CERIAS Tech Report 2002-08

Trustworthiness Based Authorization on WWW

Y. Zong, B. Bhargava, M. Mahoui
Center for Education and Research in
Information Assurance and Security
&
Department of Computer Science, Purdue University
West Lafayette, IN 47907

Trustworthiness Based Authorization on WWW*

Y. Zhong B. Bhargava M. Mahoui
Center of Education and Research in Information Assurance and Security
and
Department of Computer Science
Purdue University
West Lafayette, IN, 47907
{zhong, bb, mmahoui}@cs.purdue.edu

Abstract

Current approaches for authorization on Web servers are mostly based on a predefined set of users or domains. They are not suitable for Internet Web sites where the user set is unbounded and authorized users can be non-predefined. We propose an authorization approach that applies Role-based access control (RBAC) to WWW. Under this approach, system administrators predefine roles, role-permission relations and the policies that assign roles to users (user-role assignment policy). The system automatically collects trustworthy information (valid evidence) and assigns roles to Internet users according to user-role assignment policies. Trustworthiness information plays an important role in user-role assignment. The validity of evidence is assessed based on the trustworthiness information of the evidence provider. In addition, system administrators can specify the trustworthiness constraints that users have to satisfy for holding roles. In this paper, the schema of using RBAC on the Web and the procedure of user-role assignment are presented. The classification and evaluation of trustworthiness are discussed.

1. Introduction

WWW as an open environment provides a tremendous opportunity to share information and provide services to a large scale of users. However, authorization of Internet Web sites becomes more complex than traditional closed systems mainly due to the following factors:

- Unbounded user set: In traditional systems, the number of users is bounded. It enables user based access controls such as Access Control List (ACL) to be used. Applying user based access controls in the Web suffers from scalability issues, as the size of the users set is not bound.
- Non-predefined but authorized users: It is possible that an authorized user of Internet Web sites is non-predefined (e.g. the application example in [1]). New users are continuously introduced to the system and granted different permissions.
- Less prior knowledge about users: In traditional systems, permissions are granted to a user by system administrators based on the prior knowledge about the user (e.g. the user's job function or job title within the organization). In the Web, strangers are introduced to the system on the fly, which introduces the need to collect information about users (e.g. user's identity, age and job) and assign permissions to strangers according to the knowledge users accumulated by the system about users.

Current approaches for authorization on Web servers are mostly based on a predefined set of users or domains. They are not suitable for Internet Web sites where the user set is unbounded and authorized users can be non-predefined [3].

We propose an authorization approach that applies Role-based access control (RBAC) to WWW. RBAC is a promising technology for managing and enforcing security in large-scale systems [2]. The basic notion of RBAC is that permissions are associated with roles, and users are assigned appropriate roles. Security administration consists of two independent parts: role-permission assignment and user-role assignment. RBAC can support access control on Internet Web sites. System administrators predefine roles, role-permission relations and the policies that map users to roles, instead of predefining users and user-permission relations. The Internet users are assigned roles based on the mapping polices. A user's permission set is determined by the roles he holds.

Most research works on RBAC focus on closed enterprise-wide systems, where user-role assignment is relatively simple [2]. A user is assigned roles according to his specific job responsibilities in the enterprise manually or automatically [3,4]. For Internet Web sites, user-role assignment is much more complex. Unbounded user set makes it difficult, if not impossible, for system administrators to assign roles to individual users manually.

* This research is supported by CERIAS and NSF grants CCR-9901712 and CCR-0001788. This paper is published in IEEE workshop on "Security in Distributed Data Warehousing", New Orleans, Oct. 2001.

Before assigning a role to a user, Web sites need to collect information such as the user's age, job, etc. from different information sources (e.g. trusted third parties, local database, etc.) to verify that the user qualifies for the role according to the policies that map users to roles. For security reason, high privilege roles should be assigned to trustworthy users whom the system believes not to be defect. Web sites form trust opinions on users by receiving references from trusted intermediaries, analyzing users' interactions with system, etc. Our research investigates automatic user-role assignment on WWW. Trustworthiness information that is automatically assessed by the system is used in user-role assignment.

