Хоанг Ван Кует, аспирант кафедры оптимизации систем управления Института кибернетики ТПУ.
E-mail: student8050@sibmail.com
Область научных интересов: исследование методов поддержки работы с семантическими базами знаний и их эффективности.

УДК 004.415

МНОГОУРОВНЕВАЯ БЕЗОПАСНОСТЬ ДЛЯ СЕМАНТИЧЕСКИХ БАЗ ДАННЫХ

Хоанг Ван Кует

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

Предлагается поход к выполнению многоуровневой безопасности семантических данных. Рассматривается простой алгоритм метода поддержки работы с онтологическими данными. Рассмотрены два вида безопасности семантических данных: произвольная и многоуровневая, которая имеет большую эффективность по сравнению с произвольной безопасностью. Для определения многоуровневой классификации данных использованы характеристики языка описания ресурсов и онтологий. Предложен метод выполнения логических выводов для построения многоуровневой безопасности.

Ключевые слова:
Многоуровневая безопасность, язык описания ресурсов (RDF), язык описания онтологии (OWL), определение политик, доступ к данным, семантические данные, база данных (БД).

Введение

В настоящее время семантические информационные системы (СИС) начинают активно использовать в экономике, науке и политике. СИС обладают способностью анализировать, интегрировать данные и выполнять над ними логические операции.

Одной из важнейших задач при разработке СИС является безопасность работы семантических баз данных. Для её решения необходимо выполнить следующие этапы: создать алгоритм для поддержки безопасности работы с семантическими данными, контролировать доступ к данным, обеспечить безопасность RDF и OWL документов и логический вывод для них.

Одной из самых важных задач для обеспечения многоуровневой безопасности данных является разделение прав доступа пользователей к этим данным и определение уровней политики доступа к каждой части документов, хранимых в семантической базе данных. Эти задачи могут быть решены с помощью многоуровневой безопасности.

Целью данной статьи является описание построения многоуровневой безопасности, обеспечивающей работу с семантическими данными. Для достижения поставленной цели рассматривается произвольная безопасность для базы данных, предлагается общий алгоритм обеспечения многоуровневой безопасности данных, исследуется метод построения многоуровневой безопасности для управления семантическими данными, и приводится логический вывод многоуровневой безопасности онтологических данных.

Произвольная безопасность

Произвольная безопасность касается доступа к данным отдельных пользователей, групп пользователей, а также пользователей, имеющих одинаковые роли. Произвольная безопасность включает в себя 3 компонента: контроль доступа и политику авторизации, политику администрирования, политики идентификацию и аутентификацию [1]. При произвольной безопасности все данные в базе данных будут безопасными, но в некоторых случаях какая-то часть данных может потребоваться для использования в операциях, к которым у пользователей нет доступа. В связи с этим произвольная безопасность не обладает гибкостью.

Для повышения безопасности семантических баз данных требуется обеспечивать ее на всех уровнях прав использования данных пользователями. В этом случае каждый пользователь будет получать собственные права доступа к конкретным частям базы данных. Для этого необходимо осуществлять многоуровневую безопасность данных.

Многоуровневая безопасность базы данных

Многочисленные публикации о многоуровневой безопасности системы управления базами данных были сделаны в 80-х и начала 90-х гг. Развитие таких систем началось с многоуровневых защищенных операционных систем. Оно было в основном связано с многоуровневой безопасностью реляционных баз данных. Затем внимание было обращено к многоуровневой системе управления объектными базами данных и многоуровневым распределенным системам баз данных. Многоуровневое управление безопасностью данных включает в себя многоуровневую политику безопасности, многоуровневую модель данных и многоуровневую систему управления функциями баз данных [2].

На рис. 1 показаны различные права доступа пользователей. Пользователь имеет доступ к данным, но к разным частям данных его права могут быть разными. Например, на первом уровне безопасности пользователь может изменить часть данных, а на более высоком уровне безопасности его права могут быть более ограниченными, он имеет право только на запись или чтение данных (рис. 2).

