A Controlled Access to Spatial Data on Web

Elisa Bertino, Maria Luisa Damiani
Dipartimento di Informatica e Comunicazione, Universita' degli Studi di Milano
Via Comelico 39, 20135 Milano (Italy)
{bertino,damiani}@dico.unimi.it

SUMMARY

The proliferation of Geographical Information services on the Web is creating unprecedented opportunities for the shared use of geographical information and at the same time an increasing need of controlled access to spatial data. In the paper we present an approach to the definition of an access control system for spatial data on the Web. The goal is to provide a mechanism for controlling what spatial data can be accessed by whom, for doing what and where. For that purpose, we propose an access control model based on a spatially extended notion of authorization rule. An architectural framework for enclosing such a model into a Geographical Information service has been developed and applied to a case study.

KEYWORDS: *Web Services, Geographical Information Services, Spatial Data Security, Access Control Systems*

INTRODUCTION

Motivations

The proliferation of Geographical Information services on the Web (Spatial Web services for short) is creating unprecedented opportunities for the dissemination and shared use of geographical information.

In general, Spatial Web services are software objects performing geo-processing functions, such as map visualization, conversion of coordinate systems and spatial analysis, that are invoked by Web clients through a programmatic interface. Spatial Web Services are important because they represent a viable alternative to the traditional monolithic, complex and often under-utilized GIS systems. Moreover, Spatial Web Services are increasingly used for the development of modern Spatial Data Infrastructures (INSPIRE, 2002).

An issue, however, that has not been much explored by the GIS community is how to ensure secure access to spatial data on Web. Indeed, there is a strong need for data security, motivated by several factors: first, geographical data may contain sensitive information, so that data cannot be freely disclosed; moreover, the users of Spatial Web services, such as public administrations, urban planners, surveyors and professionals, because of their different roles and expertise, need to be assigned different rights for operating on data; a controlled access to corporate and government data is also of utmost importance for the development of Spatial Data Infrastructures. Likewise data providers need to protect the resources they publish on the Web.

The data security issue has thus several facets that can be analysed from multiple perspectives. In this paper we look at the data security problem from the data management technology point of view. In general, two classes of services are recognized as crucial for a secure Internet infrastructure: access control services and communication security services. Access control services protect Internet resources from unauthorized use, whereas communication security services protect data transmitted over the network (Joshi, 2001).

In this paper we propose an access control mechanism to ensure confidentiality and integrity of spatial data on the Web (Bertino, 2004). Ensuring confidentiality means preventing improper disclosure of information to users that are not authorized to access it. Ensuring integrity means to protect data from unauthorized modifications.

As a motivating scenario, we consider the case of an environmental database located on some server that is accessed and cooperatively populated by surveyors through a Spatial Web service. The database contains different kinds of spatial objects such as illegal waste deposits and unauthorized construction sites in protected areas. The users, who indirectly access the service through a client computer, may perform different operations, depending on the role they play: for example surveyors are authorized to insert into the database the spatial objects that have been detected on the field, whereas other operators, say citizens, might only be authorized to view a subset of data. Because of the different roles users play and also because some roles are allowed to modify the database, a controlled access is needed to provide a differentiated view of the information content and preserve data integrity. Moreover, another major requirement of the application is that operations are spatially bounded, that is they can be performed only within a given spatial context. For example a surveyor can only be allowed to insert the spatial objects located in the area of competence; likewise the regional officer may only be allowed to access the data pertaining to his own region.

In order to fulfill these requirements we propose an approach based on an extended notion of *authorization rule* (Bertino, 1998). The idea is to attribute a spatial dimension to the authorization rule by assigning a *geographical scope* that defines the spatial region in which the authorization is valid. Therefore, operations that users may request on spatial data may vary, depending on user identity and spatial objects' position.

Related work

Issues concerning data protection have been widely investigated for conventional database management systems (Bertino, 1998), (Bertino, 1997). There have been also several efforts extending conventional access control models to deal with new data types and models. Such efforts include access control models for Web pages and XML data, temporal access control models, extended access control models for relational databases, and access control models for Digital Libraries (Bertino, 1999), (Bertino, 1998), (Castano, 1997).

