УДК 004.415

МЕТОДЫ ОПРЕДЕЛЕНИЯ УРОВНЕЙ БЕЗОПАСНОСТИ ЭЛЕМЕНТОВ ОНТОЛОГИИ

Хоанг Ван Кует, А.Ф. Тузовский

Томский политехнический университет
E-mail: student8050@sibmail.com

Рассматриваются основные проблемы многоуровневой безопасности в семантических данных. Описаны принципы задания уровней безопасности понятий, свойств и индивидуумов онтологии. Предложены алгоритмы, позволяющие определить уровни безопасности основных элементов онтологии и логических выводов, полученных в логических правилах.

Ключевые слова:
Онтология, уровень безопасности, алгоритм, логические выводы.

Key words:
Ontology, level security, algorithm, inference.

Проблема обеспечения безопасности семантических данных является достаточно актуальной в связи с активным ростом использования семантических технологий для хранения и обработки информации в информационных системах различных организаций [1]. Для решения данной проблемы предлагается многоуровневая система обеспечения безопасности семантических данных, заключающаяся в том, что каждому элементу семантической базы данных задаётся собственный уровень безопасности. В семантических данных каждый элемент одновременно может являться субъектом или объектом, а также может содержать или принадлежать другим элементам, имеющим разные уровни безопасности, следовательно, его уровни безопасности не являются постоянными [2].

Кроме этого, следует также учитывать, что при работе с семантическими данными на основе фактов, извлечённых из базы данных, пользователи могут с помощью логических правил получить новую, не разрешённую им информацию (новое утверждение), имеющую неизвестный уровень безопасности [3].

С целью обеспечения работы с семантическими данными необходимо гарантировать, что уровень каждого их элемента является определенным и единственным. Для выполнения этого требования необходимо создать метод, позволяющий определить уровни безопасности элементов данных.

Целью данной статьи является пояснение алгоритмов определения уровней безопасности каждого элемента онтологий и логических выводов, полученных при выполнении логических правил в семантических данных.

Постановка задач и основные понятия

Онтология – это формальная модель некоторой области знаний с помощью концептуальной схемы. Обычно такая схема состоит из структуры данных, содержащей все релевантные классы объектов (понятия), их связи и правила (теоремы, ограничения), принятые в этой предметной области. Онтологии используются в информационных системах как форма представления знаний о реальном мире или его части. Современные онтологии в основном строятся одинаково, независимо от языка написания, и обычно они состоят из индивидуумов, понятий, атрибутов и отношений.

В базе данных хранятся онтологии и индивидуумы. Под онтологической моделью (онтологией) O понимается знаковая система $<C,T,P>$, где $C=\{C_1,...,C_n\}$ – множество элементов, которые называются понятиями (классами), где n – количество понятий в онтологии; T – частичный порядок на множестве C, задающий отношения «подкласс» и «суперкласс»; $P=\{P_1,...,P_m\}$ – множество элементов, называющихся свойствами (атрибутами, отношениями), где m – количество свойств в онтологии (двуместными предикатами) [4]. Множество индивидуумов обозначается как $E=\{E_1,...,E_k\}$, где k – количество индивидуумов.

С целью безопасности доступа к семантическим данным элементу в онтологии и утверждению в базе данных задаются уровни безопасности s, значения которых выбираются из множества меток, например таких, как {неклассифицированный (s=1), конфиденциальный (s=2), секретный (s=3), сверхсекретный (s=4)}. Таким образом, множеству элементов онтологии O соответствует множество уровней безопасности, которое обозначается, как $<S_C,S_P,S_E>$, где S_C – множество уровней безопасности понятий, S_P – множество уровней безопасности свойств и S_E – множество уровней безопасности индивидуумов [5]. На рис. 1 показана онтология, у которой элементы имеют 4 уровня безопасности.

В онтологии класс C_a может являться подклассом класса C_b или суперклассом класса C_c. В процессе создания онтологий для классов C_a, C_b и C_c разработчики могут задать уровни безопасности s_a, s_b и s_c. Таким образом, для класса онтологии может быть задано несколько или ни одного значения уровней безопасности. В этом случае актуальной является задача определения уровня безопасности для каждого элемента онтологии.

