http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 13 (6): 1045-1053, 2014
ISSN 1812-5638 / DOI: 10.3923/itj.2014.1045.1053

Design and Implementation of Mobile System Based Attribute Certificate

A. Xu Dawei and B. Du Yibing
Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China

Abstract: Based on the access control problem of mobile agent system, we introduce attribute certificates into mobile agent system. This study introduces principle of access control based on attribute certificates, including the principle of attribute certificates and policy certificates, access control process, certificate distribution, authorization distribution and verification of user access permissions. The scheme can provide sufficient security, flexibility and high efficiency.

Key words: Mobile agent, attribute certificates, X.509, PKI, PMI

INTRODUCTION

With the rapid development of Internet, various network technologies and applications deepening, it put forward higher requirements for Internet information service based on the Web platform to people. Traditional distributed computing model can not meet the requirements, so mobile agent technology came into being. However, in practice, due to the openness of Internet and mobility of mobile agent, security remains a concern. In order to improve the security of mobile agent, the X.509 certificate is introduced to mobile agent system.

A mobile agent is one program code independented of the operating platform and operating system, it can migrate from one node to another node in the computer network environment independently, with use of appropriate computing resources, accomplish a specific task on behalf of the user (Miaoliang and Yu, 2001). It can reduce network traffic, balance network load, support mobile users, support services customization and it's fault tolerance is good.

Although the mobile agent looks very suitable for distributed applications, but it also brought serious problems, especially security issues, which is the major consideration to design mobile agent system. Because of the complexity of secure mobile agent system, many existing mobile agent system does not provide adequate security mechanism.

ADVANTAGE OF CERTIFICATES

In the mobile agent system, the agents will decide the security policy according to the data transited in the tunnel. The ability to choose various security policies makes the mobile agent system more flexible. To enhance the efficiency and lower the burden of the server, attribute certificates are deployed in authentication.

Mobile agent systems have the following shortcomings as for managing the access control of mobile agents:

- Safety strategy tends to change all the way in applications. But for mobile agent system, the strategies are usually fixed and won't change. Thus applications agent-based applications can only follow the security policy designed by their author. But this policy will not necessarily match that of the application (Brooks, 2004)
- The change of policy-description language and resources of mobile agent system makes it difficult for the application developers to modify and extends the applications
- According to above discussion, the application developers have to consider all kinds of scenarios and try to involve them all. If not, policy will not meet the requirement of application
- The traditional solutions use ACL to solve the problems, which is low in flexibility and efficiency (Elgamal, 1985). For these solutions, the access control is provided by the server, which consumes lots of resources. Further more, the server will lead to a single-point-failure which blocks the whole access control
- Solutions with ACL are not suitable for access control of cosmically deployed system. With great amount of clients, the server will run into shortage of space and time in querying, updating and maintenance of ACL

Corresponding Author: A.Xu Dawei, Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China

In the mobile agent system, the security policy is based on each agent, which is based on the data in the communication channel (Robert, 1996). Agent of the sender and the receiver can be based on different security policies. Sender and receiver use different security policies for mobile agent system to provide more flexibility. In order to improve efficiency and reduce the load, we will combine the attribute certificate-signed and the agent and solve authorization management problem flexible and efficient.

Advantage of attribute certificates:

- Based on users' mark-attribute combination, it can make the access control policy. In the mobile agent system, server can make control policy based on some sensitive attribute values. Attribute certificates improve the make policy efficiency by divide users' attributes and users' mark
- Attribute certificates with short expire date are associated with users' work. When users participate in new projects, they need to apply AA for attribute certificates. Expire date of attribute certificates is the time that agent accomplish this project. With the agent quit, attribute certificates will finish its lifecycle without produce lots of CRL. It will be more convenience to add, modify or delete security policy (Nash *et al*., 2002)

If AA becomes invalid, there will be less affect for only the access control involved this attribute is affected. Every attribute certificate is published by an AA, if some AA becomes invalid, what need to do is establish new AA to publish this attribute certificate and the old certificate will become invalid when it expires.

Design of system: The goal of the system is to implement base environment for mobile agent system to provide security services, to ensure the safety of the agent environment, security of agent transport, security of resource access control and security of the agent communication.

