1998 International Computer Symposium
Workshop on Cryptology and Information Security
December 17-19, 1998, N.C.K.U., Tainan, Taiwan, R.O.C.

Extension of the X.509 Authentication Framework for Access Authorization in Distributed Computing Environments

Kou-Chen Wu, Jing-Jang Hwang, and Duen-Ren Liu

Institute of Information Management,
National Chiao Tung University, Hsinchu, Taiwan, R.O.C.
Email:{u8434801,jjhwang}@cc.nctu.edu.tw, dliu@iim.nctu.edu.tw

ABSTRACT

The authentication framework recommended by ITU-T X.509 standard serves as a basis for entity authentication in distributed computing environments. This study extends this standard so that it also serves as a basis for access authorization. Such an extension is achieved by carrying the user's role-based authorization attributes with X.509 public-key certificates. The relevance of certificate management when the user's role assignment changes is also addressed.

1. Introduction

In commercial and governmental organizations, employees are frequently users but not owners of information resources. In contrast to the conventional discretionary or mandatory access control methods, accessing permission to information resources under RBAC (Role-based access control) are based on an individual's roles within the organization, rather than on user-object relationships. Thus, RBAC offers the advantages of a clearer representation and programmable enforcement of enterprise-specific security policies, as well as an improvement in the manageability of access control.

The RBAC concept can be illustrated with the *buyer's* role. As a buyer, he or she is authorized to perform operations, such as "***adding a new supplier record*** and ***checking the exact cost for some purchased materials*** on the ***inventory system***". This semantic level construct of "roles" associates users with their operatic permissions on objects. Thus, roles can be regarded as a user security attribute. Hereinafter, the term "user role attributes" refers to the granted roles which an organizational employees can play.

In a networking environment, the up-to-date values of user role attributes must be made available to all servers. Two approaches can be used to ensure that the session phase, in which a user activates his qualified role, has up-to-date values of user role attributes [1].

Cached user information: User information (including role attributes) is kept on each application server. Without having to access any other server, session processing can occur on each server. Although extra network communication is unnecessary when a session is established, the cache must be abreast of up-to-date user information whenever user information changes.

Uncached user information: User information, including role attributes, stored on a specific central server, e.g. the privilege attribute server (PAS) in SESAME system. This server is accessed during session processing, thereby requiring an additional network communication to establish a role session. Although used to validate a user's identity at each service server, the X.509 authentication framework is only used for identification purposes. With the user information stored in a central repository, the need to connect with the central server still interferes with the effect of distributed processing.

In light of above discussion, this study presents a novel scheme for storing users' role attributes to support both authorization and authentication functions in the application server. The proposed scheme offers the following benefits: (a) potential users do not need to register themselves to application servers before asking for services, and (b) servers do not need to cache users' role attributes. Thus, actual distributed processing can be supported.

To achieve the above objectives, this study integrates user authentication and RBAC (Role-based access control) by conveying user role attributes with the user's X.509 public-key certificate. Thus, the X.509 certificate not only declares the validity of a user's public key, but also serves as a credential of his or her role attributes. In the proposed scheme, revoking the whole certificate and issuing a new one would be unnecessary if part of the role attributes included in the credential changed. This certificate management problem is resolved herein by using reasonCode extension of X.509 CRL(Certificate Revocation List) entry to declare the changed user role attributes. Thus, only changed user role attributes are invalidated without revoking the certificate. By doing so, the uncompromised key and unchanged user role attributes are still valid.

The rest of this paper is organized as follows. Section 2 briefly reviews the concepts of Role-Based access control (RBAC) mechanism. The main features of X.509 public-key certificate are also presented. Section 3 then demonstrates how user role attributes can be incorporated into X.509 certificate. This section also describes how to manage the certificate revocation problem attributed to user role attributes change. Next, section 4 analyzes the characteristics of the proposed access authorization method.

Conclusions and areas for future research are finally made in Section 5.

2. Literature review

Before describing the proposed method which uses X.509 public key certificate as a credential of user's RBAC attributes given in section 3, this section reviews RBAC and the X.509 standard.

