Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 10, October 2014, pg.799 – 803

RESEARCH ARTICLE

Role-Based Cryptography

P. Vijaya Sree[1], G. Praveen Babu[2]

[1]Dept. of Computer Science & School of IT, Jawaharlal Nehru Technological University, India
[2]Dept. of Computer Science & School of IT, Jawaharlal Nehru Technological University, India
[1] vijayasreepagolu@gmail.com

Abstract: Even though role-based access control (RBAC) can tremendously help us minimize the complexity in administering users, it still needs to realize the notion of roles at the resource level. In this paper, we propose a practical cryptographic RBAC model, called role-key hierarchy model, to support various security features including signature, identification and encryption on role-key hierarchy. In addition, several advanced features, such as role or user revocation, tracing, and anonymity, are implemented as well with the help of rich algebraic structure of elliptic curves; we introduce a unified and complete construction of role-based cryptosystem to verify the rationality and validity of our proposed model.

Keywords: role-based access control, cryptography, encryption, role-key hierarchy, elliptic curve, role-based cryptosystem

1. Introduction

Role-based access control (RBAC), as a proven alternative to traditional access control including discretionary access control (DAC) and mandatory access control (MAC), has been widely adopted for various information systems over the past few years [14]. Even though RBAC can tremendously help us minimize the complexity in administering users, it is still needed to realize the notion of roles at the resource level. In other words, RBAC systems need to control a user's access to resources as well as resource- level management based on roles. In distributed environments, we can leverage RBAC models to enforce fine-grained policies for sharing resources [10]. However, the current cryptosystems do not support such shared modes because the encryption/decryption keys can- not be recognized between RBAC systems. As a consequence, the resources should be re-encrypted when they are transferred into another domain. Obviously, it is necessary to design an efficient cryptographic mechanism compatible with corresponding access control systems. In fact, the research for cryptographic hierarchical structure has a long history since hierarchical structure is a nature way to organize and manage a large number of users. Several approaches on cryptographic partial order relation supporting hierarchical structure have been proposed. Akl and Taylor introduced a simple scheme to solve multilevel security.

2. Partial Orders

Let = hP,_i be a (finite) partially ordered set with partial order relation _ on a (finite) set P. A partial order is a reflexive, transitive and anti-symmetric binary relation. Inheritance is reflexive because a role inherits its own permissions, transitivity is a natural requirement in this con- text, and anti-symmetry rules out roles that inherit from one another and would therefore be redundant. Two distinct elements x and y in are said to be comparable if x _ y or y _ x. Otherwise, they are incomparable, denoted by xky. An order relation _ on P gives rise to a relation ≺ of strict inequality: x ≺ y in P if and only if (or iff) x _ y and x 6= y. Also, if x is dominated by y, we denote the domination relation as x ≺d y. In addition, if x ≺ y and x _ z ≺ y, it then implies z = x. The latter condition demands that there be no element z of P satisfying x ≺ z ≺ y. We define the predecessors and successors of elements in = hP,_i as follows: For an element x in P, ↑ x = {y ∈ P|x _ y} denotes the set of predecessors of x, ↓ x = {y ∈ P|y _ x} denotes the set of successors.

3. Role-Key Hierarchy Structure

In order to incorporate cryptographic schemes with RBAC, we propose a new hierarchy structure called Role-Key Hierarchy (RKH). Based on the hierarchical RBAC model, we define RKH as follows: Definition 2. [Role-Key Hierarchy]: Given a role hierarchy hR, i in RBAC, role-key hierarchy is a cryptographic partial order relation for the sets of users, keys, and roles, denoted by H = hU,K,R,_i, satisfying the following conditions:

1. K = PK∪SK, the key set K includes the role-key set PK and the user-key set SK;
2. UKA ⊆ U ×SK, a one-to-one user to key assignment relation, i.e., each user ui,j ∈ U is assigned to an exclusive user-key ski,j ∈ SK;
3. RKA ⊆ R×PK, a one-to-one role to key assignment relation, i.e., each role ri ∈ R corresponds to a unique role-key pki ∈ PK;
4. KH ⊆ PK × PK, is a partial order on PK called the key hierarchy or key dominance relation, also written as _; and
5. Each user ui,j can access the resources associated with rl if and only if rl _ ri ∈ RH and (ui,j , ri) ∈ UA.

where, hK,_i is the smallest partially ordered set satisfying the above conditions. The user holds multiple user keys if he is member of multiple roles in role hierarchy. In RBAC systems, various access control functions are designated by permissions P. In the same way, the RBAC permissions can be designated by some cryptographic algorithms, such as Encrypt and Decrypt, which can realize various access control functions by using role keys and user keys in role-key hierarchy.