As do the following:

- System should be the structure of B/S, which to meet the requirement of mobile agent and implement authentication safely in internet to ensure the safety of network resources and the legality and secrecy of user access
- System should combine the certificate and mobile agent effectively and use the technologies of data envelope, digital signature based public key scheme to ensure data integrity and non-repudiation
- System needs to increase the agent access control server, monitor the work process of mobile agent and to determine the type of user access to control the authentication and rational use of server access resources

According to the above design objectives, the system needs at least a few basic services, such as encryption, certificate management, certificate issued and secure communications. In addition, it is necessary to combine authentication and authorization access control together for the introduction of the certificate to mobile agent system. Therefore, the system is based on the standard of Public Key Infrastructure to design, mainly composed of two parts of the CA certificate management system and the agent authorization access control management system and communicate through the SSL security protocol. PKI is a universal security infrastructure to use public key technology to implement and provide security services, suited to the environment of large-scale heterogeneous network structure system architecture in Fig. 1.

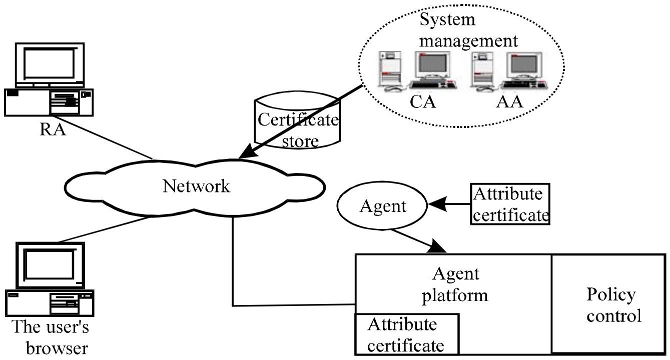

Fig. 1: Mobile agent system based certificate

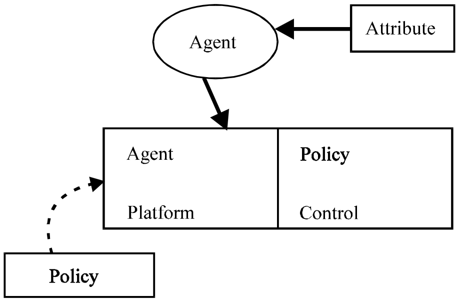

Fig. 2: Attribute certificate based mobile agent system model

CA certificate management system is the core of the entire system, including the response to the user's application, certificate generation, certification issue, key storage recovery, directory services, certificate revocation and other types of services. Certificate/CRL repository is used to store the user's digital certificate and certificate revocation list, certificates and public key for the user query get their own and other users and the latest certificate revocation list. Agent authorization access control is used to determine various types of access for the user to resources.

The basic idea of model is to use two forms of privilege management certificates: attribute certificate and policy certificate to expand the mobile agent system processing environment. This article uses a simple mobile agent model to describe the attribute certificate based mobile agent system shown in Fig. 2. Its realization will be discussed in depth below.

System design: A key problem of model is how to introduce attribute certificates into mobile agent to implement management. Java supports the code transferring, code dynamic downloading, digital signature coding, RMI, object continuous, Heterogeneous platform and so on; all those characteristics provide a ideal basic environment for module. So, java is chosen to implement this module.

Java language provides JVM, the dynamic class load ability and methods invoke ability of JVM provide a simple and efficient way to support agent platform extending (Weiqin, 2007). Java also support JAR file, a kind of ZIP compress file. Using JAR file can manage all class files and other resource as one file and the authentication and integrality of the content in JAR file can be guaranteed by signature. Therefore, one agent class can be packaged after signature in order to transfer to different platforms. This project uses JAR file to protect agent and simplify its management. In addition, there is a database used to store private key information and related digital certificates which is called secret key storage supported by Java.

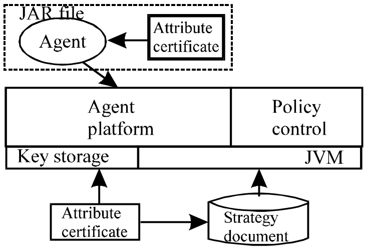

Fig. 3: Java-based model

Using the services provided by Java, we can realize the perfect combination of attribute certificates and mobile agent system, shown as Fig. 3.