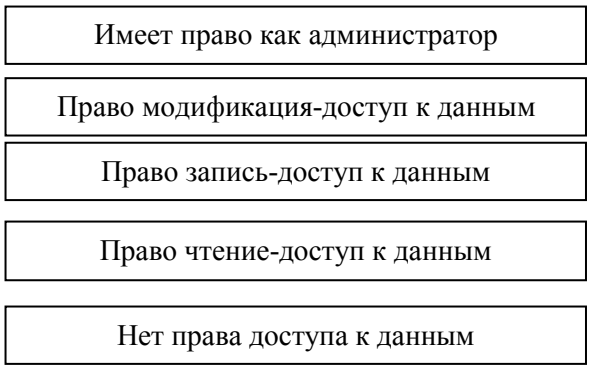

Рис. 1. Права доступа пользователей

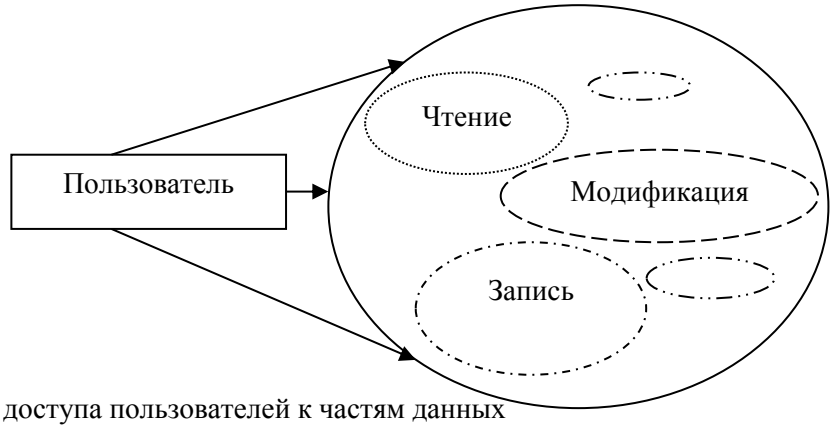

Рис. 2. Права доступа пользователей к частям данных

Доступ пользователей к данным может быть предоставлен в виде уровня их прав и уровня чувствительности данных [3]. Данные могут иметь разные уровни классификации, такие как: неклассифицированные, конфиденциальные, секретные и сверхсекретные. На неклассифицированном уровне все пользователи имеют доступ к данным, но на конфиденциальном уровне права доступа имеют не все пользователи. На секретном и сверхсекретном уровнях доступ имеют только некоторые, специально выделенные пользователи [4]. Пользователи также могут

иметь разные права доступа к данным, находящимся на конфиденциальном, секретном и сверхсекретном уровнях. Они могут читать, писать, изменять данные и выполнять какие-либо операции с данными.

Алгоритм работы подпрограммы обеспечения работы с онтологическими данными

Основные требования по безопасности онтологических данных во многом совпадают с требованиями, предъявляемыми к безопасности реляционных данных: контроль доступа, криптозащита, проверка целостности, протоколирование. Но для безопасности онтологических данных, которые зависят от характеристик и особенностей языков RDF и OWL, необходимы дополнительные требования: определение политики доступа пользователей, многоуровневая безопасность данных, т. е. определение существования данных, определение классификации политик доступа к частям данных, определение прав доступа пользователей к данным.