Для работы с семантикой информации может быть использован язык Semantic Web Rule Language (SWRL), основанный на объединении языков

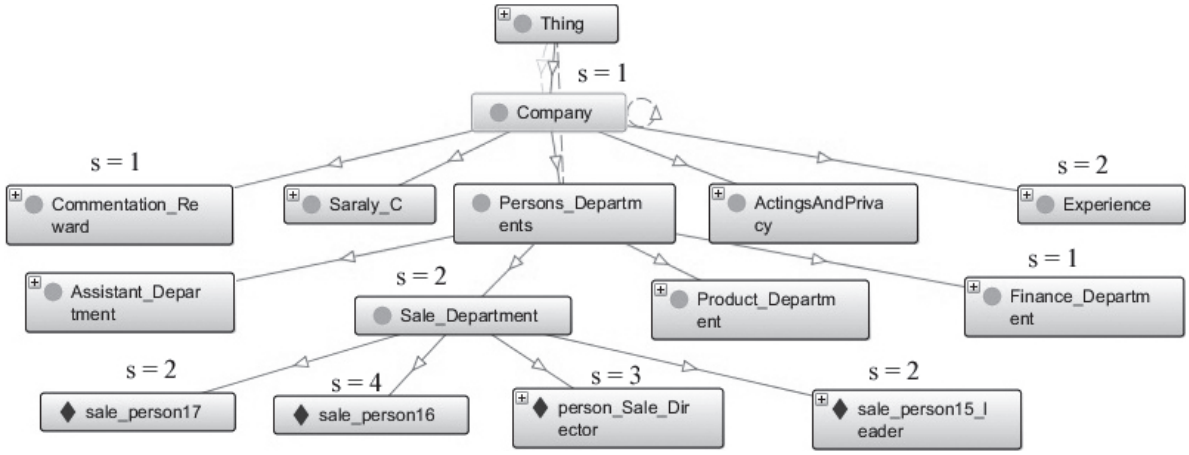

Рис. 1. *Пример онтологии с уровнями безопасности*

OWL и RuleML, т. е. объединены онтологии (OWL-DL) и правила [6]. Он позволяет формально описать способ работы с объектами предметной области, а так же закономерности предметной области. С помощью данного языка на основе уже известных семантических данных могут быть получены новые утверждения (т. е. сделаны логические выводы). На рис. 2 показан пример логического правила, написанного на языке SWRL.

```
(?a rdf: type owl: NamedIndividual) ∧
(?b rdf: type owl: NamedIndividual) ∧
(?a rdf: type ontology: Finance_Department) ∧
(?b rdf: type ontology: Finance_Department) ∧
(?a ontology: isLeaderDepartmentOf ?b)
→ (?a rdf: type ontology: Leader_Finance)
```

Рис. 2. *Пример описания логических выводов на языке SWRL*

В данном примере описывается следующее правило: если два человека *a* и *b* работают в финансовом отделе, и *a* руководит *b*, то следовательно *a* является начальником данного отдела.

Теперь предположим, что *a* имеет уровень безопасности s_i=1, а классу *Leader-Finance* задан уровень безопасности s_j=2, тогда в соответствии с данным логическим правилом *a* является элементом данного класса и ему должен быть задан уровень безопасности s_i=s_j=2. Таким образом, необходимо создать метод для определения уровней безопасности элементов, полученных в результате применения логических правил.

Алгоритмы определения уровней безопасности элементов в онтологии

Для построения алгоритмов, позволяющих определить уровни безопасности элементов онтологии, предположим следующие принципы:

- В онтологиях нет элементов, не имеющих уровней безопасности.
- Если элементу не адан начальный уровень безопасности, то его уровень безопасности равен нулю.
- Уровень безопасности подкласса должен быть больше или равен уровню безопасности суперкласса.
- Уровень безопасности объекта должен быть больше или равен уровням безопасности классов, которым он принадлежит.
- Уровень безопасности свойства должен доминировать над уровнем безопасности других свойств, которым оно принадлежит.
- В зависимости от логических операций, каждый индивидуум может принадлежать нескольким классам, следовательно, логические выводы могут обладать некоторыми значениями уровней безопасности.