In addition, Java provides a standard policy file, which is a special tool used to assign policy rules. Each item in this file describes the admission policy of agent. Policy rules are described using one “approve” language pattern which give out the agent admission policy definitely (Condell *et al*., 1999). All descriptions in the admission policy are the behavior of system objects been authorized. Loader can use specific admission policy to manage the name space, this name space can provide a protection area for all code loading. In addition, developers can define admission policy for special application besides this standard admission policy.

Certificates policy as follow:

- **Agent attributes certificates:** This project uses agility of JAR file carrying information to implement the binding of attribute certificates and mobile agents. One signature JAR file not only includes document file, it also includes a pair of files: one signature instruction file and a data signature file. Each publisher owns one or more files documented. All file are documented in a special directory META-INF, META includes the information of identify certificates to simplify the authentication process in JAR container (Jansen, 2000)

There is a container in agent JAR file used to store the attribute certificates published to this agent. Once the mobile agent is put into JAR file, it can be bind with attribute certificates. This attribute certificates are stored in directory META-INT to be used in the future. Certificates describe policy rules assigned to agent; those policy rules are stored in the attributes of attribute certificates in standard policy file format.

In addition, we will add other useful information to attribute certificates as extend area. Extend area includes constrain instruction area and service restore area. Constrain instruction area used to indicate whether the entity of publishing certificate is terminal user and whether assign special right again to specific number quit users; service restore area used to whether to extend the lifetime when attribute certificates has expired (Boeyen, 2000).

Platform attribute certificates

Description of policy rules: Edit of policy rules is based on agent platform and does not care certificates publisher not related with platform. Under the "admission" language pattern, in order to authorize agent, the direct way is to define an admission license managed by Farrell and Housley (2002).

Under the "admission" language pattern, policy rules begin with name "right adjusting" and then set the alias of user's secret key shown as "*" and end with actor operation. In this way, a simple description can implement code access right described in attributed certificates. For example, if authorize right to a trust agent developer ESO a encapsulated agent code, Java can describe policy rules like this:

- Grant
- Permission privilege Adjustment"*" "launchedBy"
- Permission privilege Adjustment"ESO" "sealedBy"

Using this way, it bring lots of agility to system, once there is a necessary to add some right to system to process special business, what need to do is just add related description of policy rules.

Function implement of platform attribute certificates: Attribute certificates in this project includes following functions:

- Ability to manage the certificates number based on the right level of publisher
- Ability to decide who owns the right to change accessing admission
- Ability to adjust the anonymous certificates right level

Instruction of right level exists in the extend area of platform attributes certificates, including level and certificates number. In general, user is the default owner of the objects created by him. He has the right to adjust the right level and even adjust the authorized management right without any authorization. However, if after the authorizing, all the operation to adjust the right level will be denied with a warning given out. Description of platform attributes certificates includes the control information of calculating resource; therefore, it has the highest priority. This description can use sparse matrix to adjust the right level of other users. This method is very smart, it allows every publisher control his right level separately (Adams and Farrell, 1999).

The current Java edition includes some new features: including defining new security attributes, assigning exchange class for standard policy class, defining new admission policy, allowing trust code to exist in directory. All those features can be used in our authorized management mechanism. Defining new security attributes can let operator to locate position, validate and transform the control policy format. Defining new admission policy can adjust admission controlled by standard policy entrance. Trustable extension can realize the all right access to resource by changing the special management component into a virtual directory.

Principle of access control based on attribute certificates: The correspondence relation of certificates, users, roles and permissions in model:

- **Users:** Hold public key certificate, proof of identity
- **Roles:** Reflected through the attribute certificate
- **Permissions:** Reflected the access control policy through the Policy certificate

The process of the agent's access control is: Receive access to the proxy agent platform, it is to determine the validity of the certificate of agent first and then to determine whether the role of agents installed in the attribute certificate, the final security policy platform established by policy certificate form a safe environment for the implementation of agent.