The rest of this paper is organized as follows. Section 2 summarizes related research. Section 3 describes our schema. Assigning roles to users is presented in section 4. Section 5 discusses the assessment of trustworthiness. Section 6 concludes the paper.

2. Related work

RBAC has rapidly emerged in the 1990s as a promising technology for managing and enforcing security in large-scale enterprise-wide systems. J. Park and R. Sandhu present a comprehensive approach to apply RBAC on the web by using technologies such as secure cookies and smart certificates [3]. However, their research is in the context of enterprise-wide system. They assume that the system administrator assign users to roles on the basis of users' job responsibilities in the enterprise.

In [1], A. Herzberg et all. propose an approach to map Internet users to predefined business roles based on public key certificates issued by third parties. The most important difference between their approach and ours is that the former does not consider trustworthiness of users and certificate issuers.

Among the significant body of research on trust management, the trust model and subject logic proposed by A. JØsang are close to our work [5,6,7].

3. Schema

Figure 1 shows a schema of using RBAC on the web. We distinguish two main components: Role server and RBAC enhanced web server.

Role server (figure 1) is responsible for assigning roles to a user: (1) determining the roles assigned to a user, (2) creating and signing a statement specifying the roles assigned, (3) sending the statement to the user's machine.

Statements should have two properties: (1) a user can prove that he is the owner of a statement. Since statements are stored in the user's machine, it is convenient for users to copy statements of other users. For this reason, when a user presents a statement, the system should verify that he is the statement owner. (2)

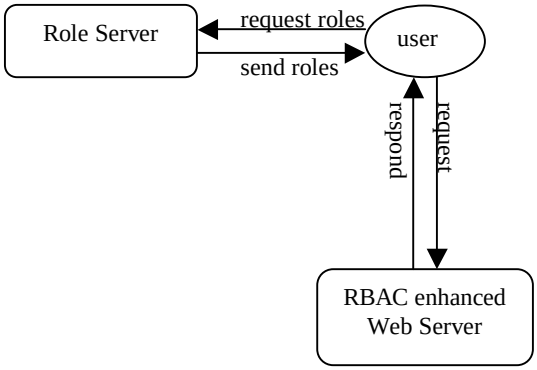

Figure 1 Use RBAC on the WWW

Statements are short-lived. For each statement, its validity is limited by an expiration date. Problems arise when system administrators want to revoke the roles held in a statement before it expires. A revocation scheme determining whether a certificate has been revoked can be used. However, we do not prefer revocation schemes for simplicity and performance reasons. If statements expire in hours or days, a revocation scheme is not necessary [10]. Therefore, we use short-term statements. Secure cookies and smart certificates are ideal forms of the statements. The technologies of secure cookies or smart certificates are discussed in [3].

The RBAC enhanced web server (figure 1) is a web server with role based access control. A reference implementation of an RBAC web server has been provided by National Institute of Standards and Technology [2].

If a user wants to access resources in a Web site, the Web site prompts what role(s) he should have. If the user has been assigned the role(s), he selects the proper statement that shows he holds the role(s) and sends it to the Web server. The Web server verifies the validity and the ownership of the statement. If all the checks are passed, the Web server returns the requested resources to the user.

If a user does not have the proper role(s) or the statement has expired, the user is directed to role server, which checks if the user qualifies for the role(s), as we discuss in Section 4.

In the process discussed above, a role is assigned to a user when the user needs it. As we discussed in section 1, the system may collect necessary user information from third parties such as public key certificates. The overhead of collecting process could be large since several chains of certificates may be followed. In addition, the required certificate set of different roles can be overlapped. In order to make best use of the information collected, we provide a pre-assignment mode. In this mode, we assume that resources on a Web site are categorized. A category-role table is needed. For each information category, there

is one row in the table storing the roles accessing the resources in the category. We also assume the existence of a user-modeling component that generates users' interested information categories. In a pre-assigning process, the system first determines a set of roles possibly required by a user based on his preferences and the category-role table. We call this set R_1. Then, the system determines the roles for which the user qualifies in R_1.