By contrast, we are not aware of many projects that are concerned with the protection of spatial data. An approach is presented in (Atluri, 2002) in which an access control system for a database of Earth images has been proposed to ensure a controlled dissemination of satellite images at different levels of resolution. In such a case, however, spatial data is limited to geo-referenced images whereas spatial objects, i.e. objects with a sharp boundary that occupy a position, are not considered.

Also commercial Spatial DBMSs have limited functionalities for controlling the access to spatial data. The spatial dimension is not taken into any special account so that the access is regulated by the same rules defined for the other kinds of data.

As far as we know, our approach is new in proposing the development of an access control model for spatial data on the Web. The paper is structured as follows: first we outline the access control model; next we introduce the architectural framework for the secure Spatial Web service; finally we briefly illustrate the case study.

OVERVIEW OF THE ACCESS CONTROL MODEL

A spatial authorization rule model

In general, when an access control system is applied, data access is regulated by an *access control policy*. Such a policy is expressed through a set of authorizations rules. In essence, an authorization rule states who is authorized to do what. Such rules are defined in accordance with an *access control model*.

The proposed access control model accounts for both the spatial dimension and the Web service context. It is based on the classical discretionary models defined for data management systems. As such it is based on the concepts of user, role, authorization rule. A user is a registered user, whereas a role is a function performed within an organization by a user. A user can be assigned multiple roles. Finally, an authorization rule states which operation can be performed by a role on an object within a geographical region. Let us consider in more detail the structure of the authorization rule.

An authorization rule is defined as a tuple consisting of the following fields:

- *Role*: the role specifies the "who" of the rule. A role is identified by a name, denoting a set of users sharing the same rights on spatial objects. For example "Regional Officer" is a role name. The role Administrator is a system-defined role and represents the top-level role.
- *FeatureClass:* it specifies the "what" of the rule. It consists of a OpenGIS Feature class (OpenGIS Consortium, 2003). For sake of simplicity, features are *Simple Features* (OpenGIS Consortium, 1999). A Feature class is thus identified by a name, has one or more attributes of the OpenGIS geometry type and one or more alphanumeric attributes. Note that in our model it is not possible to define authorization rules for objects at a finer level of granularity, for example, on attributes of Feature classes.
- *Privilege*: it specifies the "how" of the rule that is the operation that is to be performed on the above Feature class. The set of privileges are basically the named operations that the Web Service exposes through its interface and that can be indirectly invoked by the user through the Web client. The privileges are typically those for selecting, creating and modifying features.
- *Window*: it specifies the "where" of the rule. We have enriched the concept of rule with the notion of Window to indicate the geographical area in which the role is authorized to perform the operation on the Feature class. The Window is represented by either a polygonal region or a linear element.
- *Grantor*: the grantor is the role that granted the authorization. Likewise the classical authorization models, we assume that a role can delegate the administrative security functions to other roles. The mechanism used for delegating such functions is the classical mechanism of the grant option (see below).
- *Grant Option*: it is expressed as a Boolean variable; if TRUE the role is authorized to grant/revoke the rule to some other role and perform administrative operations.

The model consists also of constraints and operations. For what concerns the first, two constraints are defined over the Window field of the authorization rules. The first states that the Window in a rule must be contained in the windows of the grantor's rule. The second is introduced for expressing dependencies among privileges.

Concerning the operations of the model, operations are specified for manipulating users, roles and authorization rule sets. The most relevant are:

- CheckARule (r: Role, op: Privilege, f: FeatureSet): it checks whether a rule exists in the rule set authorizing a role *r* to perform the operation *op* over the feature set *f* . Should the rule exists the function computes the subset of f overlapping the rule's window;
- CreateARule (r: Role, rl: Rule): if the role *r* is authorized to perform administrative operations and if there is no violation of constraints, a new rule is added to the rule set, with *r* as grantor
- RevokeARule(r: Role, rl: Rule): if the role *r* is authorized to perform administrative operations and if there is no violation of constraints, the rule *rl* is revoked

The other operations are for: creating/removing users, creating/removing roles, assigning/de-assigning roles to users, activating a role for a user during a session.