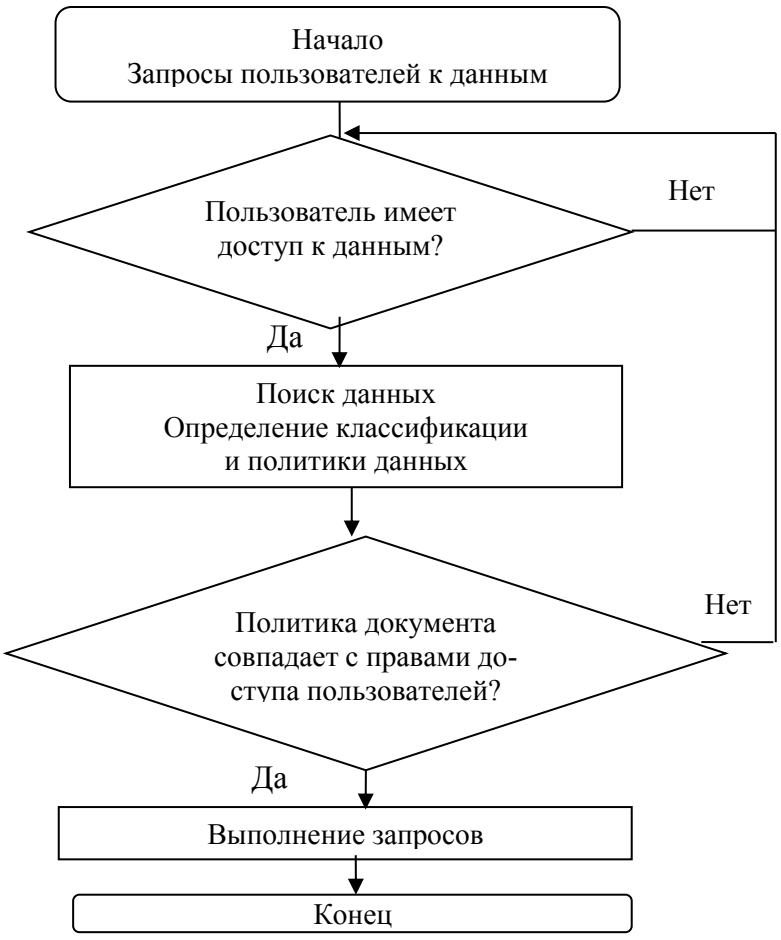

Рис. 3. Алгоритм для обеспечения работы с онтологическими данными

На рис. 3 приведён алгоритм работы подпрограммы поддержки работы с онтологическими данными с помощью использования многоуровневой безопасности. В соответствии с этим алгоритмом только авторизованный пользователь имеет права доступа к онтологическим данным. В зависимости от должности пользователей их право доступа к данным может быть разным: одни могут только читать данные, а другие – изменять, добавлять и удалять данные.

Если пользователь отправляет запросы к данным, имея право доступа к ним, то подпрограмма будет выполнять такие последовательные операции, как: поиск данных, определение классификации политик данных. Если у пользователя отсутствует право доступа к данным, то

он не может отправлять запросы к данным. После выполнения последовательных операций подпрограмма проверяет соответствие между классификациями политик частей данных и прав пользователей на чтение, изменение данных. Если они совпадают, то запрос выполняется, если нет, то пользователь должен отправить другой запрос к данным.

Многоуровневая безопасность эффективнее, чем произвольная, так как она позволяет определять права пользователей по доступу к различным частям БД, вплоть до конкретного элемента. При этом имеется возможность не только предоставить доступ тому или иному пользователю, но и указать разрешенный тип доступа: что именно может делать конкретный пользователь с конкретными данными (читать, модифицировать, удалять и т. д.).

Многоуровневая безопасность RDF и OWL документов

Основными компонентами семантической базы данных являются RDF и OWL документы [5]. Для выполнения многоуровневой безопасности на семантических данных необходимо осуществить безопасность на разных уровнях RDF и OWL документов.

Язык RDF является основным языком описания семантических данных. Он позволяет описать содержание документов и отношение между различными их разделами [6]. Использование языка RDF обеспечивает улучшение взаимодействия между данными, повышает качество выполнения поиска и категоризации данных. С помощью языка RDF можно указать политику доступа пользователей к каждой части данных по уровням их классификации, которые приведены на рис. 4.

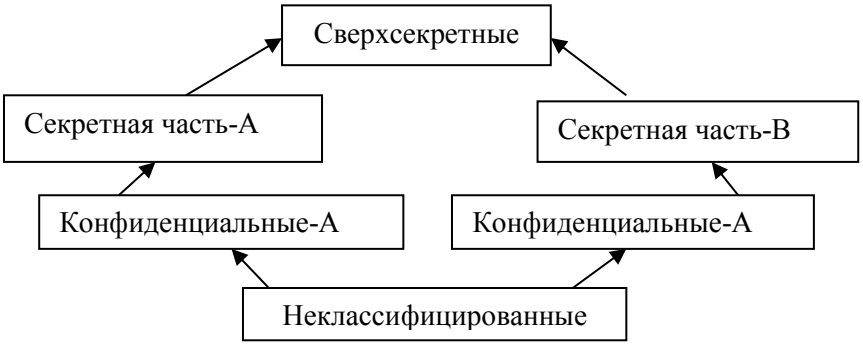

Рис. 4. Уровни классификации данных

Для выполнения многоуровневой безопасности доступа к RDF-документам необходимо решить следующие задачи:

1. Классифицировать RDF-данные во всех документах или только в некоторых частях [7].
2. Определить политику безопасности для прав доступа пользователей к RDF-документам и их частям.
3. Применить принципы обеспечения безопасности реляционных данных к RDF-данным.
4. Применить политики безопасности в схеме RDF-данных.
5. Исполнить правила в отношении содержания, контекста и динамической безопасности.
6. Обеспечить многоуровневую безопасность к запросам на изменения и процесса обмена данными для баз RDF-данных [8].
7. Обеспечить логический результат, получаемый пользователем на выходе, совершенно точным и обладающим полной структурой RDF-представления: субъект–свойства–объекты.

