ANALYSIS OF ACCESS CONTROL MECHANISMS FOR SPATIAL DATABASE

Jiayuan LIN[a, *], Yu FANG[a], Bin CHEN[a], Pengfei WU[a]

[a]Institute of Remote Sensing and Geographic Information System, Peking University
Beijing, P.R.China, 100871

WG VIII/5

KEY WORDS: Spatial Database, Access Control, Authorization, SDE, ORDBMS, View

ABSTRACT:

The development and application of GIS ultimately creates the demand for spatial information security, and access control is our primary concern. Spatial information is mostly stored and managed by database technologies, and SDE/RDBMS and ORDBMS are the most popular two. Correspondingly, there are two possible solutions for restricting access to spatial information in database: the SDE-based access control mechanism and the view-based access control mechanism. In the SDE-based mechanism, we have to modify the source code of SDE and alter the schemas of spatial tables to realize access control functionalities. In the view-based mechanism, we use the view techniques to define various views to meet different security requirements, and then authorizing privileges to views. The basic components of an access control system mainly contain user authentication, authorization rule repository, PEP, PDP, and PIP, which are distributed and implemented differently in the two mechanisms. As the view-based mechanism is much cheaper and flexible, it is the better choice for individual users or small companies to add access control functionality in their spatial applications.

1. INTRODUCTION

Geographic Information Systems (GIS) are becoming more and more widely used in a variety of application domains, creating the need for multi-user support, sophisticated data management, global connectivity, and information security (Sahadeb, 2003). Spatial information in some application domains, for example government agencies, is usually classified according to levels of security, which can only be accessed by corresponding security level users. However, spatial information security is little addressed in most professional literatures of spatial fields, either GIS or spatial database.

Spatial Information security usually contains three aspects: confidentiality, integrity, and access control (Oh, 2004). Confidentiality means that only legal users can get understandable spatial information while spatial data are being transported on the network, even though intercepted by illegal user. As usual, it is achieved with cryptography. Integrity means spatial information cannot be tampered (added, deleted, or altered) by illegal users and is usually reached with digital digest and signature. Access control has two meanings: authentication and authorization (Sandhu, 1994). Spatial information system should firstly ensure log-in users are legal users of the system, and then legal users are executing permitted operations on spatial objects of interest. Authentication is the prerequisite of authorization. Confidentiality and integrity of spatial information have already had satisfactory solutions in IT domain, but access control has not taken geometric properties of spatial data into consideration and we have to make our efforts to find appropriate solutions which meet special access control requirements in spatial domain.

So far, spatial data used in GIS are mostly stored and managed with database technology. Due to its complex and unstructured characteristics, spatial data cannot be directly stored in RDBMS. Spatial data Engine (SDE) technology is developed to realize integrated management of both non-spatial and spatial properties of spatial data in RDBMS (Shekhar, 2003). Thus access control to spatial data can also be accomplished by SDE. With the advent of ORDBMS technology, spatial objects can be stored and managed by database without the help of SDE. Therefore, access control to spatial data becomes simpler and easier to implement.

The rest of this paper is organized as follows. In Section 2, we describe the authorization model for spatial data. In Section 3, we discuss the framework of a general access control system. In Section 4, we introduce the SDE-based access control mechanism and analyze its complexity of implementation. Then, In Section 5, we propose view-based access control mechanism for ORDBMS and its implementation details. Advantages and disadvantages of SDE-based mechanism and view-based mechanism are also listed.

2. AUTHORIZATION MODEL AND SPATIAL CONSTRAIT

It is a remarkable advancement to use database technology to manage both descriptive properties and geometric property of geospatial data, whether with the mediation of SDE or not. Such a system has many advantages for geospatial data management inheriting from database technology, such as concurrent access control, transaction management, security

* linjiayuan@gmail.com; phone 86-10-62769285-803;

management, and so on. As for security management, general authorization mechanisms provided by database cannot fully meet the demands of geospatial application. Since SDE/RDBMS and ORDBMS are two solutions for geospatial database, there are correspondingly two geospatial access control mechanisms. We will analyze their characteristics and differences in detail in Section 4 and Section 5. Here, we shall discuss the authorization model, which is used to define authorization rules in the access control system.

The authorization model is a triple <S,O,P>, whose elements represent subjects, objects, and privileges, respectively (Ferraiolo, 2003). It establishes which subjects are authorized to perform which operations (privileges) on which objects.

Subjects usually denote users or processes which are executing on behalf of users. In database users includes administrators and ordinary users. From the perspective of multilevel secure systems, subjects can be categorized into three levels: confidential, secret, or top-secret (Ferraiolo, 2003).