Example: Table 1 illustrates an example of category-role table. If Alice is interested in Health and Education, R_1= {Doctor, Cardiologist, Oncologist, Student, Faculty, Principal}. The system checks what roles in R_1 should be assigned to Alice.

Table 1 An example of category-role table

Category	$Role_1$	$Role_2$	$Role_n$
Health	Doctor	Cardiologist	Oncologist
Education	Student	Faculty	Principal
Business	CEO	Sales Manager	VIP Customer

4. Assigning role(s) to user(s)

The main responsibility of the role server is to assign roles to users. Unlike [1], we consider various information sources for assignment in addition to public key certificates. Furthermore, trustworthiness information is used. When a user provides signed statement (e.g. certificates) issued by a third party, the issuer's trustworthiness determines how much the statement can be trusted. System administrators specify the minimum trustworthiness value required for holding this role. The system assigns roles to users satisfying required constraints including trustworthiness constraints. Details of trustworthiness classification and evaluation are discussed in section 5.

4.1 Information sources

The information sources we consider can be classified as the followings:

1. A statement on the properties of the user made by known intermediaries. For example, a hospital wants to make some of its online resources available for health professionals from other hospitals. In order to access these resources, a user should provide a public-key certificate signed by another hospital, which proves that he is a doctor. This type of information is presented to the system as public-key based certificates [1]. The certificates coming from intermediaries cannot be 100% trusted. The reasons are: first, the intermediaries may not be honest; second, it is possible that a statement made by an honest intermediary cannot be directly used. For example, the system asks other website how much a user can be trusted. Two trustworthy websites may make very different conclusions because they use different evaluation criteria. A characteristic of our system is to consider how much a certificate can be trusted when using it for role assignment.
2. Information provided by users through interactive dialogues. For example, when a new user registers in a website, he is prompted to a set of questions (i.e. survey) related to his personal information. An example of smart user-modeling agent was proposed in [9]. Its main functionalities consist of generating the survey, verifying users' answers, and describing users' characteristics by using a set of field variables. We assume the existence of a similar user-modeling agent in our system.
3. Analysis of a user's interactions with the system. The system analyses Web logs to discover security events, summarizes users' behavior and adjusts the trustworthiness of users based on these results. For example, a user frequently attempting to access unauthorized data is considered as a potentially harmful user and his roles with high privileges may be revoked. The security events to be controlled are described as associated rules or sequential patterns [8]. Another example is that a user who has purchased many items on a commercial web site is assigned to a role with high privileges. For instance, the user is granted to access coupons that are only provided to selective users.

4.2 Assigning process

Figure 2 shows the procedure of assigning a role to a user.

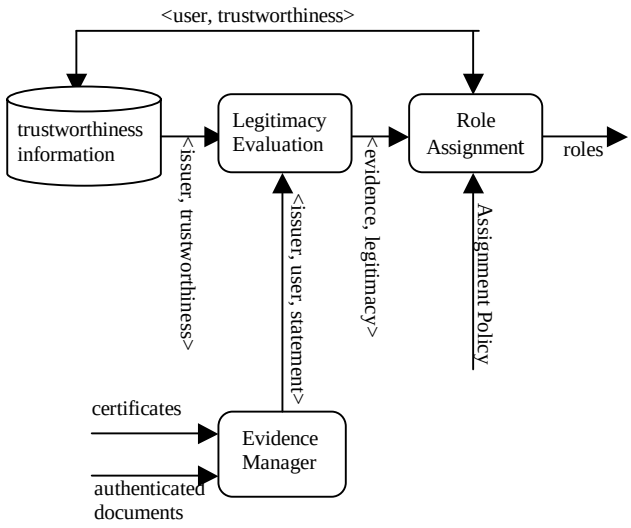

Figure 2 Assigning process

Step 1. Information with different formats (e.g. public key certificates or membership relations stored in local databases) is first transformed into the common interface of evidence by the evidence

manager (figure 2). Evidence is a 3-tuple <issuer, user, statement> representing a statement provided by an issuer on a user. An issuer can be an intermediary or the local system.