The principle of attribute certificates and policy certificates shown in Fig. 4. User 1 and User 2 both apply

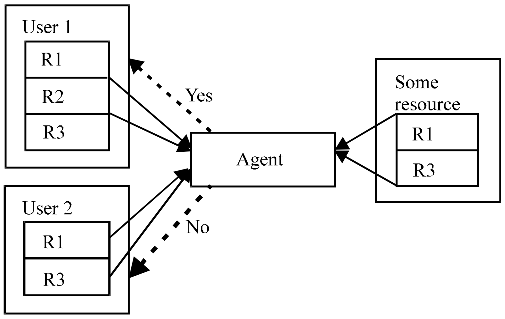

Fig. 4: Access control process

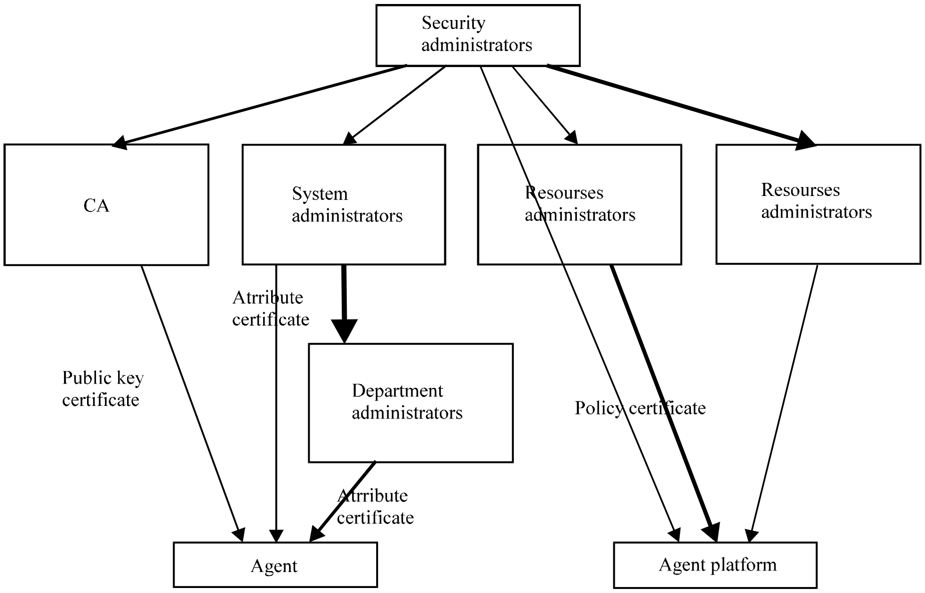

Fig. 5: Certificate distribution

for a resource, the user1 include the roles of R1, R2 and R3, the user 2 include the roles of R1 and R2. Resource policy rules require permission include the R1 and R3. So, by comparison, the user 1 will to be allowed access the resources and the request of user 2 will be refused.

Certificate distribution: In the mobile agent system, the certificate-based access control, it need to issue certificates for each agent to give the authority to determine security policy of resource. Because the system has several roles, different levels system administrator can define and manage roles of different range. First, security administrators create a variety of roles, the lower administrator issue certificates following the PKI/PMI specifications (Dengguo, 2001). Certificate distribution shown in Fig. 5.

With public key certificates, authority certificates can form a authorization chain, that is, in addition to agents and agent platforms, authorization certificates can be issued to the designated individual. When necessary, these individuals can request the privileges once again and assign to other entities in the form of issuing the certificate, to form the authorization chain. Therefore, this mechanism can support the authorization and commission with a variety of ways. Such as a "push" mode: before the privileges assigned to the agent, agent user use their own public key certificate to get attribute certificate issued by AA first and then refer the platform to verify; another example of a "pull" mode: the agent platform accessed contact with the AA to query the agent user's attribute certificate. Generally policy of mobile agent system defined by a fixed, inflexible, but authorization certificates can provide flexibility and the formation of the authorization chain (Sandhu and Coyne, 1996).

Authorization distribution: Attribute certificate contains policy rules of the agent and policy certificate contains the security policy of agent platform's to manage all the mobile agent's behavior to access the agent platform to make the attribute certificate role to be recognized, the most direct way is to define a license, a license P can be described by a triple:

- P (R_i, M_j, O_k) express that the role R_i's operation by calling method j (M_j) of the Object K (O_k)

Then a license of the role Ri can be expressed as one Ordered pair (method, object):

$$P_{j,k}(R_i) = (M_j, O_k),$$

On role R_i of a user U_φ its license sets is the union sets of R_i's license in U_q:

$$Pu_q(R_i) = \cup \{P_{jq,\, kq}(R_i)\},$$

kq = 1 ... m   m is the number of objects in U_q
jq = 1 ... n   n is the number of methods in each object