Objects have two equivalent representation schemes. One is using terms of database, namely relations (tables), tuples (records) and fields. The other is using terms of geospatial domain, namely map layer, geospatial objects, geometric or descriptive properties. We should also note the hierarchical relationship among these objects. Like subjects, every type of objects has one of the three clearance levels: confidential, secret, or top-secret.

Privileges, also called permissions, denote what operations certain subjects can execute on specific objects. In terms of database, operations include **SELECT**, **INSERT**, **DELETE**, **UPDATE**, while geospatial domain contains query, edit of geospatial objects, etc. To simplify the discussion, this paper in the following sections primarily takes into account SELECT or query functionality for geospatial objects.

In principle, subjects can access objects only if security level of subjects equal or excel that of objects. When making access decision against predefined authorization rules, besides security levels, it should also count in spatial constraints. Thus, the <S,O,P> is extended to <S,O,P,C>, and C represents spatial constraints, which distinguishes spatial access control from no-spatial.

3. GENERAL ACCESS CONTROL FRAMEWORK

The framework of a typical access control system is showed in Figure 1, which is mainly composed of the authentication module, the authorization rule repository, PEP, PDP, PIP and the protected data or resources (Sandhu, 1994).

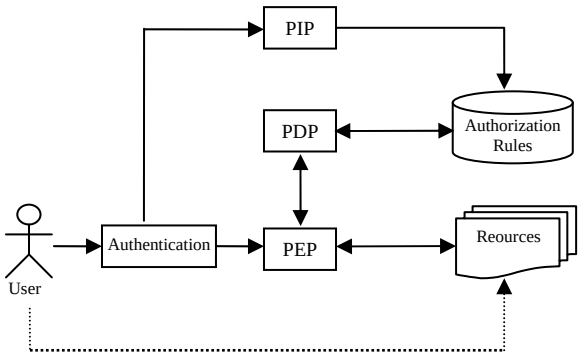

Figure 1. the framework of general access control system

The data or resources are the targets users intend to access, which will be protected by the access control system. The authorization rules defined according to the <S,O,P> model shall be stored in a repository. PIP, short for policy information point, is responsible for creating and modifying the authorization rules in the rule repository.

Before issuing access requests for target resource, users must firstly be authenticated to be legal users by the authentication module. PEP, short for policy enforcement point, is in charge of translating original access requests from users into the format conforming to that of authorization rules. PDP, short for policy decision point, is used to retrieve related authorization rules from the repository, match them with access requests, and then make access decisions (allow or deny) and forward the decisions to PEP. PEP translates the decisions from PDP into the format readable for users, and then returns the decisions to users.

4. SDE-BASED ACCESS CONTROL MECHANISM

The basic functionality of SDE is to manage unstructured spatial data in structured RDBMS; therefore, it is an ideal place to make access decisions on spatial data stored in database with respect to spatial semantics.

In database, all geospatial objects in the same map layer are stored in a table. Each geospatial object is represented by a record of the table. The geometric property of a geospatial object is stored as a field of the record, which is usually of the binary type BLOB, to implement integrated management of spatial and non-spatial information (Gan, 2003). SDE acts as the mediator which is responsible for interpreting from binary spatial data to geometric meanings or vice versa. In addition, SDE also provides Boolean functions to evaluate topological relationships between spatial objects, such as *Within*, *Contains*, etc; and spatial analysis functions, such as *Overlay*, *Buffer*, etc to generate new spatial data.

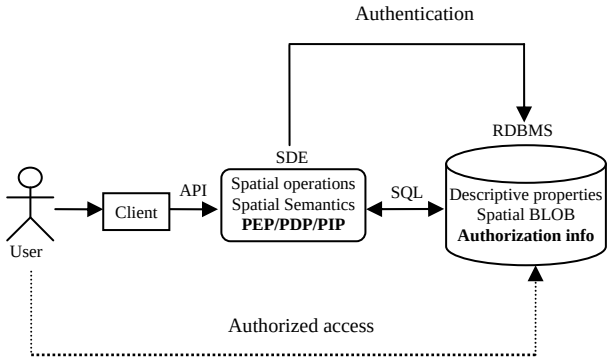

Figure 2. SDE-based access control mechanism for spatial database

In SDE-based access control system for spatial database (see Figure 2), the protected resources are tables whose records represent spatial objects; the functionalities of PEP, PDP, and PIP are undertaken by SDE; authorization information are stored in the same database as spatial data.