Step 2. Evidence is presented to legitimacy evaluation module (figure 2) that generates a 2-tuple <evidence, legitimacy>. The field of legitimacy indicates the degree to which the system believes that the corresponding evidence is true. It is a numerical number ∈ [0,1]. The higher the value, the more the system thinks the associated evidence is convincing. A legitimacy field is evaluated by using the issuer's trustworthiness information.

Step 3. <evidence, legitimacy> is provided to role assignment module (figure 2). A user is assigned appropriate roles based on the 2-tuples, assignment policies and trustworthiness threshold specified by system administrators. For each role, a set of assignment policies defines the content and number of evidences needed for the assignment. The policies also specify the minimum legitimacy of the evidences (legitimacy threshold). If the legitimacy of evidence provided to the system is smaller than the minimum legitimacy, this evidence is considered as invalid. It helps us to control the validity of the evidence. System administrators can specify that legitimacy threshold of evidences for assigning a role with high privileges is higher than that for assigning a role with low privileges. The trustworthiness threshold specifies the minimum trustworthiness that a user should have within the system to get a role. In this step, the role assignment module needs to assess a user's trustworthiness. For a new user, the assessment is based on the evidences (e.g. credit rating from third parties) and policies (e.g. All user from domain X has initial trustworthiness value of 0.8). For a known user, his trustworthiness is a function of the original trustworthiness, evidences and policies. A role is assigned to a user only if the user provides necessary valid evidences and has high enough trustworthiness. This step assigns roles to a user and stores his trustworthiness information into trustworthiness information database.

5. Trustworthiness

Trustworthiness plays an important role in our approach. As described in section 4, an issuer's trustworthiness information is used to determine the degree to which a statement made by the issuer is trusted. On the other hand, a user's trustworthiness is considered before roles are assigned to him. Intuitively, the trustworthiness in the former case is different from that in the latter case. In this section, we discuss the classification and evaluation of trustworthiness.

5.1 Trustworthiness Classification

We classify the roles into two types to simplify the design. We describe the role classification before discussing trustworthiness classification.

- Authority role*:* Making statements on certain properties of other users is one type of privileges that is only assigned to authority roles. The system only accepts the statements made by authority roles. Before accepting the certificates from an intermediary, the system checks whether the intermediary holds the proper authority role. For example, an intermediary can certify a user's identity as health professional only if it holds the authority role of hospital. An authority role has no permission to access the resources on the website (e.g. read or write documents). If a user needs both types of privileges (i.e. making statements and accessing the website), he must be assigned access roles as well as authority roles.
- Access role: A user must have access roles to access the documents on the Web. Different permission sets are associated with access roles.

The implications of trustworthiness of a user acting as authority roles and acting as access roles are different. The trustworthiness of a user acting as access roles is defined as clearance of the user. The trustworthiness of a user acting as authority roles is his sign trust.

- Clearance: clearance is a numerical number ∈ [0,1]. It represents the degree to which the website believes that a user will not do harm to the system (e.g. not disclosing sensitive data, not attempting to access unauthorized data). The higher the clearance is, the more the system believes that the user is trustworthy and valuable. Clearance is accumulated gradually over time. It drops sharply if harmful actions or potential harmful actions are discovered. In addition, system administrators can specify the policies to raise clearance as award to a user's contributions (e.g. the clearance of a user may increase as loyalty award for the user's purchase). We revoke roles with high privileges from hostile users and secure the system by using clearance.

  Example: A new user with certificates from a trusted hospital asks for the role of doctor whose minimal trustworthiness is 0.5. Assume the user's initial clearance is 0.75. He can get the role. The system assigns the doctor role to the user by storing a secure cookie on his machine. When the secure cookie expires, role server checks if the user qualifies for the role. Assume the user's clearance drops to 0.3. Although he

still holds required certificates, he cannot get the role since his clearance is smaller than the threshold 0.5.

- Sign trust: sign trust is a numerical number $\in$ [0,1]. It indicates the degree to which the system believes the statements made by the user are true. The higher the sign trust is, the more the system believes the user's statements. One important reason we introduce sign trust is that the conditions under which certification authorities issue certificates are different [7]. Most certification authorities state the conditions in Certification Practice Statements (CPS), which is a piece of lengthy and complex prose text. Instead of assessing CPS automatically to set sign trust, we attempt to use a simple approach to capture the difference. The idea is that an issuer's sign trust is adjusted based on the behaviors of the users introduced by the issuer. The following example explains this idea. A professor has recommended ten students to a graduate school. All students have poor academic performance. The recommendation letters from this professor become less convincing consequently.