An example

As an example, we consider the case of an environmental organization in charge of collecting and managing information about illegal waste deposits in the region named Lombardy. The (oversimplified)

information which is relevant for the definition of the access control policy is summarized in what follows.

We assume two roles: a) the regional officer in charge of analysing data (role *OfficerLombardy*); b) the surveyor that collects data exclusively on the Waste Deposits located in the area named Agrate (role *Surveyor*). The feature classes of interest are: the Waste Deposits and additional classes however used only for visualization and analysis purposes. We consider the following operations: GetFeature for simply retrieving the features of a given class, InsertFeature for creating a newFeature and AnalyzeFeature for performing some analysis, for example of proximity. We can define more formally the access control policy as follow:

Let R, P, FC, W be the sets of Roles, Privileges, Feature Classes and Windows defined as:

R={administrator, OfficerLombardy, surveyor}
P= {GetFeature, InsertFeature, AnalyseFeature)
FC = {WasteDeposit, AdministrativeBoundary, Road}
W={MBR, Lombardy, Agrate}

Note that for sake of readability, the windows are labelled with the name of the region.

The authorization rules are defined as follows:

a1= <administrator, ALL, ALL, MBR, _,true>
a2= <OfficerLombardy, GetFeature, ALL, Lombardy, administrator, true>
a3= <Surveyor, GetFeature, ALL, Lombardy, OfficerLombardy, false >
a4= <Surveyor, InsertFeature, WasteDeposit, Agrate, OfficerLombardy, false>

Rule a1 is the default rule stating that the administrator has full privileges on all feature classes over the full extent of the data (the label MBR stands for Minimum Bounding Region). The keyword ALL stands for "all possible values" for that field.

Rule a2 states that the role OfficerLombardy is authorized to retrieve only those features from all classes that overlap the Lombardy region. The role however is not authorized to create any new feature.

Rule a3 is similar to the previous rule, but for the role Surveyor.

Rule a4 states that the role Surveyor is authorized only to create features of class Waste Deposit exclusively in the area that is called Agrate. Note that such a region must be necessarily contained in the grantor's Window because of the model constraint.

Moreover the Officer can delegate administrative functions to other roles, while the Surveyor can't.

Consider now what happens when a user invokes an operation through the Web client. Suppose a surveyor asks to insert a set $\{d_1,d_2,\ldots,d_n\}$ of Waste Deposits located somewhere in a given region. The request is interpreted in this way: it is called the CheckRule operation to verify the existence of a rule in the form <Surveyor, Insertfeature, WasteDeposit, Window,-,->. Should the rule not exist, the operation would not be authorized. In this case, the rule exists (see rule a4), therefore the subset of features overlapping the rule's Window is selected and thus inserted.

ARCHITECTURAL FRAMEWORK

We consider now a possible architectural framework for a Web service including the above access control mechanism. The approach we propose is based on the well-known three-tier architecture consisting of Presentation, Application and Data Storage layers.

The Application layer consists of two main services as depicted in Fig. 1: the Access Control Service (ACS) and the Application Service (AS). The first exposes and implements the operations for authorization rules checking and administration. The second service exposes and implements the application logic and access the application data. When an operation is invoked by the user through the

Web client, the Application Service interacts with the Access Control Service to checks whether the operation is authorized and if it is, the operation is performed.

A convenient way for organizing the authorization rules is to store them into a spatial Dbms compliant with the OpengGIS Simple Feature model (referred to as Access Control DB). In such a way, the mapping of the authorization rules onto the spatial database is straightforward, as the schema of the authorization rule can be easily mapped onto a Simple Feature class as follows:

Feature Class AuthorizationRule

ID: ObjectID
Role: string
Privilege: string
FeatureClass: string
Window: OpenGIS Geometry Type
Grantor: string
GrantOption:Boolean

Following the above definition, each rule is given a unique identifier whereas an attribute is defined for each of the components of the rule. It should be noticed that the rule's Window is described through a geometric attribute. In such a way both spatial data and access control information are uniformly modelled in terms of Simple Features.