The procedure of GIS clients to access spatial database is also illustrated in Figure 2. SDE does not have independent user authentication functionality, and it completely relies on the authentication mechanism offered by database. SDE as spatial middleware resides together with RDBMS on the server side and provides APIs to interact with both client and RDBMS. The client revokes the client API to send its access requests for spatial data to SDE. Upon receiving access requests from the client, SDE translates client requests into standard SQL and revokes RDBMS API to communicate with RDBMS to retrieve requested spatial BLOB data. Then SDE transforms the BLOB binary data into spatial semantic data, and then returns them to the GIS client generating the original access request.

4.1 Spatial data organization

SDE usually uses layers to store features (spatial objects) and each layer contains only one type of features: point, line or polygon. SDE logically partitions layers into grids which are used to locate each individual feature.

SDE creates layer tables for all map layers to index features to speed up spatial queries. Each layer is composed of business table, feature table, spatial index table, and point table. All of the tables are linked with feature ID (Gan, 2003).

Each record in business table represents a feature and stores attribute properties of the feature. Feature table stores shape types and boundary boxes of features in feature tables. Spatial index table contains information of the grid unit and boundary boxes of features. Point table stores coordinate values of each shape in a binary type of BLOB, which is translated into spatial meanings by SDE.

4.2 SDE-based access control

As described in last subsection of spatial data organization, to implement access control for geospatial objects, we have to extend the functionalities of SDE and modify those tables in spatial database.

As for authentication, SDE relies on the mechanism provided by RDBMS, namely user information is stored in database and RDBMS is in charge of authenticating users. SDE merely acts as a mediator. Spatial authorization must alter schemas of related tables to store authorization information (legal users and corresponding privileges) according to granularities of control. For map layers, the schema of layer tables is added fields: user and privilege, which will be filled when creating feature layers according to their specific authorization requirements. For features, the similar modification will be made to the schema of business tables, as each record of business tables stores properties of a single feature. Likewise, authorization information for features should be filled when loading spatial data into business tables of spatial database. As for spatial context, for example features in a rectangular window of certain privilege, the authorization information is filled in feature tables on the fly. Those features falling in the window are calculated with the window rectangle and the boundary boxes stored in the feature table.

While application clients make access requests for spatial data through SDE, SDE will firstly certificate their IDs. If legal users, SDE reads authorization information of intended map layer, and then compared legal users and privileges from layer tables with that of users and intended operations to decide whether authorizing access to the map layer, or just rejecting. After gaining the access permission of the map layer, the similar procedure will be made to achieve access permission to specific features.

4.3 Evaluation

As discussed above, the implementation of SDE-base authorization is very complex and overhead.

- Maintenance of authorization information is overhead and authorization policies are lack of flexibility. Schemas of map layer table and feature table have to be extended to contain authorization information. The fill and update of authorization information is also very cumbersome, especially taking into account access control to spatial context. On the other hand, if we plan to extend authorization to the granularity of fields, more modifications to table schemas have to be made and update overhead amounted. It will gradually become a mission impossible.

- Currently, most SDE products have not provided spatial access control mechanisms. To add spatial authorization functionality, source codes of SDE must be modified to deal with authorization decisions and maintain authorization information. As most SDE software is proprietary, such a task can only be done by those big SDE vendors. Individuals or small GIS companies, who do not have access to source codes of SDE, cannot improve their applications with such security functionality.

5. VIEW-BASED ACCESS CONTROL MECHANISM

ORDBMS-based spatial database is extended with the support for spatial object types and relevant spatial operation functions. As an example, PostGIS is such an extension for PostgreSQL (PostGIS, 2007). Those obstacles confronted with SDE-based access control in RDBMS can be subtly got over by view-based access control in ORDBMS. The view-based mechanism typically has four components: database accounts, database login (authentication), privileges, and views (see Figure 3)

(Liang, 2005). First of all, we should create some views (View1, View2 …) from the same base table in spatial database according to different security requirements and restrictions, whatever spatial or non-spatial, and then grant them to different security level database accounts with different access privileges. After a user is authenticated by spatial database, he gains the database account's privilege to the specified view, which is actually a subset of the whole dataset.

Views are virtual tables composed of rows and columns, which look much like real tables of the database (Bruce, 1996). When executing a query, the result records retrieved from a view are not entities actually stored in the form of a table with the schema and record number of the view, but dataset of the base table from which the view creates. Views can be established from one or multiple base tables, or even other views. Therefore, view-based access control mechanism is fairly flexible and extensible.