In concrete realization a set of access control rules for the design of security policy are as follows:

A role R_i is allowed to perform operations P on some resource O_A, only when: Policy rule does not explicitly prohibit permission of R_i performing operations P on O_A and:

- Explicitly authorize the permission of R_i performing the operation P on O_A. Or
- R_i have permission to perform the operation P on O_B, the ancestors node resources of O_A and between O_B and O_A or any ancestors node resources of O_A R_i have not been explicitly prohibit permission of performing operations P. Or:
- R_i inherited R_j and authorize the R_j permission of perform the operation P on O_A

External reference policy file library of policy Certificate list the Master Properties of each entity and permission of entities. When needed, it is just loaded the file address to the policy certificate. When the entity access some resource, you can verify that the appropriate permissions. Reference policy file use a "name space", examples such as shown in the Table 1.

Verification of user access permissions: Suppose it is to verify whether the user u* have permission to operate a* on the files f*, verify steps as follow:

- According to the access control rules, the elements of the licenses and all the elements of meet the constraints form a set of P*: All the main form set S*, operation constitutes a set A*, a collection of objects constitute the O*
- The operator action $(s, O_1) = A_1$, where $O_1 \leq O^*$, $s \in S^*$, means: in the collection of P* it is to find all the elements whose subject is s and whose object $o \in O1$ and these elements constitute the set of A_1
- The operation check $(s, a, o) = cr$, where, $s \in S^*$, $a \in A^*$, $o \in O^*$, $cr \in \{yes, no, unspecified\}$, is used to test in the combination of P* whether grant the subject s the access permission of performing the operation a on the object o by the definition of subject s. If it is granted to return yes, refused to return no, if it is not specified return unspecified

Table 1: Node specific offline parameters

Name space string	Policy identifier	Licensing policy rules
Sdu.edu.cn/U1	Sdu.edu.cn/U1	Sdu.edu.cn/U1
Sdu.edu.cn/U2	Sdu.edu.cn/U2	Sdu.edu.cn/U2
	Sdu.edu.cn/U1	Sdu.edu.cn/U1
Sdu.edu.cn/U1/A	Sdu.edu.cn/U1/A	Sdu.edu.cn/U1/A
	Sdu.edu.cn/U1	Sdu.edu.cn/U1

Validation process is divided into two steps: First, it is that verify the main constraints and object constraint, which get effectively sets, then according to the relationship between the main body, actions and objects, search the previous collections, which "grant" or "reject" the user's access permissions.

SYSTEM IMPLEMENTATION

Based on the above design of the system, the key technology of implementation process is the encryption key, the certificate format and code processing, communication based SSL protocol and access control agent, the following is a analysis in detail.

Encryption: The system use encryption algorithm, including triple-DES symmetric encryption algorithms, public key algorithm RSA, message digest algorithm SHA. These algorithm can achieve the basic requirements for encryption of CA, encryption levels structure as shown in Fig. 6.

Encryption algorithms, certificate generation, signing and user interface between the layers are separate, the upper layer use the lower layer services provided by calling the function.

Certificate format and code process: Digital certificate issued by a certificate authority CA, including user identity information, user public key information and authentication of digital signatures and other data, it is equivalent to life identification card in the networked world. Public key certificate use the format of X.509v3. X.509 is part of X.500 series standards of the ISO and CCITT/ITU-T's. The specification definite and standardize a common, flexible certificate format (Zhensheng, 2002), which has been widely adopted.

