационных технологий является переход к работе с семантикой информации и создание семантических БД [1]. Основным преимуществом семантических БД является то, что на основе их содержания могут выполняться логические выводы (ЛВ), позволяющие получить новую информацию, которая непосредственно в них не содержится.

Так как создание и использование семантических БД и их практическое применение начались недавно, то вопрос обеспечения их безопасности решен ещё не в полной мере. В связи с этим актуальным является разработка методов и алгоритмов, позволяющих поддерживать безопасность работы пользователей с семантическими БД. Целью данной статьи как раз и является решение таких задач.

Постановка задачи. В настоящее время разработаны классические подходы (политики) обеспечения безопасности баз данных, такие, как дискреционная, мандатная и политика ролевого разграничения доступа [2].

Дискреционная политика безопасности (DAC – discretionary access control) основывается на дискреционном (необязательном, выполняемом по усмотрению программы) управлении доступом и определяется следующими двумя свойствами: все субъекты и объекты идентифицируются; права доступа субъектов к объектам системы определяются на основании некоторых внешних по отношению к системе правил.

Мандатная политика безопасности (MAC – mandatory access control) основывается на мандатном (принудительном) разграничении доступа, определяющемся четырьмя условиями: все субъекты и объекты системы идентифицируются; задается решетка уровней безопасности информации; каждому объекту системы присваивается уровень безопасности, определяющий важность содержащейся в нем информации; каждому субъекту системы присваивается уровень доступа, определяющий уровень доверия к нему в компьютерной системе.

Для политики доступа MAC задача проверки безопасности является алгоритмически разрешённой. Кроме того, по сравнению с политикой DAC, мандатная политика имеет более высокую степень надёжности. Правила данной политики являются более простыми и понятными для разработчиков и пользователей, что также положительно влияет на уровень безопасности системы.

Политика ролевого разграничения доступа (RBAC – Role-based access control) является развитием политики DAC. В политике RBAC права доступа субъектов системы на объекты группируются с учётом специфики их применения, образуя роли. Задание ролей позволяет определить более чёткие и понятые для пользователей компьютерной системы правила разграничения доступа. RBAC позволяет реализовать гибкие, изменяющиеся динамически в процессе функционирования компьютерной системы правила разграничения доступа. С другой стороны, реализация систем с политикой безопасности данного типа довольно сложна и требует значительных ресурсов компьютерной системы, а также она может быть реализована с использованием политики MAC.

Модели DAC и MAC имеют одно фундаментальное отличие от модели RBAC, заключающееся в том, что первые две модели заранее определяют политику безопасности системы и позволяют её

настраивать для каждой конкретной ситуации. А модель RBAC не предопределяет политику безопасности, а позволяет её настроить так, как это требуется организации. Такая настройка производится в два этапа: настройка политики безопасности системы и определение прав доступа для субъектов и объектов в системе.

Из вышесказанного можно сделать вывод о том, что применение политики MAC путём использования меток безопасности для контроля базы данных, описывающей содержимое конкретной области и имеющей определённую структуру, позволяет реализовать простым способом безопасность доступа не менее надежную, чем политики DAC и RBAC.

В связи с этим для контроля доступа пользователей U при работе с семантическими БД предлагается использовать политику MAC. В соответствии с этой моделью предполагается, что в семантических данных каждому элементу онтологической модели (онтологии) и содержащимся в ней утверждениям задаются уровни безопасности, значения которых выбираются из множества меток, например, таких как открытый ($L=1$), конфиденциальный ($L=2$), секретный ($L=3$), сверхсекретный ($L=4$). Пример размеченной онтологии показан на рис. 1.