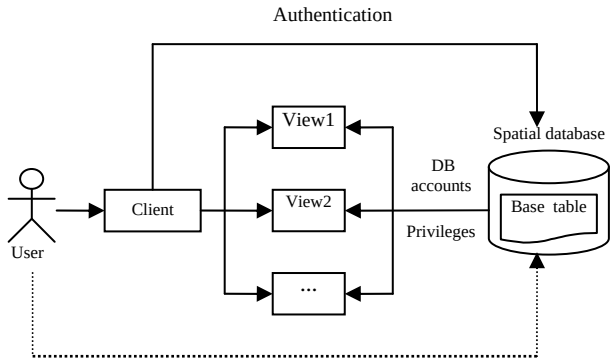

Figure 3. View-based access control mechanism for spatial database

As showed in Figure 3, the components and control flow for the view-based access control mechanism is greatly simplified in comparison to a general access control mechanism. The functionality of PEP in view-based access control mechanism is a little changed. The other necessary components, such as user authentication, PDP, the authorization rule repository, and PIP are all embedded in spatial database, and their functionalities are automatically accomplished by database.

5.1 Use views to restrict access

By creating views, we can easily implement table level (map layer level), record level (feature level), field level (property level) or even spatial context authorization, which is much more difficult for SDE-based mechanism. Since the approach for users to access views is identical to that of base tables, users usually do not perceive the differences and will not try visiting base tables. In order to secure base tables, we had better use different names or even grant different access rights between views and base tables.

5.1.1 Restriction to properties

With respect to security, we usually want to let specific users access some columns of base tables while hiding other sensitive ones. We can reach this objective by creating a view which only contains fields accessible to users. The Syntax is as follows.

CREATE VIEW <view_name> AS SELECT [(<field_name> [, <field_name>] …)] FROM <base table name or name list or other view names>

For example, there is a map layer of administrative districts of a given region. The schema of the base table to store the map features has the following fields: district_name, population, GDP, and the_geom (spatial property). The population of each district is accessible for the public, but the GDP inaccessible. To hide the GDP information from the public, we can create a view which does not has the GDP column, and then authorize public users the access right to the view. Public users can execute SQL-based query on the view or just utilizing information arrow on the toolbar to click on the map to get population information of an individual district, but they have no way to get GDP information of the district.

5.1.2 Restriction to spatial objects

As each record in base tables represents one spatial object, including its spatial (the geometric field) and non-spatial properties. When we need to restrict access to some spatial objects, we can use the **WHERE** clause to create a view which contains records corresponding to the intended spatial objects, but shields those unwanted. The Syntax is as follows.

CREATE VIEW <view_name> AS SELECT <field_name list> FROM <table_name list> WHERE <spatial or/and non-spatial relationship expressions>

The conditions used to filter records can either the spatial field or non-spatial ones as spatial relationship judgement functions are supported in the **WHERE** clause in ORDBMS. As an example, given a North America map, each feature represents one state or province of the countries in the continent. We plan to allow users from U.S. or Canada only to access the part of the map belonging to their countries, respectively. To achieve this aim, we create two views based on the country field of each record whether U.S. or Canada, and then grant such two views to two different database accounts. When one person is accessing the map over the Internet, the server assigns him the database account according to his IP address and then he get the access to the view of his country. In another case, we only permit those states or provinces adjacent to Five-Lake Region to be accessible for certain purpose. We can create such a view with *Intersects* function in **WHERE** clause to judge the intersect relationship between spatial objects of the map and Five-Lake Region. The subsequent procedure to implement access control to this view is similar to last case.

5.2 Privilege authorization

After views satisfying security level requirements are created, the next thing is to grant access rights for specific users to intended views. The procedure typically includes two steps.

- The first step is to prohibit users from directly accessing base tables. To achieve this objective, we must assure that the owners of the base tables, usually also the creators of views based on those base tables, cannot be used as user accounts exposed, as they by default have full privileges on the base tables they own. Simultaneously, you also cannot explicitly grant access rights to view access users or **PUBLIC**, which is likely to break the separation between base tables and views.
-

- The second step is to grant access rights to each user, depending on its security level, for corresponding views with SQL command **GRANT**. In addition to establishing mapping between user accounts and views, we must also grant specified access privileges to users. In principle, we cannot modify base tables through their views (PostgreSQL, 2007). Therefore, as for such privileges as **INSERT**, **UPDATE**, and **DELETE**, there is an additional step to create **RULE**s (**ON DELETE DO INSTEAD**, etc) to implement their modification functionality, respectively.