5.3 Trustworthiness Evaluation

Clearance Evaluation:

- Initial values: The initial value of clearance is computed when a user gets the first access role by using the default values for the access role specified by the system administrator. We also consider the following factors if applicable: (1) the default clearance value associated with the domain/subnet from which the user comes (e.g. requestors outside United States may be assigned a different clearance value from those coming from United States), (2) the statements on clearance properties provided by intermediaries, (3) the results generated by the user-modeling agent.
- Adjustment: The system scans the Web log periodically and uses threshold-based approach or rule-based approach to detect "harmful behavior" (e.g. 10 attempts to access secure documents) and "suspicious behavior" (e.g. frequent request the online coupons without purchasing anything). System administrators specify penalty values for each type of misbehavior. The clearance values of misbehavior users decrease by certain penalty values. System administrators also determine the reward values. For those who do not behave in a harmful or suspicious manner, the clearance values should rise.

Sign_trust Evaluation:

- Initial value: Initially, sign trust is set when a user gets the first authority role by using the default values for the authority role specified by the system administrator.
- Adjustment: The sign trust of a user u is modified periodically together with the modification of clearance. Suppose u_1, u_2, ... u_n are assigned to access roles based on the certificate signed by u. The modification of sign trust of u is related with the changes of clearances of u_i ($1 \le i \le n$). We consider the behaviors of the users introduced into the system by u when modifying the sign trust of u.

  Formula:

$$\delta(\text{sign trust}(u)) = \text{sign trust}(u) - \text{previous\_ sign trust}(u)$$

$$\delta(\text{sign trust}(u)) = f(\delta(\text{clearance}(u_1)), \delta(\text{clearance}(u_2)), \ldots \delta(\text{clearance}(u_n)), n)$$

$$\delta(\text{clearance}(u_i)) = \text{clearance}(u_i) - \text{previous\_clearance}(u_i) \quad \text{where } (1 \le i \le n)$$

6. Conclusion

We present a new authorization approach on WWW. This approach utilizes RBAC for access control. Our research focuses on user-role assignment. Roles are automatically assigned to users based on user-role assignment policies. Especially, we mimic real-world assessment of trustworthiness in an automated fashion and use the assessment results in the user-role assignment.

7. References

[1]. A. Herzberg, Y. Mass, and J. Mihaeli, "Access Control Meets Public Key Infrastructure, Or: Assigning Roles to Strangers", In *Proc of the 2000 IEEE Symposium on Security and Privacy*, 2000

[2]. The website for ROLE BASED ACCESS CONTROL. http://csrc.nist.gov/rbac/

[3]. J. Park, R. Sandhu, "Role-based Access Control on the Web", In *ACM Transactions on Information and System Security, Vol. 4, No 1*, 2001

[4]. R. Sandhu, V. Bhamidipati, "Role-Based Administration of User-Role Assignment: The URA97 Model and its Oracle Implementation", In *Journal of Computer Security, Vol. 7*, 1999

[5]. The website for Trust Management. http://security.dstc.edu.au/projects/trust/

[6]. A. JØsang, "Trust-based Decision Making for Electronic Transactions", In *Proc of the Fourth Nordic Workshop on Secure Computer Systems*, Sweden, 1999

[7]. A. JØsang, "Trust Management for E-commerce", http://virtualbanking2000.com

[8]. R. Cooley, B. Mobasher, J. Srivastava, "Data Preparation for Mining World Wide Web Browsing Patterns", In *Knowledge and Information Systems, Vol. 1, No 1*, 1999

[9]. W. Tu, S. Tang, M. Mohania, "Web Databases and Agent Technology", *Technical Report, Department of Computer Science, West Michigan University*, 2001

[10]. A. Herzberg, H. Yochai, "Mini-Pay: Charging per Click on the Web", In *Proc of 6th International World Wide Web Conference*, 1997