2.1 RBAC

The increasing trend towards role based access control (RBAC) is apparent, as evidenced by its ability to express access control policy in the manner in which an administrator views organizations. Under RBAC, access permissions are associated with roles but not based on an individual user's permissions on objects. A role is primarily an enterprise-level construct, implying multiple meanings. In addition to representing specific task competency, such as that of a physician and a pharmacist, a role can also embody the authority and responsibility. Moreover, it can also reflect the duty assignments rotated through multiple users such as for a duty physician.

In a related work, Sandhu et al.[3] proposed the most fundamental model for RBAC, $RBAC_0$. This model consists of the following components: (1) users, roles, permitted operations, and sessions (2) the many-to-many relationships between roles and permitted operations (3) the many-to-many relationships between users and roles (4) the one-to-many relationships between users and sessions, i.e. a user can create many sessions at a time, while every session must belong to just one user (5) the one-to-one relationships between sessions and roles, implying that a user's session is associated with the partial set of his or her authorized roles.

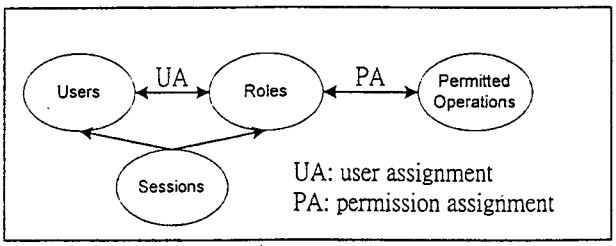

Fig.1. Role-based access control model 0 [3]

From the perspective of processing, RBAC can be divided into three phases: administration, session, and enforcement phases [1]. During **administration**, the business administrator defines roles in line with the organizational policies and goals, as well as the permitted operations associated with each role to perform its job function. Finally, users are assigned as members of specific roles by the administrator based on their responsibilities and competency. During **session**, each session is mapped for a user and a partial set of his or her qualified roles. A user can establish several communication connections and activate a specific role. In addition, identity authentication of the subject (either the user or the program on his behalf.) and the association between the subject and the user's security attributes must be processed. During **enforcement**, access request is accepted or denied according to the user's role attributes and the object's security attributes.

Although the above $RBAC_0$ fulfills the requirements of ordinary access control, more complex RBAC models more thoroughly address the issues of role hierarchies and constraints on role activation. In sum, RBAC can support the security principles of least privileges, separation of duties, and enterprise level data abstraction. Consequently, the security administrator can regulate the access privileges of organizational employees to commercial resources, access points, operation precedence, as well as the context dependent constraints in a manner more closely approaching organizational management.

2.2 X.509 standard

The ITU-T X.509 standard (ISO/IEC 9594-8)[5] provides an entity authentication framework to users of open system directories. The authentication method is established on the basis of public key cryptography. To bind a user with his or her public key, CA (Certificate Authority), a trusted third party is needed to issue the key owner a public key certificate. In addition, the public key certificate is a statement to the certificate user that the public key belongs to a specific user.

The X.509 public key certificate consists of a version number, serial number, signature algorithm identifier, certificate authority name, valid period, subject name, certificate authority identifier, subject identifier, subject's public key, and signature on the above information. The version 3 of X.509 standard adds extensions, subsequently providing more flexibility to carry more public key related information on the certificate, such as the public key identifier of the certificate authority, certificate issuing policy, and restrictions on usage of the public key.

The structure of X.509 v3 certificate extensions consists of three fields: type, criticality, and value. Type is the unique identifier of the extension field. Extensions can be labeled as critical or non-critical by setting the criticality flag. The value field contains the exact value of this extension. Figure 2 illustrates the X.509 v3 extensions [4]. This figure also contains two extensions: (a) the certificate type, which indicates the applicable service, and (b) the identifier of the authority's public key identifier.