The syntax of granting privilege to users on specific views is as follows:

GRANT {SELECT|INSERT|UPDATE|DELETE|…} ON view_name to user_name

When we want to cancel from users some access privileges to views, the syntax is similar to above, except using **REVOKE** to replace **GRANT**. Typically, there is a set of spatial operation functions, such as Distance for computing distance between two points whatever, applied to views with spatial elements. If we want to prohibit certain functions from operating on views for some purpose, we can explicitly revoke them. With the numbers of users and views increasing, it is a good practice to utilize user **GROUP**s to simplify the management of access rights, as users in the same group have the same access privileges to the corresponding views.

5.3 Evaluation

Compared with SDE-based access control mechanism discussed in Section 4, view-based mechanism has the following advantages:

- While creating views from a table according to security levels, it does not duplicate the dataset, and thus saves a large amount of storage space, especially in the case of huge-volume spatial data. At the same time, due to operations on the same dataset, inconsistence between multiple copies is avoided.

- Those base tables referred by views can be in either local database or remote database. While combining schema alike spatial data from databases located at different places, distributed spatial queries can be used to define views.

- By combination of the two kinds of view restrictions discussed in Section 5.1, non-spatial and spatial access restrictions can be seamlessly integrated into the definition of a single view.

- There is no need for view-based mechanism to alter the schemas of base tables or add extra tables to store authorization information. It primarily exploits built-in functionalities of spatial database. Therefore, personal users and small companies can easily add spatial access control functionality to their spatial applications.

View-based mechanism may have the problem of concurrent control, which has, to some extent, been solved by the internal mechanism of database. In addition, the abusive usage of views can result in redundancy and unnecessary overhead of the access control predicates, and the potential information leakage through exceptions and errors caused by user-defined functions (Kabra, 2006).

6. CONCLUSION AND FUTURE WORK

Nowadays, most spatial data used in GIS are stored and managed with the database technology. SDE/RDBMS and ORDBMS are the most popular two modes to realize spatial database. To meet the incremental requirements for securely accessing spatial information stored in spatial database, we discuss SDE-based access control mechanism for SDE/RDBMS and view-based access control mechanism for ORDBMS. The implementation of SDE-based access control mechanism is complicated and overhead, while the view-based access control mechanism is much cheaper and flexible. The view-based mechanism can easily enforce spatial and non-spatial access control, and even realize federated access control for multiple tables or distributed databases. Therefore, the view-based mechanism is the better choice for individual users or small companies to add access control functionality in their spatial applications.

As it becomes a tendency to build GIS applications based on geo web services conforming to the specifications of OGC, our future work is to find out the appropriate access control mechanisms for spatial information and functions provided in the form of geo web services.

ACKNOWLEDGEMENTS

This work is part of Project 40501052, supported by NSFC, the National Science Foundation of China. It is also supported by Project 2006AA12Z201, sponsored by NHTRDPC, the National High Technology Research and Development Program of China.

REFERENCES

Bruce G, 1996. *Security in Distrbuted Computing*. New Jersey: Prentice Hall, pp. 125-127

Ferraiolo D F, 2003. *Role-based Access Control.* Boston: Artech House, pp. 213-215

Gan Q, 2003. Realization of Total-relationship Spatial Database Based on SDE. *Surveying and Mapping of Sichuan*. 26(02), pp. 59-61

Kabra G, 2006. Redundancy and Information Leakage in Fine-Grained Access Control. *ACM SIGMOD 2006*. New York: ACM, pp. 133-144.

Liang, 2005. Study on view-based security model for database. *Sun Yatsen University Forum*, 2005 (03), pp. 134-137.

Oh Y H, 2004. MLS/SDM: Multi-level Secure Spatial Data Model. *Computational Science and Its Applications – ICCSA*. Berlin: Springer, pp. 222-229.

Sahadeb De, 2003. Secure Access Control in a Multi-user Geodatabase. http://gis.esri.com/library/userconf/proc02/pap0355/p0355.htm(accessed 5 Oct. 2007)

Sandhu R S, 1994. Access Control: Principles and Practice. *IEEE Communications Magazine*,1994, (9), pp. 40-48.

Sasaokal L K, 2006. Access Control in Geographic Databases. *Advances in Conceptual Modeling - Theory and Practice*. Berlin: Springer, pp. 110-119.

Shekhar S, 2003. Spatial Databases: A Tour. New Jersey : Prentice Hall.

PostGIS, 2007. PostGIS official website. http://postgis.refractions.net (accessed 20 Oct. 2007)

PostgreSQL, 2007. PostgreSQL 8.1.11 Online Documentation. http://www.postgresql.org/docs/8.1/static/ (accessed 24 Oct. 2007)