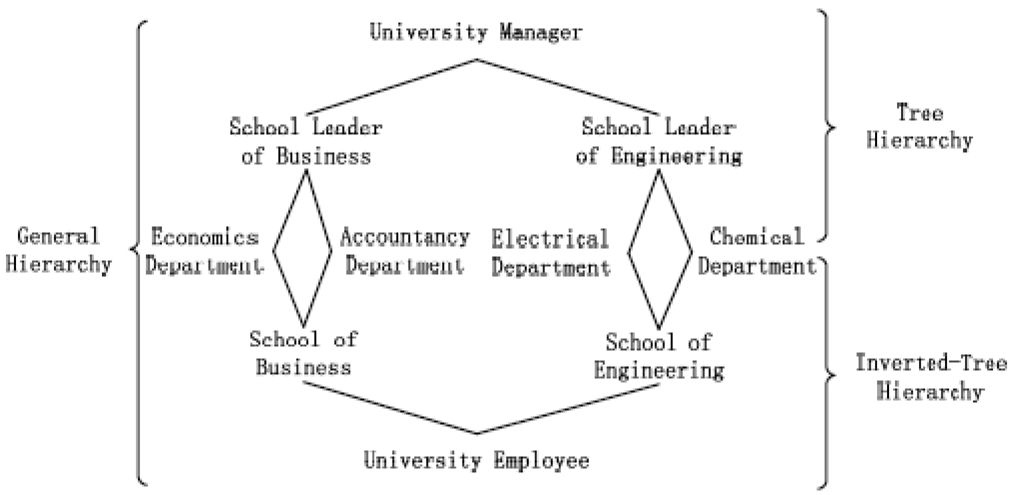

Figure 1: Example of role hierarchy with tree, inverted-tree, and general hierarchies

4. Role-based Cryptosystem

We expect that a system manager assigns the user key ski,j = (labi,j , dki,j) to a user, where labi,j is a public label and dki,j is a private key. This label labi,j can be used to realize special functions such as designation, revocation, and tracing. Given a role hierarchy = hR,_i and a security pa- rameter s, Role-based Cryptosystem (RBC) is a key management system that can construct a role-key hierar- chy H = hU,K,R,_i on and generate all keys on H, which is specified by three randomized algorithms, Setup, KeyRGen, and AddUser, described as follows:

• Setup(s,): Takes a security parameter s and a role hierarchy as an input. It produces a manager keymk and an initial parameter params, that is, Setup(s,) → {H,mk, params}. • GenRKey(params, ri): Takes the parameter params and a role index ri. It generates a role key pki in ri, that is, KeyRGen(params, ri) → pki

. • AddUser(mk, ID, ui,j): Takes a user identity ID, a user index ui,j , and the manager key mk. It out- puts a user secret key, which involves a user label labi,j and a private key dki,j , for the user ui,j , that is, AddUser(mk, ID, ui,j) → ski,j = (labi,j , dki,j). The user label labi,j is added to the public encryption key:

params = param sS {labi,j}.

In public-key settings, a user does not hold any private information and the permission process is performed only with the help of the public role key {pki} containing the user's labels {labi,j}, which is also called as ID-based RBC because the user's public labels can be used to support the various functions.

5. Security Goal of RKH

Obviously, security requirements in general cryptosystem are not sufficient enough to reflect the requirements of role- key hierarchy. It is important to consider typical attacks when we try to design key hierarchy and its schemes. In contrast with existing key hierarchy, RKH has several unique

Features

: 1. Each user ui,j is assigned to an exclusive user key ski,j , by which certain users can be chosen or identified in the processes of encryption, revocation, and tracing;

2. Public-key cryptography can be introduced to ensure the security of a user's private key even if the role key makes public in some systems. Therefore the role keys can be stored anywhere by RBAC systems; and

3. The derivation function of a user's private key is for-bidden even for the cases of partial order relations, Pr[Delegate(ski,j, cl) = skl,j′] ≤ ȩ, ∀cl _ ci. (1) where, ȩ is small enough. Hence a user cannot use this capability to obtain new keys or identities.