Extensions	
Identifier	Certificate Type
Critical	No
Certified Usage	SSL Client Secure E-mail
Identifier	Authority Key Identifier
Critical	No
Key Identifier	Ac:21:86:de:0f:4f:90:80:e5:e7:22: 1b:4c:77:bd:f3:76:10

Fig.2. X.509 Version3 Extension sample [4]

Among the standardized extensions, CertificatePolicies attempt to identify the policies under which the certificate is issued. The certificate policy value refers to the applicable group and application classes under ordinary security needs. For example, an e-mail policy indicates that the public-key certified by the CA is used for securing daily electronic mail transmissions, while an e-commerce policy indicates that the key usage is for network commerce.

The certificate policy is represented by a previously registered object identifier (OID) on the certificate, in which the application system should be configured well in advance to recognize the necessary policy. For further details, the application system must be able to obtain the OID of the necessary policy, and refer to the full statement relating to the policy by this OID.

A sub-field associated with policy is the policy qualifier, which can convey policy related information. Figure 3 presents the syntax notation of policy qualifier represented in ASN.1. The X.509 standard does not restrict the usage and the syntax for the data. As long as the organization agrees upon the usage purpose and syntax, usage of the policy qualifier is meaningful. For example, a policy qualifier can be used to carry some meaningful textual information, such as the WWW URL for the public-key certificate policy statement. Thus, the user of the public-key certificate can obtain the complete statement of the certificate policy statement.

```
PolicyQualifierInfo ::= SEQUENCE {
policyQualifierID   CERT-QUALIFIER.&id
qualifier           CERT-POLICY-QUALIFIER.&Qualifier
                    OPTIONAL }
CERT-POLICY-QUALIFIER ::= CLASS {
&id                 OBJECT IDENTIFIER UNIQUE,
&Qualifier          OPTIONAL }
```

Fig.3. ASN.1 for policy qualifier [6]

Certificates must occasionally be revoked. A notable example is when the certificate holder leaves the issuing organization or when the private key is compromised. The mechanism defined in X.509 for revoking certificates is the Certificate Revocation List (CRL). A CRL, signed by a CA, contains a list of unexpired certificates that have been revoked. The distribution of CRL can be either pushed from the central repositories or pulled by the verifiers on a needed basis.

In X.509 v2, CRL also defines optional extensions both for the X.509 CRL as a whole and for each entry in the list. In particular, the reasonCode, i.e. an extension of the CRL entry, is used to identify the reasons for revocation, including key compromise, affiliation change, and so on.

Extension	Use	Critical
ReasonCode	**Identifies the reason for the revocation of this certificate**	No

Fig.4. ReasonCode extension for each CRL entry

3. Proposed method

As mentioned earlier in the Introduction, even X.509 authentication procedure can be applied under RBAC, a communication bottleneck may also occur when fetching user role attributes from the central privilege server. The solution proposed herein to resolve this problem integrates the user role attributes into the X.509 public key certificate.

The X.509 public key certificate serves as a statement by the certificate authority (CA) to the users that the public key belongs to a specific subject. Following the same reason, the security administrator can put the user's role attributes on the public-key certificates, thereby claiming the relationship between the certificate holder and the role attributes.

Details of how to implement the above notion are given in the same manner as RBAC procedure: administration, session, and enforcement.

3.1Administration stage

During the administration stage, the business manager assigns users to roles, and embeds this role attributes into the users' X.509 public-key certificates. As mentioned earlier in section 2, the certificate policy specifies the applicable group and application services under common security requirements. The authentication function of X.509 standard can be extended to include authorization by the certificate policy to imply RBAC authorization.

During implementation, the organization can define a certificate policy identified by "Role-assignment". Doing so enables the X.509 certificate to support both identity authentication and access authorization. Upon recognizing this specific Object identifier—**Role-assignment**, the server knows that user role attributes are conveyed in this certificate. The server can then proceed with the challenge-response procedure to authenticate the communicating entity, and make the access control decision based on the role attributes.

Although the public key certificate and the proposed Role-assignment policy are available, details of the method for carrying the user role attribute on the certificate must be given. According to the standard, the syntax for certificate policy and policy qualifier in ASN.1 according to Fig. 3 is described as follows.