Алгоритм определения уровней безопасности классов

Понятия или классы – это абстрактные группы, коллекции или наборы объектов. Они могут включать в себя индивидуумы, другие классы, либо же сочетания и того, и другого. Классы онтологии составляют таксономию (иерархию понятий) на основе отношения включения (*SubclassOf*). На основе вышеуказанных принципов, алгоритм (алгоритм 1) определения уровней безопасности понятий в онтологии может быть описан следующим образом:

1. Если начальный уровень безопасности s_x понятия C_x не создан, то ему присвоено значение равное нулю (s_x=0).
2. Если понятие C_x является подклассом другого понятия C_y, имеющего уровень безопасности s_y, то необходимо сравнять уровень безопасности s_x с уровнем безопасности s_y.
3. Если s_y>s_x, то s_x будет присвоено значение s_y, т. е. s_x=s_y, в противном случае s_x=s_x и алгоритм заканчивается.
4. Если понятие C_x не является подклассом никакого другого понятия C_y, то алгоритм заканчивается.

Схема алгоритма, позволяющего определять уровень безопасности понятия, показана на рис. 3. В результате выполнения алгоритма уровень безо-

пасности каждого понятия будет определён и будет единственным.

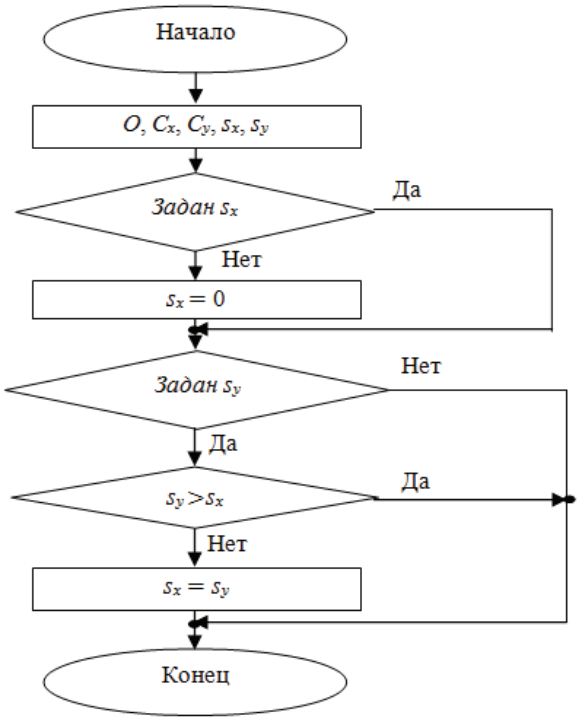

Рис. 3. *Алгоритм определения уровня безопасности понятия*

В онтологии класс может одновременно являться подклассом одного класса или суперклассом другого, тогда для определения уровней безопасности всех классов необходимо выполнять рекурсивные операции для разработанного алгоритма. На рис. 4 показано описание метода определения уровней безопасности для всех классов в онтологии с помощью языка SWRL.

```
(?Cx rdf: type owl: Class) ∧
noValue (?Cx ontology: Level ?sx)→
(?Cx ontology: Level 0).
(?Cx ontology: Level ?sx) ∧
(?Cy ontology: Level ?sy) ∧
(?Cx rdfs: subClassOf ?Cy) ∧
greaterThan (?sy,?sx)→ drop (1) ∧
(?Cx ontology: Level ?sy).
```

Рис. 4. *Определение уровня безопасности классов онтологии с помощью языка SWRL*

Определение уровней безопасности свойств

Объекты в онтологии могут иметь атрибуты. Каждый атрибут имеет, по крайней мере, имя и значение и используется для хранения информации, которая специфична для объекта и связана с ним. Важная роль атрибутов заключается в том, чтобы определять отношения между элементами онтологии. Обычно под отношением понимается атрибут, значением которого является какой-то объект.