It has to be noticed that the application layer also includes an Authentication Service that can be based on username/password, Secure Socket Layer (SSL) or provide more complex service.

The typical interaction between the client and the services is as follows: the client connects to the system through the Authentication Service. Next the user decides which role to activate. Each request from the user is thus mapped onto one or more operations of the Application Service. This in turn interacts with the Access Control Service to verify whether the operation can be performed and where.

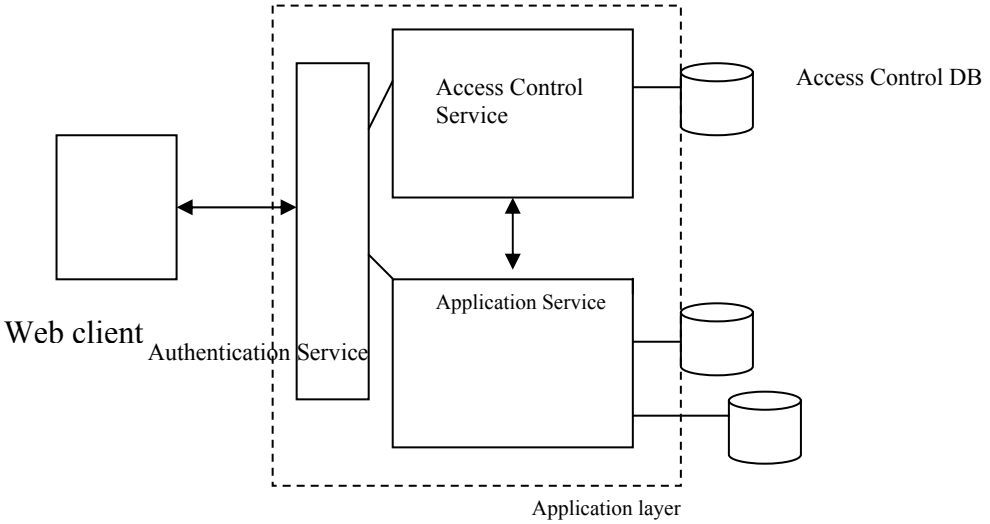

Figure 1: General architecture of the Web service

CASE STUDY AND PROTOTYPE

In order to prove the concept a case study has been developed consisting of a spatial Web service for the regional offices of a widely known environmental organization (WWF)(Momini, 2003). The general goal of the application is to allow the collaborative population of a database of environmental data pertaining to a major river basin in Italy. Spatial data consist of: soil use, critical elements in proximity of the river such as illegal waste deposits and illegal buildings and administrative entities. Features can be queried, created and modified using a Web browser . The user connects to the service through an authentication service based on username/password whereas each user has a profile that is described through a set of authorization rules.

We illustrate here, through a number of screen shots (in Italian), the effect of authorization rules on user interaction.

The first screen in Fig.2 shows what is initially presented to the user. The user needs to be authenticated if advanced functionalities are requested. The user *admin* (in the figure), when logged in, is assigned the role of administrator by default.

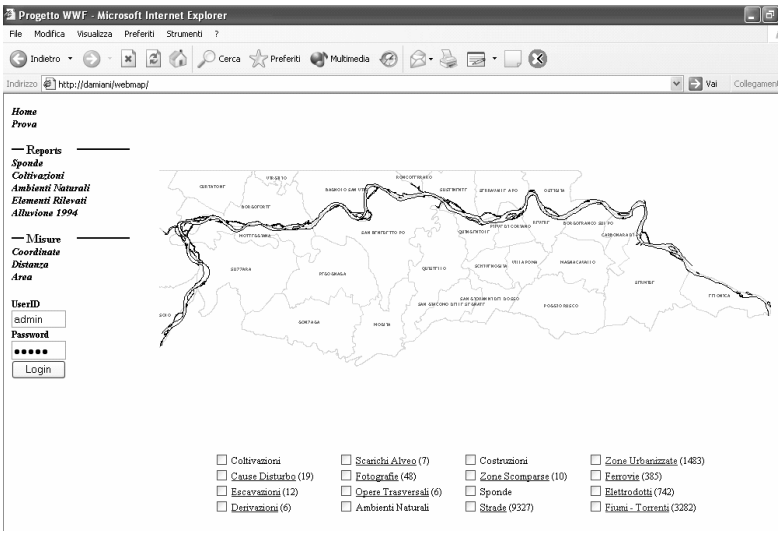

Figure 2: the initial screen

The Administrator has the full set of privileges over all feature classes. Therefore the administrator is allowed to retrieve, create and modify all the features of the class “urban centres”. These features are visualized by the client as reported in Fig. 3.