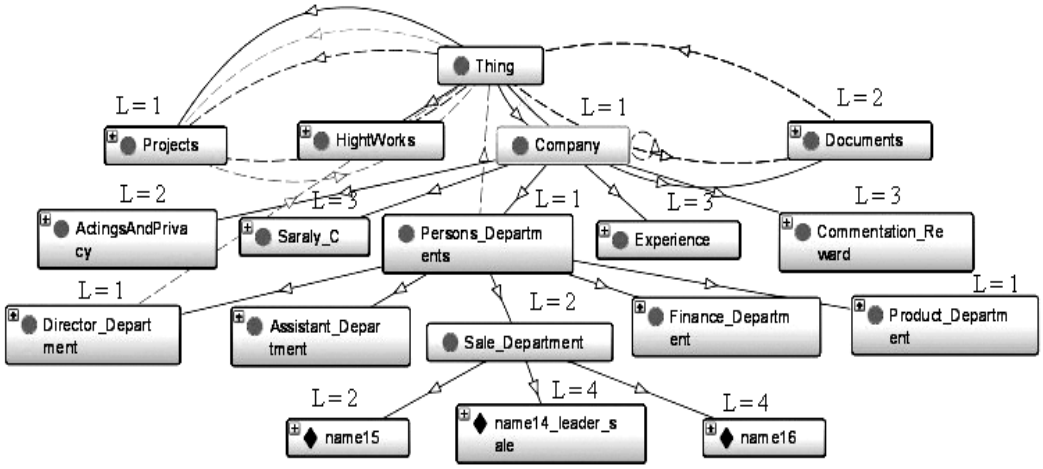

Рис. 1. Пример онтологии с уровнями безопасности

Всем пользователям семантической БД также задается доступный им уровень безопасности L_u. Сущность использования MAC для обеспечения безопасности работы с семантическими БД заключается в сравнении уровней доступа пользователей L_u с уровнями безопасности L элементов, таких как классы (понятия), свойства, атрибуты, индивидуумы (экземпляры) онтологии [3] или RDF-триплетов [4] (RDF – resource description framework) в БД. Из этого следует, что для всех элементов семантической БД требуется задавать согласованные уровни безопасности, что требует создания специальных алгоритмов.

Кроме этого, пользователи семантической БД могут применять различные логические правила (ЛП), описанные с помощью языка RIF (RIF – rule interchange format) [5]. А это может привести к тому, что они получают возможность вывода данных, превышающих заданный им уровень доступа. В связи с этим возникает проблема выявления ситуаций, в которых получаемые логические выводы будут нарушать заданные пользователям уровни доступа. Для решения данной проблемы требуется разработать алгоритмы контроля выполнения логических выводов в семантических данных.

Таким образом, для обеспечения безопасности работы пользователей с семантическими данными требуется разработать методы определения уровней безопасности всех элементов онтологии и методы контроля возможности получения пользователями недопустимых логических выводов.

Определение значений уровней безопасности элементов в семантической БД. Семантическая БД состоит из онтологий O и множества RDF-триплетов. Основными элементами онтологии являются понятия, атрибуты, отношения и индивидуумы. Построение алгоритмов, позволяющих определять уровни безопасности L_a элементов K_a онтологии O, должно основываться на следующих принципах:

- в онтологиях нет элементов, не имеющих уровней безопасности;
- если элементу K_a не создан начальный уровень безопасности L_a, то его уровень безопасности равен нулю, т.е. $L_a = 0$;

• уровень безопасности L_a подкласса K_a должен быть больше или равен уровню безопасности L_b суперкласса K_b, т.е. $L_a \geq L_b$;

• уровень безопасности L_a объекта K_a должен быть больше или равен уровням безопасности L_b классов K_b, которым он принадлежит;

• уровень безопасности L_a свойства K_a должен доминировать над уровнем безопасности L_b других свойств K_b, которым оно принадлежит;

• в зависимости от логических операций каждый индивидуум может принадлежать нескольким классам, следовательно, ЛВ могут обладать некоторыми значениями уровней безопасности.

Общая схема данных алгоритмов показана на рис. 2.

Чтобы определить уровень безопасности класса онтологии, используется алгоритм, в схеме которого класс и его уровень безопасности обозначаются как K_a и L_a, а K_b и L_b используются для обозначения его суперклассов и их уровня безопасности.

В схеме алгоритма, использующегося для определения уровней безопасности свойств онтологии, K_a и L_a используются для обозначения свойства и его уровня безопасности, а свойство, которому K_a принадлежит, и его уровень безопасности обозначаются как K_b и L_b.

Для определения уровней безопасности индивидуумов онтологии может использоваться алгоритм, в схеме которого индивидуум и его уровень безопасности обозначаются как K_a и L_a, а класс, которому индивидуум K_a принадлежит, и его уровень безопасности обозначаются как K_b и L_b.

В результате использования описанных алгоритмов может быть определен уровень безопасности каждого понятия, свойства, атрибута и индивидуума.

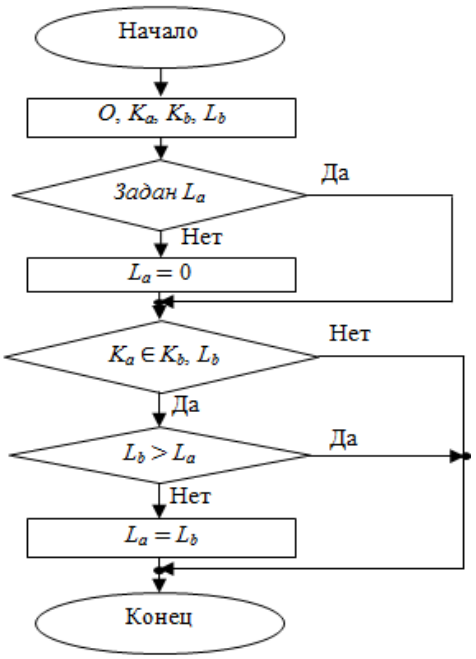

Рис. 2. Общая схема алгоритма для определения уровня безопасности элементов онтологий