В онтологии между свойствами также могут задаваться отношения, например, одно свойство может включать в себя другие свойства или принадлежать им. Уровень безопасности свойства может быть описан в процессе создания онтологии или определён с помощью уровней безопасности других элементов. Способ, позволяющий определить уровни безопасности свойств онтологии, аналогичен подходу, описанному в алгоритме 1. Для этого также необходимо рекурсивно выполнять алгоритм, используемый для определения уровня безопасности одного свойства. На рис. 5 показан способ определения уровней безопасности s_x всех свойств P_x в онтологии, описанный с помощью языка SWRL.

```
(?Px rdf: type owl: ObjectProperty) ∧
noValue (?Px ontology: Level ?sx)→
(?Px ontology: Level 0).
(?Px ontology: Level ?sx) ∧
(?Px rdfs: subPropertyOf ?Py) ∧
(?Py ontology: Level ?sy) ∧
greaterThan (?sy,?sx)→ drop (1) ∧
(?Px ontology: Level ?sy).
```

Рис. 5. *Определение уровней безопасности свойств онтологии с помощью языка SWRL*

Алгоритм определения уровней безопасности индивидуумов в онтологии

Индивидуумы (экземпляры) – это основные элементы нижнего уровня онтологии. Индивидуумы могут соответствовать как физическим объектам (люди, дома, планеты), так и абстрактным сущностям (числа, слова).

Аналогично алгоритму 1 алгоритм определения уровня безопасности индивидуума создан таким образом:

- если начальный уровень безопасности (s_x) индивидуума (E_x) не задан, то его значение равно нулю (s_x=0);
- если индивидуум E_x является элементом класса C_y, имеющего уровень безопасности s_y, то необходимо сравнять s_x с уровнем безопасности s_y;
- если $s_x<s_y$, то s_x будет присвоено значение s_y, т. е. $s_x=s_y$, в противном случае $s_x=s_x$ и алгоритм заканчивается.

Схема алгоритма (алгоритм 2) для определения уровня безопасности индивидуума показана на рис. 6.

В результате выполнения алгоритма 2 уровень безопасности индивидуума в базе данных будет единственным и будет доминировать над уровнем безопасности класса, которому данный индивидуум принадлежит.

В базе данных хранится много индивидуумов, имеющих свой уровень безопасности. С помощью алгоритма 1 могут быть определены уровни безопасности s_y всех классов онтологии C_y, после этого могут быть определены и уровни безопасности s_x всех индивидуумов E_x онтологии с помощью способа описанного языка SWRL, как показано на рис. 7.

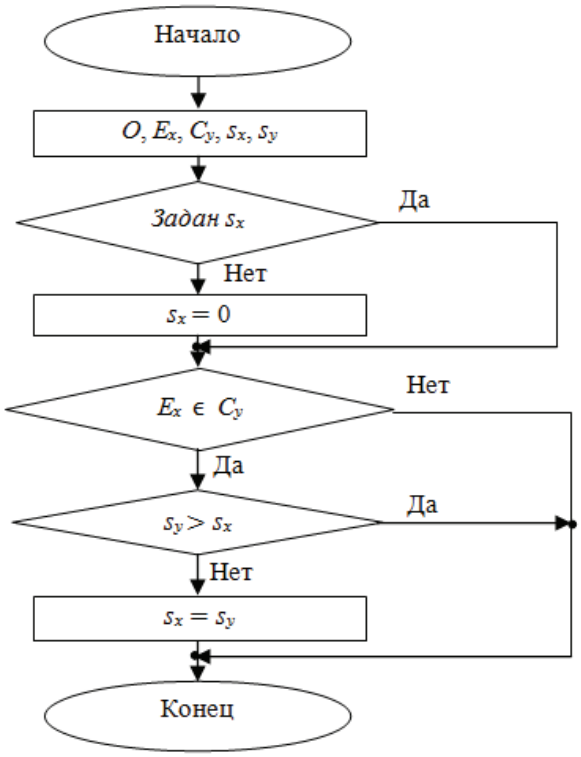

Рис. 6. *Алгоритм для определения уровня безопасности индивидуумов онтологии*

```
(?Ex rdf: type owl: NamedIndividual) ∧
noValue (?Ex ontology: Level ?sx) →
(?Ex ontology: Level 0).
(?Ex ontology: Level ?sx) ∧
(?Ex rdf: type ?Cy) ∧
(?Cy ontology: Level ?sy) ∧
greaterThan (?sy,?sx) → drop (1) ∧
(?Ex ontology: Level ?sy).
```