Policy Identifier	Policy_Object_Identifier	
	Policy Qualifier	Policy_Qualifier_ID
		Qualifier

Fig.5. Certificate policy and its qualifier

The above figure illustrates the proposed design for conveying the user's role attributes on X.509 public-key certificate. According to this figure, the Role_Assignment policy is a registered unique object identifier. Role-ID is also a unique identifier, which represents an organizational role. The Policy Qualifier field states the role activation constraints. This design allows the certificate user to further understand the public key owner's role attributes from

the certificate. For example, a business member assigned to the role of IT-Department-Manager will have the following information recorded in his public key certificate.

Policy Identifier	Role_Assignment	
	Policy Qualifier	IT-Department-Manager
		Role Activation is limited at IT department

Fig.6. Example of a Role_assignment policy

In the recommendation of X.509 standard, CRL can be used as a statement to the verifier under a situation in which the user's key is compromised. When X.509 public key certificate is used to convey user role attributes, either key compromise or user role attributes update leads the certificate into revocation. If the attributes included in the certificate change, the certificate must be revoked and a new one issued. However, issuing a new certificate would be too expensive.

To avoid issuing a new certificate while making the up-to-date user role attributes available, we can publish the changed user role attributes, for example in a directory service. By doing so, the certificate verifier can refer to it. The solution proposed herein to resolve this problem uses the CRL to carry the revoked role IDs. In this manner, the user's X.509 certificate is still effective even though some of the user's role attributes have changed.

Extension	Use	Critical
ReasonCode	**IT-Department-Manager**	**No**

Fig.7. Declaring role change with ReasonCode

3.2 Session and enforcement phase

Figure 8 depicts the message exchanges of entity authentication and service request during session and enforcement. The challenge-response procedure consists of four steps. Step 1: The user sends his or her X.509 public-key certificate as well as access request to the server. Step 2: The server selects a random number as nonce, encrypts it with the received public key, and sends it back to the requester. Step 3: The user decrypts the received information, signs the result, says r, and sends it back to the server. Step 4: The server verifies the signature with the public key, and confirms whether if the result r is the exact value selected in step 2.

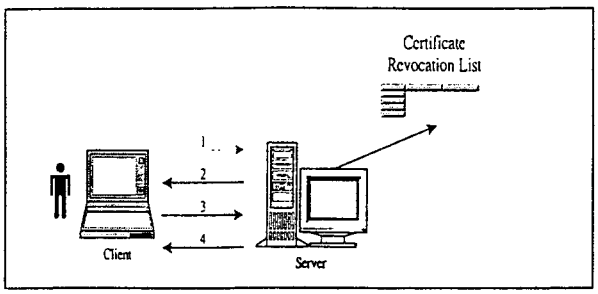

Fig.8. Entity authentication during session

In step 1, the server acquires the user's X.509 public key certificate, and can obtain the user's role attributed to it. Although carrying the role identifiers with it, the authentication procedure is necessary to ensure that he or she is who the certificate claims to be. In addition, to ensure the effectiveness of this public key certificate (no key compromise or user role attributes change before the certificate expires), the server must check the CRL (Certificate Revocation List, CRL). If finding that the user's certificate is an entry of the CRL, the server can further check the revocation reason extension to know the exact reason why the certification is revoked by referring to the reasonCode field. If this accounts for why the key is compromised, then the public key is no longer effective. If this accounts for why the user's role attribute is changed, the attributes except the revoked one are still valid.

The following steps 2 ~ 4 ensure that the client user is the owner of the certificate. To authenticate the user, server encrypts a randomly selected r with the public key on the certificate. The server accepts the notion that the client user is the one who owns the certificate if and only if the client returns the exact r. Notably, the authentication procedure is based on securely controlling the user's private key.

Once the client user passes the authentication procedure, the server assumes that the client is the owner of the certificate. Thus, the relationship between the user and his or her role attribute is established. The server then makes the access decision on the basis of the user's role attributes and the local stored object security attributes.