В семантических БД элемент может принадлежать другим элементам или содержать другие элементы. В связи с этим операции разработанных алгоритмов для определения уровней безопасности всех элементов онтологий необходимо выполнять рекурсивно. Для выполнения таких рекурсивных операций может быть использован язык RIF. На рис. 3 показаны правила, описанные на языке RIF, позволяющие определить уровни безопасности всех классов онтологии.

$$(?K_a \text{ rdf:type owl:Class}) \wedge \text{noValue}(?K_a \text{ ontology:Level } ?L_a) \rightarrow (?K_a \text{ ontology:Level 0}).$$
$$(?K_a \text{ ontology:Level } ?L_a) \wedge (?K_b \text{ ontology:Level } ?L_b) \wedge (?K_a \text{ rdfs:subClassOf } ?K_b) \wedge$$
$$\text{greaterThan}(?L_b, ?L_a) \rightarrow \text{drop}(1) \wedge (?K_a \text{ ontology:Level } ?L_b)$$

Рис. 3. Определение уровня безопасности классов онтологии с использованием правил языка RIF

На рис. 4 показан метод определения уровней безопасности для всех свойств онтологии с помощью правил, описанных на языке RIF.

$$(?K_a \text{ rdf:type owl:ObjectProperty}) \wedge \text{noValue}(?K_a \text{ ontology:Level } ?L_a) \rightarrow (?K_a \text{ ontology:Level 0}).$$
$$(?K_a \text{ ontology:Level } ?L_a) \wedge (?K_a \text{ rdfs:subPropertyOf } ?K_b) \wedge (?K_b \text{ ontology:Level } ?L_b) \wedge$$
$$\text{greaterThan}(?L_b, ?L_a) \rightarrow \text{drop}(1) \wedge (?K_a \text{ ontology:Level } ?L_b)$$

Рис. 4. Определение уровней безопасности свойств онтологии с помощью языка RIF

На рис. 5 показан метод определения уровней безопасности для всех индивидуумов онтологии с помощью правил, описанных на языке RIF.

(?K_a rdf:type owl:NamedIndividual) ∧ noValue(?K_a ontology:Level ?L_a) → (?K_a ontology:Level 0).

(?K_a ontology:Level ?L_a) ∧ (?K_a rdf:type ?K_b) ∧ (?K_b ontology:Level ?L_b) ∧

greaterThan(?L_b , ?L_a) → drop(1) ∧ (?K_a ontology:Level ?L_b)

Рис. 5. Определение уровней безопасности индивидуумов с помощью языка RIF

С помощью созданных алгоритмов могут быть определены уровни безопасности всех элементов семантической БД в соответствии с формулой $L=\mathrm{MAX}\{L_S,L_p,L_o\}$ [6], где L_S – уровень безопасности субъекта; L_p – уровень безопасности отношения; L_o – уровень безопасности объекта.

При использовании мандатного разграничения доступа для обеспечения безопасности БД, если уровень доступа пользователя больше уровней безопасности RDF-триплетов, то он имеет доступ к данным триплетам и в соответствии со своими правами может выполнить разные операции над ними. В противном случае пользователь не может иметь доступа к триплетам и не может выполнять над ними каких-либо операций.

Контроль получения логических выводов пользователями в семантических БД. Семантическая БД – это множество простых RDF-утверждений вида (s , p , o), где s – это субъект утверждения, o – объект, p – отношение между субъектом и объектом. В общем виде семантическая БД представляет собой RDF-граф Q, состоящий из множества вершин V (множество субъектов и объектов), множества рёбер P (множество отношений) и обозначается как $Q=(V,P)$.

Пользователь U, имеющий уровень доступа L_u, может отправлять запрос R, описанный на языке SPARQL (RDF Query Language), к БД для получения новой информации (новых триплетов).

Тогда видимым графом Q_S графа Q для пользователя U является граф, содержащий все триплеты, уровни безопасности которых меньше, чем L_u (пользователь имеет к ним доступ), где $Q_S \subseteq Q$. Аналогично, невидимым граф Q_h графа Q для пользователя U является граф, содержащий все триплеты, у которых уровни безопасности больше чем L_u (к которым пользователь не имеет права доступа), следовательно, $Q_h = Q \setminus Q_S$.