Пример на рис. 5 показывает, как используется RDF-язык для определения многоуровневой политики. В этом примере имя `Director` находится в секретном уровне, его адрес находится в конфиденциальном уровне, а информация о зарплате является неклассифицированной.

```
<rdf:RDF ….>
<company:Director rdf:ID=” 1234”>
      < company:name>Антон</company:name>
            Level = Secret
< company:address>Russia, Tomsk, Usova </company:address>
            Level = Confidential
      < company:salary>2000 $</company:salary>
            Level = Unclassified
</company:Director>
</rdf:RDF>
```

Рис. 5. Указание политики для фрагмента RDF-данных

Язык OWL является более выразительным и имеет возможность выполнения логического вывода больше, чем RDF-язык**.** Для выполнения многоуровневой безопасности OWL-документов необходимо решить все вышеперечисленные задачи для обеспечения безопасности RDF-документов, а также использовать:

1. Особенности RDF-схемы в OWL-языке для определения многоуровневой политики. Например, элементы: `rdfs:domain, rdfs:range` можно использовать для определения домена и диапазона данных, к которым пользователи имеют доступ.
2. Характеристики свойств OWL-языка для определения прав доступа пользователей к данным, а также классификацию уровней политики данных. Например, для указания прав доступа пользователей можно использовать: симметричное свойство (`symmetricProperty`), обратно-функциональное свойство (`inverseFunctionalProperty`), транзитивное свойство (`transitiveProperty`).
3. Ограничения свойств в языке OWL для определения права доступа пользователей к разным частям данных, например: `allValuesFrom, someValuesFrom`.
4. Ограниченные кардинальности в языке OWL для указания количества классов пользователей, имеющих доступ к конкретным данным, например: `maxCardinality, cardinality minCardinality`. Если минимальная кардинальность `minCardinality=1`, то любой представитель этого класса будет связан по этому свойству по крайней мере с одним индивидом пользователей.

Пример рис. 6 показывает, как политика была определена в OWL-документах. При использовании особенности RDF-схемы `rdfs:range, rdfs:domain` была определена политика, указывающая на то, что только пользователи в домене отдела директоров `DirectorDepartment` имеют доступ к данным, находящимся в диапазоне документов `Document`.

```
<owl:Ontology rdf:about=”Director”/>
      <owl:Class rdf:ID=”Director”>
      <rdfs:subClassOf rdf:resource=”#DirectorDepartment”/>
Level = L1
</owl:Class>
<owl:ObjectProperty rdf:ID=”canAccessTo”>
      <rdfs:domain rdf:resource=”#DirectorDepartment”/>
      <rdfs:range rdf:resource=”#Document”/>
      <rdfs:subPropertyOf rdf:resource=”#Access”/>
Level = L2
</owl:ObjectProperty>
```

Рис. 6. Использование языка OWL для определения политики доступа пользователей к данным

Использование логических выводов для построения многоуровневой безопасности

Язык OWL основан на языке RDF, все логические отношения на RDF могут быть выполнены на OWL, но некоторые сложные логические выводы ограничены в OWL. Например, с помощью языка OWL может выполняться только такая логическая операция: если X – это человек, то X – это мужчина или женщина. А такая логическая операция не может выполняться с помощью языка OWL: если X – это человек, и X – не мужчина, то X – это женщина. Для решения недостатка по логическому выводу OWL-языка был создан немонотонный типизированный язык разметки (no monotonic typed markup language-NTML) [9]. С помощью NTML все типы логического вывода, представленные в OWL документах, могут быть выполнены. Например, если Антон является руководителем Олега, то Антон является администратором Олега. Этот тип логического вывода имеет общую форму: B1, B2, …, Bn -> A, где Bn является членом логического вывода A. Другой пример: Олег является заведующим отделом, если его зарплата больше, чем 20000 руб., и он является администратором. Это выражается таким образом:

```
Leader(Олег).
Salary(Олег, Y),Y > 20000 руб. -> NOT Leader(Олег).
NOT admin(Олег) -> NOT Ledaer (X).
```

Благодаря функциональной особенности языка NTML доступ к объектам может быть классифицирован и распределён при использовании базы данных. Например, если в техническом отделе работает Антон, имеющий зарплату больше, чем 21000 руб. и являющийся администратором, то отсюда следует, что Антон является главным человеком в этом отделе: Department(Антон, Technical), Salary(Антон, 21000 руб.), admin(Антон) => LeaderOfTechnicalDepartment.