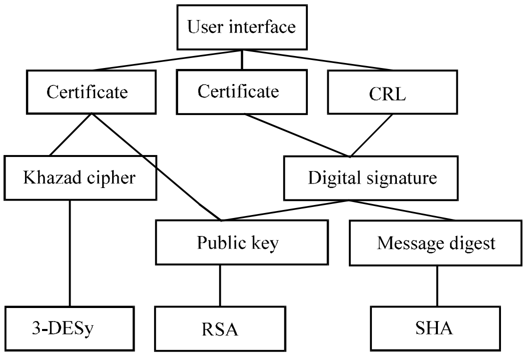

Fig. 6: Encryption levels structure

X.509 certificate format uses ASN.1 (Abstract Syntax Notation One) to describe the syntax. ASN.1 is used to describe structured information marker language established by the ITU-T, has become the international standard, whose information described needs to be passed in the communication media. ASN.1 also provides the abstract syntax and encoding, encoding is transfer syntax, you can choose a different encoding method in terms of transmission (Chadwick and Otenko, 2003). ASN.1 encoding rules include the BER (Basic Encoding Rules), DER (Distinguished Encoding Rules), CER and PER, etc. (Blobel *et al*., 2003). Description of the ASN.1 BER encoded string of bits, said 0,1, DER is a subset of BER, to ensure that the encoded unique. The system use the abstract syntax of ASN.1 and the DER encoding rules. The specific implementation will be described as the following:

```
ASN.1 grammar description of the certificate subject:
Certificate:   : =SEQUENCE{
   tbsCertificate          TBSCertificate,
   signatureAlgorithm     AlgorithmIdentifier,
   signature               BIT STRING
}
```

TbsCertificate contains the contents of the certificate, signtureAlgorithm contains the signature algorithm of CA sign to certificate, signature contains the results of the digital signature on the certificate.

The basic coding guidelines of DER are: the value of each data transferred, composed by TLC of three fields: identifier, the length of data fields and data fields. The design rules of the encoding library system are recursive, a structured object code is to link the object code of the members, using this approach, all objects code can be reduced to the basic object code sequences. The system will define all types as the class and define encoding and decoding function of corresponding to the various types as public member functions of a class:

```
DER rules Prototype:
     int DER_Encode(CBitStreame Buffer*);
     ULONG DER_Decode(CBitStreame Buffer*);
     bool IsPrimitive(void);
```

To facilitate the use and ease of expansion, the system uses ASN.1 abstract syntax to encapsulate the data type with the idea of object-oriented. ASN.1 description of the certificate subject is a structured data type, each structured data object corresponding structured data type has its own method function for coding and decoding. Therefore, the certificate data object as follows:

```
SEQUENCE()
   {
      int DER_Encode(CBitStreame Buffer* pBitStreameBfuffer);
      ULONG DER_Decode(CBitStreame Buffer* pBitStreameBfuffer);
      bool IsPrimitive(void);

      TBSCertificate  tbsCertificate;
      AlgorithmIdentifier signatureAlgorithm;
      BIT_STRING signsturevalue;
}
```

The implementation of encoding functions DerEncode of structured data object: (1) Call the base class method DER_ENCInit()first; (2) Then call coding method function as the sequence of members variable appears; (3) Call the method DER_ENCEnd() finally, the encoding process is completed. Similarly, DerDecode realization principle: (1) Call the base class method DER_DECInit (); (2) Then call the decoding method function as the order of members variable appears; (3) Call the method DER_DECEnd () finally.

Thus, the data descriped by ASN.1 is encoded in the sending end with the BER, changed into a unique string of bits, decoded with a BER at the receiver to receive the data expressed in the ASN.1 of the bit string.

Secure communication: Follow the rules of RFC2830 (Lightweight Directory Access Protocol (v3): Extension for Transport Layer Security), the model use TLS to provide secure communication guarantees. TLS was formerly known as Secure Sockets Layer protocol (SSL: Sockets Layer Protocol), which was first proposed and used by the Netscape browser and now TLS has been widely recognized by the world. In the OSI layered structure which is located in between the TCP and application layer protocols (HTTP, SMTP, FTP).

TLS is composed of Record Protocol, TLS Handshake protocol and data transmission compression and encryption, before the application layer protocol transfers data, the Handshake protocol allows mutual authentication between the client and server and negotiation of data encryption algorithm and key, record protocol is used for packaging a variety of high-level protocol, making the TLS protocol has high transparency.

TSL runs SSL handshake protocol to achieve the client and server for secure handshake connection first, which determine the parameters of cryptographic algorithms, keys and compression, then encrypts data, sent to the lower layer and the key exchange is processed during the handshake time. With the security policy and technical perspective, TLS has a strong safety. Record protocol completes the transmission of application data and SSL Control data, it will divide application data, transfer between client and server together with SSL control data.

Authentication access control: External policy control is in charge of analyzing and validating the certificates related with agent. Validating content of certificates include signature and certificates link, polishing all the certificates not related with agent platform and making the order according the right. External policy control is also in charge of extracting related platform policy form the agent platform, which is established by platform attribute certificates when initializing the platform. Certificates links is a complicated validating process, including expire date, possibility of repeal and the necessary of information restore.