3.3 Illustrative example

Next, two roles in the IC testing industry are considered as examples: the material planner and the buyer. A material planner attempts to forecast material requirements, manages the usage status of the materials, order new materials from suppliers through the buyer and maintain a good grip of the purchasing progress. The buyer largely focuses on contacting suppliers, and purchasing materials at a reasonable price from suppliers according to other employees' purchase request.

Assume that employee A is assigned two roles: "material planner" and the other is "Employee welfare committee member". Meanwhile, while employee B is assigned the "Buyer" role. According to the proposed method, the extension fields about Role-assignment policy of A's and B's X.509 public-key certificates are:

Policy Identifier	Role_Assignment_Policy	
	Policy Qualifier	Material planner (unique identifier)
		Role_Activation_Constraints
	Policy Qualifier	Employee welfare committee member (unique identifier)
		Role_Activation_Constraints

Fig.9. Role assignment for employee A

Policy Identifier	Role_Assignment_Policy	
	Policy Qualifier	Buyer (unique identifier)
		Role_Activation_Constraints

Fig.10. Role assignment for employee B

Also assume that four information systems exist: purchasing system, inventory system, manufacturing execution system, and order entry system. The privileges of material planner and Buyer on the four systems are listed below, in which the individual end system maintains these object security attributes.

	Material planner	Buyer
Purchase system	authorized to order materials with the online system.	No
	No	Authorized to transform the material orders, which are recorded on the system, into paper-based purchase order.
Inventory system	Authorized to create and modify the material specs., as well as the conditions on purchasing and repositing materials.	No
	Authorized to check the current amount of materials in the repository.	No
	No	Authorized to add new supplier data.
	No	Authorized to check the actual prices of materials.
Manufacture execution system	Authorized to check the schedule and current status of production line.	No
Order entry system	Authorized to modify customer data.	No

Fig.11. Privileges of two illustrative roles on four systems

Further assume that user A and user B separately make two communication establishments with the inventory system and manufacturing execution system. The two servers can proceed with the entity authentication procedures with the user's public key certificate. Meanwhile, these servers can make the access control decisions by referring to user A and user B's security attributes and the local stored object security attributes.

4. Analysis

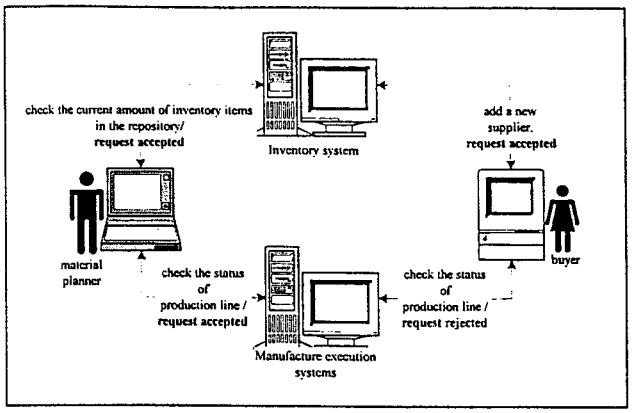

Fig.12. An example of access control

RBAC user role attributes refer to the roles, which a user is qualified to play within an organization. The advantage of storing user role attributes in a central privilege attribute server is that most up-to-date user information is made available. However, it is limited in that additional communication connections are necessary to acquire the user's security attributes.

This study presents a novel means of conveying the user'srole attributes in ITU-T X.509 public key certificate, a way which resembles the function of ordinary identity card. In practice, everyone is given an identity card issued by a trusted governmental agent. The ID card supports the function of identity authentication, as well as provides pertinent personal information, such as carrier and school credential. In the method proposed herein, the X.509 certificate intuitively models the ID card. Meanwhile, the CA, which certifies the user's public key and his role attributes, models the registry office. The ability of the proposed method to integrate both functions of authentication and authorization eliminates the problem of communication bottleneck of privilege attribute server.

L.Harn and H-Y.Lin[7] proposed the notion of integrating user authentication and authorization into a single module. Those investigators integrated the user's password and conventional access control matrix to promote the system security by mandating that authentication is executed whenever a user access request is made. Based on the same concept, the proposed method integrates user authentication and authorization by embedding the user's role attribute into the X.509 certificate. When an application server verifies the user's identity with his certificate, access control decision can be made concurrently, thereby promoting system security.