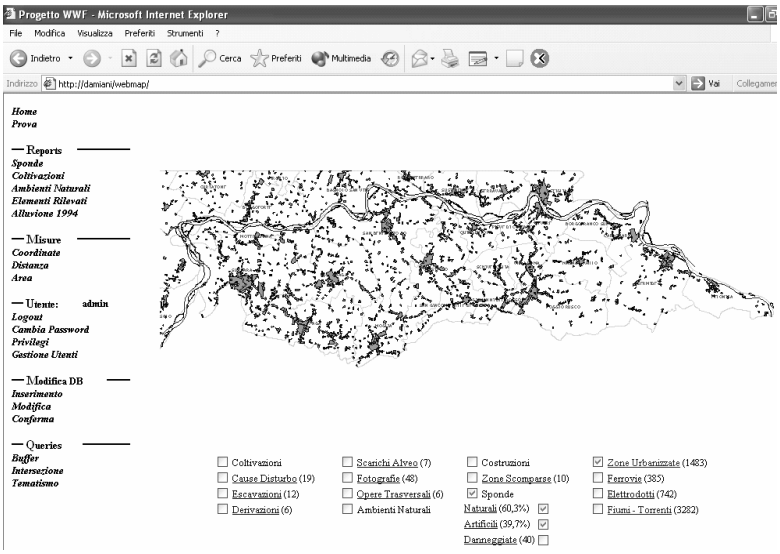

Figure 3: the whole set of features can be retrieved by the administrator and visualized though the Web client

However, a role can also grant an authorization rule to some other role, possibly restricting the geographical scope of that rule. We consider the case of the administrator that has granted the previous rule to the Surveyor specifying that the Window should be represented by the polygon depicted in Fig. 4.

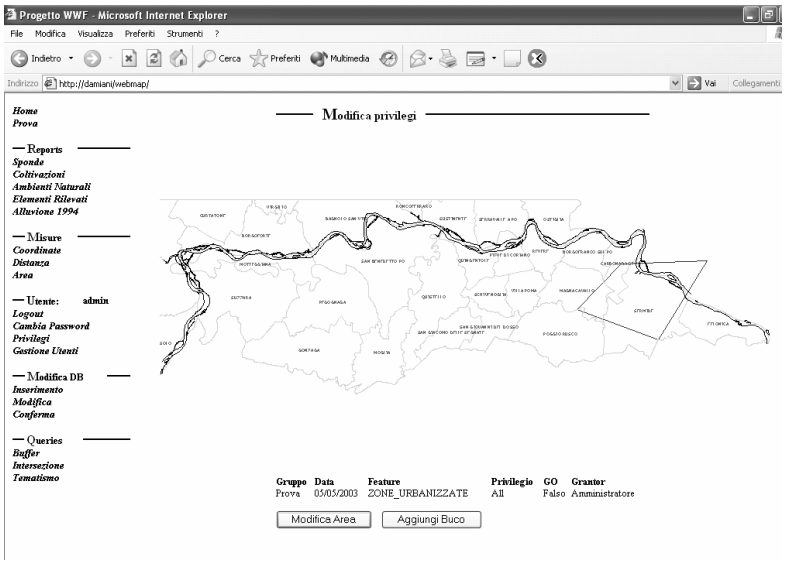

Figure 4: The polygon defining the Window of the rule

As a result, the role Surveyor, when logged in, cannot create features outside that window, and, for how the rule is defined, neither visualizes them. Therefore when the features of the class “urban centres” are retrieved only a subset of them is transferred back by the server. As you can note, only the urban centres overlapping the Window of the previous rule have been retrieved and then visualized.