Рис. 7. *Определение уровней безопасности индивидуумов с помощью языка SWRL*

Уровень безопасности утверждений

Семантические метаданные $M=\{\mu_1,...,\mu_m\}$ – это наборы семантических утверждений (триплетов) μ_i, которые имеют вид $\mu_i=(\alpha,\beta,\gamma)$, где α – это субъект утверждения (понятие, или индивидуум – контекстные метаданные некоторого понятия), β – объект (экземпляр – индивидуум, контекстные метаданные некоторого понятия), γ – отношение между субъектом и объектом. При этом понятия и отношения должны быть определены в онтологии O, а индивидуумы описываются контекстными метаданными онтологической базы знаний. Примерами утверждений являются следующие триады <C,P,C>, <E,P,E>, <C,P,E>, <E,P,V>, <$C,P,NULL$>, <$C,NULL,NULL$>, <$E,P,NULL$>, <$E,NULL,NULL$>, где C – понятие; E – индивидуум понятия; P – связь; A – атрибут; V – значение атрибута (текстовое или числовое) [7]. Для контроля доступа пользователей к утверждению необходимо сравнить их уровни доступа с уровнем безопасности утверждения.

С помощь описанных выше алгоритмов могут быть определены уровни безопасности всех элементов (понятий, связей, атрибутов, индивидуумов) семантической базы данных. В этом случае уровень безопасности s_μ утверждения μ_i определяется как максимальное значение из уровней $\{s_\alpha,s_\beta,s_\gamma\}$, где s_α – это уровень безопасности субъекта, s_β – это уровень безопасности отношения, s_γ – это уровень безопасности объекта.

Алгоритм определения уровней безопасности логических выводов

В онтологии логическим правилом R является выражение, обозначающееся как $\forall x_1,..., x_m$ $(b_1\wedge...\wedge b_k)\rightarrow q$, где $k\geq1$ и $x_1,...,x_m$ – это свободные переменные в $b_1\wedge...\wedge b_k$. Каждый b_i представляет собой утверждение, имеющее вид $[\alpha,\beta,\gamma]$, где α, β, γ – переменные, константы или OWL аксиомы. Левая часть правила $(b_1\wedge...\wedge b_k)$ называется телом, а правая часть правила (q) – головой правила. При составлении таких правил необходимо, чтобы все переменные, включаемые в голову, содержались в составе тела правила.

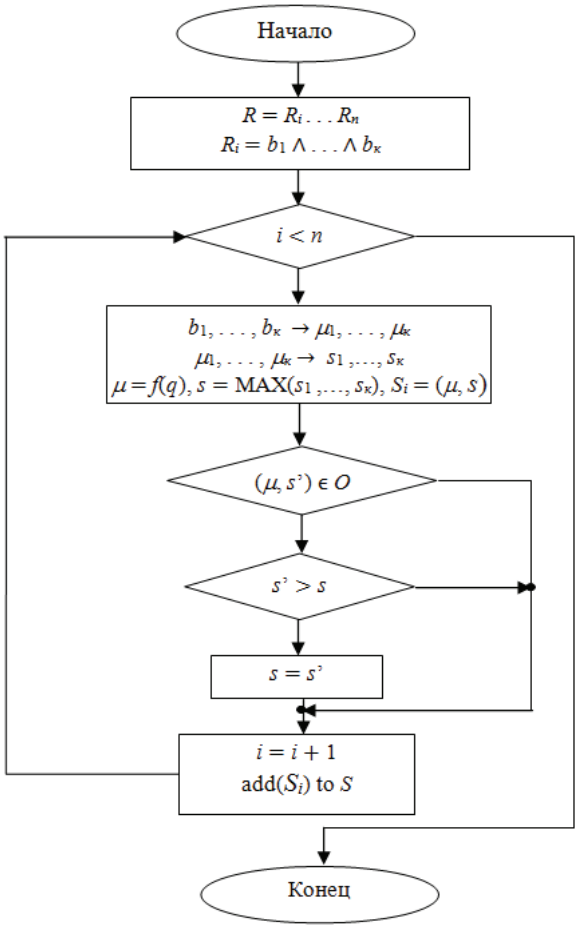

Рис. 8. *Алгоритм определения уровней безопасности логических выводов*

В семантических технологиях f называется функцией отображения только в случае, когда она удовлетворяется следующими условиями:

- f сохраняет все константы (f (*constant*)=*constant*);
- Если $f([x_1y_1,z_1])=[\alpha_1,\beta_1,\gamma_1]$ и $f([x_1,y_2,z_2])=[\alpha_2,\beta_2,\gamma_2]$, то $\alpha_1=\alpha_2$;
- $\forall f(b_i)\in M$, (i=1...k);

Тогда с помощью функции отображения логическое правило R выводит утверждение $\mu_i=[\alpha,\beta,\gamma](f(q)=[\alpha,\beta,\gamma])$.

Алгоритм определения уровней безопасности логических выводов с использованием правил может быть описан следующим образом:

1. Определяется уровень безопасности каждого логического вывода.

   Если существует функция отражения f: $b_1,...,b_k\rightarrow\mu_1,...,\mu_k$, где $f(b_1)=\mu_1,...,f(b_k)=\mu_k$, то:
   - генерируется отражение $\mu=f(q)$ (по определению – существуют логические выводы). После отражения получим утверждение μ, которое должно иметь уровень безопасности s;
   - генерируется значение для уровня безопасности логического вывода с использованием выражения s=MAX($s_\alpha,s_\beta,s_\gamma$);
   - генерируется пара безопасности $S_i=(\mu,s)$ (утверждение, уровень безопасности).
2. Выполняется контроль значений уровней безопасности.

Если в базе данных уже существует утверждение μ, имеющее уровень безопасности s', который большее уровня безопасности логического вывода ($s'>s$), то s присваивается значение s'. В противном случае, т. е. если ($s'<s$), то значение уровня безопасности логического вывода будет равно s.

3. Безопасность (μ,s) добавляется к множеству пар (утверждение, уровень безопасности) безопасности S семантических метаданных M.

На рис. 8 показан алгоритм для определения уровней безопасности логических выводов.

С помощью данного алгоритма, после выполнения логических правил, все уровни безопасности элементов логических выводов определены. В зависимости от своего уровня доступа пользователь может видеть адекватные полученные логические выводы.

Заключение

В работе рассмотрены основные задачи многоуровневой безопасности для семантических данных, которые включают в себя определение уровней безопасности каждого элемента онтологии и логических выводов, выполняемых с помощью правил. Приведены основные принципы задания уровней безопасности для каждого типа элементов онтологии, на основе которых разработаны алгоритмы, использующиеся для определения значения уровней безопасности всех элементов онтологии, а также для каждого утверждения семантических метаданных. Разработан алгоритм определения уровней безопасности элементов новой информации, полученной в результате выполнения логических правил.

СПИСОК ЛИТЕРАТУРЫ

1. Berners T. The Semantic Web // Scientific American. – 2001. – V. 120. – № 3. – P. 220–225.
2. Хоанг Ван Кует. Многоуровневая безопасность для семантических данных // Вестник науки Сибири. – 2012. – № 5 (6). – С. 93–100.
3. Хоанг Ван Кует, Тузовский А.Ф. Контроль логических выводов в семантических данных // Известия Томского политехнического университета. – 2012. – Т. 320. – № 5. – С. 148–151.
4. Тузовский А.Ф., Ямпольский С.В. Системы управления знаниями (методы и технологии) / под общ. ред. В.З. Ямпольского. – Томск: Изд-во НТЛ, 2005. – 260 с.
5. Хоанг Ван Кует, Тузовский А.Ф. Алгоритмы для контроля доступа и модификации семантических данных // Электронные средства и системы управления. – 2012. – Т. 26. – № 2. – С. 41–45.
6. Ian H. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. 2004. URL: http://www.w3.org/Submission/SWRL/ (дата обращения: 21.05.2012).
7. Тузовский А.Ф. Формирование семантических метаданных для объектов системы управления знаниями // Известия Томского политехнического университета. – 2007. – Т. 310. – № 3. – С. 108–112.

Поступила 24.02.2013 г.