However, including user role attributes in the X.509 public key certificate leads to the disadvantage of certificate revocation. If parts of the role attributes included in the certificate are changed, the certificate must be revoked and a new one issued. The fact that regenerate certificates are often quite expensive accounts for that it's only appropriate to include user attributes in the certificate when the attributes are not changed frequently.

The credential certificate will hopefully not be abandoned when an attempt is made to reduce the cost of certificate management while only parts of user's role attributes are changed but the public key is not compromised. The

method proposed herein utilizes the reasonCode extension of CRL entry to convey the changed user role attributes. Notably, a new certificate does not need to be generated when the public key is valid and some of the user's roles have changed. Thus, the cost is reduced. If the certificate is listed on the CRL, the certificate user can verify the validity of public key and the authorization code simultaneously.

5. Conclusion

Stephen Wilson has contended that widespread electronic commerce leads to a greater emphasis on parties' roles, and the ability of others to verify those roles [8]. This study largely supports Wilson's viewpoint.

This study presents a novel method to convey user's RBAC attributes with X.509 public-key certificate. This design facilitates the integration of public-key based identity authentication and user operation authorization. Thus, actual distributed processing can be achieved. Also addressed herein is the certificate management problem while certificate is revoked due to user role attributes change.

Some other security services may also be enabled with a role certificate, including electronic document verification of authorization at source, authorization of task in automatic workflow environment, inter-domain access control and delegation of user's privileges. These relevant topics have promising implications for electronic commerce and, therefore, deserve further research.

6. REFERENCES

[1] John Barkley, "Comparing Simple Role Based Access Control Models and Access Control Lists," *National Institute of Standards and Technology*, Aug.11, 1997.

[2] D. Ferraiolo, J. Cugini, and D.R. Kuhn, "Role Based Access Control: Features and Motivations," *Annual Computer Security Applications Conference. IEEE Computer Society Press*, 1995.

[3] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman, "Role-Based Access Control Models," *IEEE Computer*, pp. 38-47, February 1996.

[4] Christopher King, "Building a Corporate Public Key Infrastructure," *Computer Security Journal*, vol. X Ⅲ,No. 2, 1997.

[5] ITU-T Recommendation X.509 (ISO/IEC 9594-8) (1993 E), Information technology—Open Systems Interconnection—The Directory: Authentication Framework.

[6] ISO/IEC JTC 1/SC 21/WG 4 and ITU-T Q15/7 Collaborative Editing Meeting on the Directory, "Final Text of Draft Amendments DAM 4 to ISO/IEC 9594-2, DAM 2 to ISO/IEC 9594-6, DAM 1 to ISO/IEC 9594-8 on Certificate Extensions," April 1996.

[7] L.Harn, H.-Y. Lin, "Integration of user authentication and access control," *IEE PROCEEDINGS-E*, Vol. 139, No.2, March 1992.

[8] Stephen Wilson, "Certificates and trust in electronic commerce," *Information Management & Computer Security*, 5/5, 1997, 175-181.

[9] Yoshiki Sameshima and Peter Kirstein, "Authorization with security attributes and privilege delegation Access control beyond the ACL," Computer Communications 20 (1997) 376-384.

[10] B. Clifford Neuman, "Proxy-Based Authorization and Accounting for Distributed Systems," *Proceedings of the 13th International Conference on Distributed Computing Systems*, Pittsburgh, pp.283-291, May 1993.

[11] M. Winslett, N.Ching, V.Jones and I.Slepchin, "Using digital credential on the World Wide Web," *Journal of Computer Security*, 5 (1997) 255-267.

[12] Mark Vandenwauver, Rene' Govaerts, Joos Vandewalle, "How Role Based Access Control is implemented in SESAME," *Sixth Workshop on Enabling Technologies: infrastructure for collaborative enterprises: WET-ICE '97 (June 18-20; 1997:MIT, Cambridge, Mass.)*, IEEE Computer Society press, 1997.