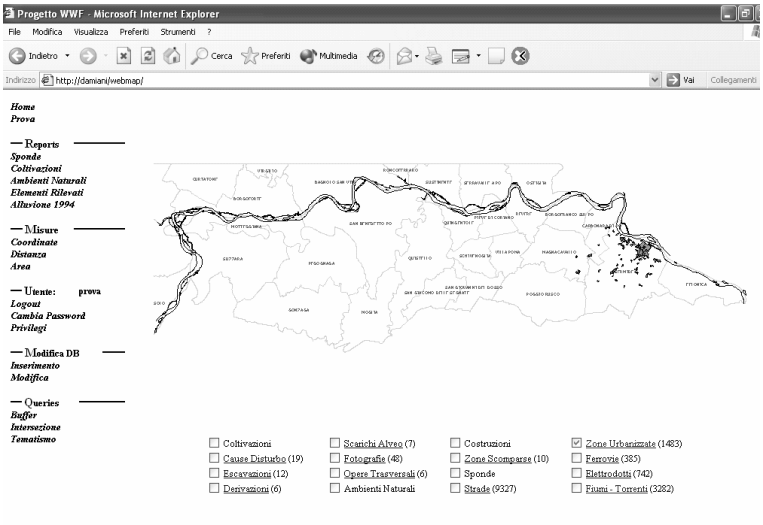

Figure 5: The privilege is thus spatially constrained

CONCLUSIONS

The issue addressed in this paper is the protection of spatial features on the Web through an access control mechanism. A simple model of authorization rules has been introduced and a possible architectural framework has been discussed. It has been presented also a case study of a Spatial Web Service for environmental monitoring support that has been developed using a commercial Web GIS platform (Intergraph Geomedia WebMap ™).

This work is a first contribution to the development of secure Spatial Web Services. Future extensions will regard: alignment of the access control model to the ongoing standardisation efforts in data security; extension of the approach to the OpenGIS Web services; model scalability and conceptual extensions.

ACKNOWLEDGMENTS

We wish to thank Andrea Agapito (WWF Italia) for the support and Davide Momini for the contribution to the development of the case study.

BIBLIOGRAPHY

Atluri V., Mazzoleni P., 2002, A Uniform Indexing Scheme for Geo-spatial Data and Authorizations,. 16th IFIP WG11.3 Working Conference on Database Security.

Adam N., Alturi V., Bertino E., Ferrari E., 2002, A Content-based Authorization Model for Digital Libraries. IEEE Trans. On Knowledge and Data Engineering, Vol. 14, N.2.

Bertino E., Damiani M.L., Momini D., 2004, An Access Control System for a Web Map Management Service. 14th International Workshop on Research Issues on Data Engineering, RIDE WS-ECEG'2004 , Boston (US).

Bertino E., Jajodia S., Samarati P., 1999, A Flexible Authorization Mechanism for Data Management Systems. ACM Trans. on Information Systems,Vol.17, No.2, pp.101-140.

Bertino E., Ferrari E., 1998, Data Security, in COMPSAC 1998.

Bertino E., Bettini C., Ferrari E., Samarati P., 1998, An Access Control Model Supporting Periodicity Constraints and Temporal Reasoning. ACM Trans. On Database Systems, Vol.23, N.3, pp.231-285.

Bertino E., Samarati P., Jajodia S., 1997, An Extended Authorization Model for Relational Databases. In Transaction on Knowledge and Data Engineering, Vol.9, N.1.

Castano S., Fugini M.G., Martella G., Samarati P., 1995, Database Security. Addison-Wesley.

INSPIRE Working Group, 2002, INSPIRE Architecture and Standards Position Paper.

Joshi J., Aref W. Ghafoor A., Spafford E., 2001, Security models for Web-based applications. Communications of the ACM, Vol. 44. N.2.

Momini D.., 2003, Sistemi Informativi Geografici via Web: studio di un caso. Tesi di Laurea, Universita' degli Studi di Milano.

OpenGIS Consortium, 1999, OpenGIS Simple Features Specification for SQL, OGC 99-049.

Open GIS Consortium, 2003, OpenGIS Reference Model, OGG 03-040.