In order to ensure system security, RKH also needs to satisfy following properties:

• Each user in a role cannot get permissions to access another role's objects except for its subordinates, Also, a user cannot forge other's secret keys;

• The role key can be modified to satisfy the requirements of constraint policy, but it should not interfere with the issued keys of others; and

• To support the capability of audit capability, there exists an efficient tracing algorithm to identify the corrupted users or gain the corresponding evidence. The RKH is a group-oriented cryptography with "1:n" character, where one role key corresponds to many user keys. Hence, in addition to passive cryptanalysis, the collusion at- tack is a major attack, which focuses on changing the privilege of the granted users or getting the other users' keys. This kind of attack involves the following cases:

• Collusion attack for framing users, in which the corrupted users in R = {uik,jk}t

k=1 wish to forge a new or unused key in U \ R (called as honest user). The aim of this attack is to avoid tracing and frame innocent users.

• Collusion attack for role's privilege, in which the corrupted users in R = {uik,jk}t k=1 wish to forge a new or unused key in R\{ri1 , · · · , rit}. The aim of this attack is to change the privilege in partial order hierarchy.

We also present a formal security model for two cases of collusion attacks in Appendix A. It is a challenging task to avoid collusion attack since the traitors (corrupted users) have been granted users before they are detected. Traitor tracing is an efficient method to tackle this attack. However, we must ensure that the traitors cannot forge an 'unused' key to avoid tracing but leave some 'foregone' clue of evidence to discover them. The number of colluders |R| = t is an important parameter. A RBC scheme is to be (t, n,m)-collusion secure if for any subset of t in R with |U| = n and |R| = m, the adversary can gain the advantage from R to break this scheme. It is said to be fully collusion secure when it is (n, n,m)-collusion secure.

6. Role-based Authentication

Authentication allows access control systems to gain sufficient assurance that the identity of certain entity is legitimate as claimed. Cryptography-based authentication is widely adopted in current systems because it provides a higher level of security than password-based authentication. In addition, a real-time authentication for high-risk operations is necessary to prevent a user from changing roles after logging in. The authentication on RBAC should support two qualitative classes of identifications:

• User-based authentication, which is used to validate a user's identity, but the systems need to store the user's role information.