В семантических базах данных также содержится множество логических правил $P=\{p_1,...,p_n\}$, где p_i – логические правила, написанные на языке RIF, с помощью которых пользователь может выполнять логические выводы. Тогда логическим графом Q_S^l для пользователя U является результат применения P к видимому графу Q_S, и получено отношение $Q_S \subseteq Q_S^l$. Таким образом, множеством несанкционированных логических выводов для пользователя U является множество Q_t, элементами которых являются триплеты, которые находятся одновременно в Q_S^l и Q_h, следовательно, $Q_t = Q_S^l \cap Q_h$. Обнаружение нарушения ЛВ является процессом поиска всех связей, находящихся в невидимой части графа Q/Q_S.

Для контроля ЛВ, выполненных пользователем на запрос R, необходимо проверить, принадлежит ответ A на запрос R множеству Q_t или нет. Если A не находится в Q_t, то A является санкционированным ЛВ, их пользователь может получить. В противном случае если A находится в Q_t, то A является несанкционированным ЛВ, и пользователь не имеет права для получения данных результатов. В этом случае необходимо контролировать A. Это может быть выполнено следующим образом:

- определение всех триплетов T, использующихся в соответствующих ЛП, для получения A;
- изменение уровней безопасности T, чтобы пользователь не мог использовать их в семантических правилах.

Алгоритм контроля ЛВ в семантических БД описан в [7]. С помощью данного алгоритма могут быть определены все безопасные и опасные связи. Только логические правила, имеющие связи и вершины, у которых уровни безопасности меньше L_u, могут выполняться пользователем, следова-

тельно, он сможет получать только логические выводы, принадлежащие Q_s, что будет гарантировать безопасность семантических БД.

Апробация алгоритмов. Все разработанные алгоритмы были программно реализованы для семантической БД Virtuoso. Были проведены эксперименты, которые показали, что пользователи могут посмотреть данные, выполнить разные операции над ними в зависимости от своих прав доступа и также получить логические выводы в семантических БД в соответствии со своими уровнями доступа. Более детально описание реализации и проверки алгоритмов приведено в [8].

Заключение. В данной статье выявлены основные задачи поддержки безопасности БД и разработаны методы их решения. Для контроля доступа пользователей к данным предложено использовать мандатную политику безопасности. Реализация данной политики основывается на разработанном алгоритме определения согласованных уровней безопасности всех элементов онтологии и RDF-триплетов семантических БД. Так как семантические БД в отличие от реляционной БД позволяют выполнять правила для получения логических выводов, то для обеспечения безопасности данных разработан алгоритм, с помощью которого пользователи могут получать только данные, имеющие уровни безопасности меньше уровней доступа пользователей.

Применение разработанных алгоритмов позволяет разработать программное обеспечение, поддерживающее безопасность работы пользователей с семантическими БД.

Литература

1. Семантические технологии [Электронный ресурс]. – Режим доступа: http://www.ultimeta.ru/technologies/semantic.html, свободный (дата обращения: 10.02.2012).

2. Девянин П.Н. Анализ безопасности управления доступом и информационными потоками в компьютерных системах. – М.: Радио и связь, 2006. – 176 с.

3. OWL web ontology Language [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/owl-features/, свободный (дата обращения: 10.03.2012).

4. Resource description framework [Электронный ресурс]. – Режим доступа: http://ru.wikipedia.org/wiki/RDF/, свободный (дата обращения: 01.02.2012).

5. Rule interchange format [Электронный ресурс]. – Режим доступа: http://www.w3.org/2001/sw/wiki/RIF, свободный (дата обращения: 01.01.2013).

6. Алгоритмы для контроля доступа и модификации семантических данных // Электронные средства и системы управления. – 2012. – Т. 26, № 2. – С. 41–45.

7. Хоанг В.К. Метод контроля логических выводов в семантических базах данных // Научно-технический вестник Поволжья. – 2013. – № 1. – С. 281–286.

8. Хоанг В.К. Контроль логических выводов в семантических базах данных / В.К. Хоанг, А.Ф. Тузовский // Изв. Том. политех. ун-та. – 2012. – Т. 321, № 5. – С. 158–162.

Хоанг Ван Кует
Аспирант кафедры оптимизации систем управления НИТПУ
Тел.: 8 (382-2) 42-14-85
Эл. почта: student8050@sibmail.com

Тузовский Анатолий Фёдорович
Д-р техн. наук, профессор каф. оптимизации систем управления Института кибернетики ТПУ
Тел.: 8 (382-2) 42-14-85
Эл. почта: tuzovskyaf@tpu.ru

Hoang Van Quyet, Tuzovskiy A.F.

Resolution of the basic problems in the software development for security semantic knowledge databases

This article discusses the importance of ensuring the security for semantic databases. The tasks in the software development for support the security of semantic database is supplied, the algorithms for resolution of these tasks are created.

Keywords: semantic database, ontology, algorithm, program.