Пользователь получает доступ к чтению каких-то данных только при условии, если он является администратором компании. Тогда описание на языке NTML будет выглядеть следующим образом: Administrator (X, Company) => Read-access (X,R), где R – это данные, классифицированные по правилу безопасности. Для указания того, что данные находятся на конфиденциальном уровне, необходимо записать: Level(R1, confidential).

С помощью NTML правила можно классифицировать политику для различных частей на разных уровнях следующим образом: Level (->| Salary (Олег, 20000 руб.), unclassified). Level (->|Salary(Олег,30000 руб.), secret). С помощью языка NTML право доступа пользователя к данным будет указано так: administrator (Олег, company) -> NOT(Read-access(Олег,Y)). Salary (Y) -> Read-access (Олег,Y), а для классификации секретного уровня безопасности рабочих, имеющих зарплату больше, чем 20000 руб., таким образом: EMP(name, salary, department) and Y>20000 руб., → Level(EMP) = Secret. Или для указания общих имени и зарплаты в таком же уровне следующим образом: EMP(name, salary, department) → Level(TOGETHER(name, salary) = Secret.

Выводы

Многоуровневая безопасность для семантических данных играет важную роль в процессе управления семантическими данными информационных систем. Она позволяет проверить права доступа каждого пользователя и четко определить, какие части данных могут быть доступны разным группам пользователей. Пользователи могут читать, заносить и изменять данные. Одни и те же данные могут являться доступными для одних пользователей, но секретными для других. Благодаря разделению данных на разных уровнях доступа степень обеспечения данных увеличивается, а это означает, что данные становятся более безопасными. Основными элементами семантических баз данных являются RDF- и OWL-данные, поэтому для решения задачи построения многоуровневой безопасности семантических данных необходимо создать многоуровневую систему безопасности для RDF и OWL документов.

В данной статье определены основные задачи для решения поставленной проблемы, показан способ классификации уровней доступа к частям документа и метод использования языков RDF и OWL для определения прав доступа пользователей. Приведены примеры по использованию RDF и OWL для создания политик доступа пользователей. Рассмотрен метод логического вывода при обеспечении многоуровневой безопасности онтологических данных, правильность которого зависит от метода использования OWL- и RDF-языков для решения логического отношения между данными.

СПИСОК ЛИТЕРАТУРЫ

1. Thuraisingham B. Secure Sematic web Services. – Texas: Department of Computer Science, 2007. – 123 p.
2. Thuraisingham B. Database and Applications Security: Integrating Data Management and Information Security. – Texas: Department of Computer Science, 2005. – 276 p.
3. Bertino E. Access Control for XML Documents, Data and Knowledge Engineering // Journal of Artificial Intelligence Research. – 2002. – V. 120. – № 5. – P. 237–260.
4. Sandhu R., Coyne E.J., Youman C. Role-based access control models // Journal of Artificial Intelligence Research. – 1996. – V. 68. – № 3. – P. 38–47.
5. Allemang D. Semantic Web for the working ontologist: effective modeling in RDFS and OWL. – Boston: DK 9000 Aalborg, 2011. – 150 p.
6. Anton B. Resource Description Framework // Technical report, Department of Mathematics and Computer Science, 2006. – V. 20. – № 3. – P. 157–168.
7. Thuraisingham B. Security for the semantic web // International Journal of Approximate Reasoning, 2005. – V. 11. – № 1. – P. 257–268.
8. Reddivari P., Finin T., Joshi A. Policy based Access Control for a RDF Store // Artificial Intelligence: Proceedings of the XXth conference. – Canada, 2005. – P. 78–83.
9. Zhang N.L. Exploiting causal independence in bayesian network inference // Journal of Artificial Intelligence Research. – 1996. – V. 301. – № 5. – P. 301–328.

Поступила 28.09.2012 г.