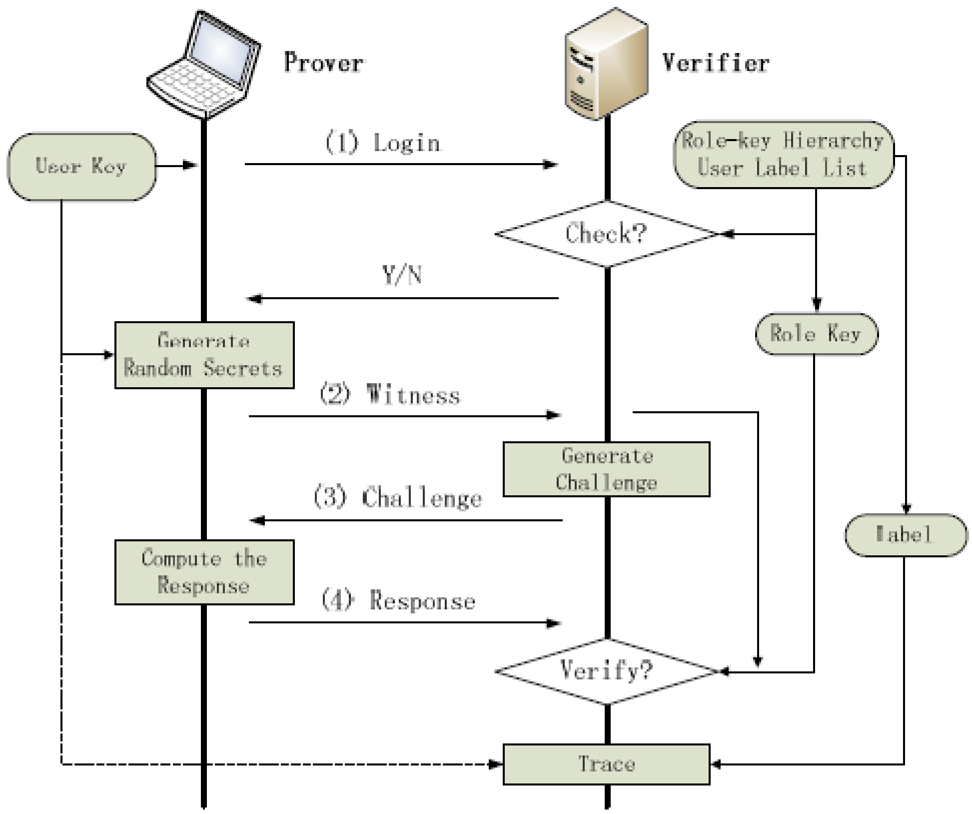

Figure2: Authentication protocol based on RBC.

7. Role-based Encryption

Encryption systems allow users to encrypt resources (files or data) on disk, or synchronously transfer messages among multiple systems. Many encryption file systems have been developed in Windows and Linux environments, e.g., Windows Encrypting File System (EFS), SiRiUS [9] and Plutus [12]. However, these systems implement some trivial schemes where the number of cipher text in the file header grows linearly with the increased number of users who have permissions to access the file. To overcome such a limitation, we introduce a new scheme called Role-based Encryption (RBE), which can be used to improve the performance of existing encryption file systems.

8. Conclusion

We have proposed a role-key hierarchy structure along with hierarchical RBAC model to accommodate the requirements of cryptographic access control for large-scale systems. Based on this hierarchy model, we further proposed several practical role-based security mechanisms to support signature, authentication and encryption constructions on elliptic curve cryptosystem. Our experiments clearly demonstrated the proposed schemes are flexible and efficient enough to support large-scale systems. For our further work, we plan to accommodate other access control features of RBAC such as session management and constraints. Also, our promising results lead us to investigate how emerging distributed computing technologies such as service computing, cloud computing and mobile computing can leverage the proposed schemes with possible extensions.

9. Acknowledgement

This work is supported and made under the esteemed guidance of G. Praveen Babu garu, who is a professor in software engineering department at School of IT, Jawaharlal Nehru Technological University, Hyderabad.

REFERENCES

[1] S. Akl and P. Taylor. Cryptographic solution to a multilevel security problem. In Advances in Cryptology (CRYPTO'82), 1982.

[2] S. Akl and P. Taylor. Cryptographic solution to a problem of access control in a hierarchy. ACM Transaction Computer System, 1(3):239–248, 1983.

[3] E. Bertino, N. Shang, and S. Wagstaff. An efficient time-bound hierarchical key management scheme for secure broadcasting. IEEE Trans. on Dependable and Secure Computing, 5(2):65–70, 2008.

[4] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. In Advances in Cryptology(CRYPTO'01), volume 2139 of LNCS, pages 213–229, 2001.

[5] D. Boneh and M. Hamburg. Generalized identity based and broadcast encryption schemes. InASIACRYPT, pages 455–470, 2008.

[6] D. Boneh and H. Shacham. Group signatures with verifier-local revocation. In ACM Conference on Computer and Communications Security, pages 168–177, 2004.

[7] B. W. D. Boneh, C. Gentry. Collusion resistant broadcast encryption with short ciphertexts and private keys. In Advances in Cryptology (CRYPTO'2005), volume 3621 of LNCS, pages 258–275, 2005.

[8] C. Gentry and A. Silverberg. Hierarchical id based cryptography. In Advances in Cryptology(ASIACRYPT 2002), volume 2501 of LNCS, pages 548–566, 2002.

[9] E. Goh, H. Shacham, N. Modadugu, and D. Boneh. Sirius: Securing remote untrusted storage. In Proceedings of the Internet Society (ISOC) Network and Distributed Systems Security (NDSS) Symposium, pages 131–145, 2003.

[10] J. Jing and G.-J. Ahn. Role-based access management for ad-hoc collaborative sharing. In Proc. of 11th Symposium on Access Control Models and Technologies (SACMAT), pages 200–209, 2006.