Function of internal policy control is using the information provided by external policy control and provides security environment for agent platform. Internal policy control processes the content of valid attributes certificates related with platform policy and obtain the conclusion about whether allow this agent access resource and with which right to access those resource.

For extending, in this project, internal engine implement related simple task and adjust the adaptation of program. External engine processes those public and general but complicate explaining and validating tasks.

Policy control is implemented by a pair of objects and in charge of necessary calculating and deciding the agent right. Right is decided by a authorized admission policy file, including attribute certificates and policy rules associated with platform attributes certificates. No right means it has not been authorized and will be not admitted when processing. During initializing, policy control class is assigned to agent platform as a trustable component through the entrance of standard security attributes file or system attributes file. Since both policy control parts are Java classes, it makes problem simple. Therefore, when processing multiple processes and multiple applications, one agent platform can support multiple policy control. There is a key "policy mark" defined in the extend area of all certificates used to comply the policy control and certificates.

External policy control supports a single method: validates attribute certificates and returns the admission policy. Once all certificates are analyzed and reordered, external engine will call the internal engine to do processing. Since all exchange classes in Java standard control policy mechanism keep integrate, policy control can simplify the implement course during processing.

CONCLUSION

According to the security requirements of mobile agent system, especially the problem of access control, the author analyzes the features of mobile agent and the shortcomings of current schemes and points out that attribute certificates should be introduced into mobile agent system in order to extend the processing environment of mobile agent system and implement authorization management more efficiently. The scheme can provide sufficient security, flexibility and high efficiency.

All above shows this mechanism provides enough agility to include a more extendable security policy and can adapt most of the current mobile agent application. At present, with the economic development, economy and Internet are combined more and more tight, the electronic business scale becomes larger and larger, internal mobile agent technology can be applied to electronic business with bright foreground and PKI/PMI can provide more security policy for electronic business.

ACKNOWLEDGMENTS

This study was supported by the access of the Beijing Institute of Technology which supported by Shandong Provincial Department of Education and this work was supported in part by Shandong Provincial Natural Science Foundation (ZR2011FL027), National Natural Science Foundation (61101162\61101085\60873041\61272091) and Program of Integrating Informatization and Industrialization of Shandong Province (2012EI020/2012EI025).

REFERENCES

Adams, C. and S. Farrell, 1999. Internet X.509 public key infrastructure certificate management protocols. The Internet Society. http://www.ietf.org/rfc/rfc2510.txt.

Blobel, B., P. Hoepner, R. Joop, S. Karnouskos, G. Kleinhuis and G. Stassinopoulos, 2003. Using a privilege manage-ment infrastructure for secure web-based e-health applica-tions. Comput. Commun., 26: 1863-1872.

Boeyen, S.X.509, 2000. Overview of PKI and PMI Frameworks (4th Edition). http://www.entrust.com/resourcecenter/pdf/509_overview.pdf

Brooks, R.R., 2004. Mobile code paradigms and security issues. IEEE Int. Comput., 5: 54-59.

Chadwick, D.W. and A. Otenko, 2003. The PERMIS X.509 role based privilege management infrastructure. Future Gener. Comput. Syst., 19: 277-289.

Condell, M., C. Lynn and J. Zao, 1999. Security policy specification language. Internet Draft.

Dengguo, F., 2001. Computer Communication Network Security. Qinghua University Press, Beijing.

Elgamal, T., 1985. A public-key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inform. Theory, 31: 469-472.

Farrell, S. and R. Housley, 2002. An internet attribute certificate profile for authorization. The Internet Society. http://www.ietf.org/rfc/rfc3281.txt.

Jansen, W.A., 2000. Countermeasures for mobile agent security. Comput. Commun., 10: 1667-1676.

Miaoliang, Z. and Q. Yu, 2001. Summary of mobile agent system. Comput. Res. Dev., 38: 16-25.

Nash, A., W. Duane, C. Joseph and D. Brink, 2002. Public Key Infrastructure (PKI) Implementation and Management of Electronic Security. Qinghua University Press, Beijing, China.

Robert, G.S., 1996. Agent tcl: A flexible and secure mobile-agent system. Proceedings of the 4th Annual Tcl/Tk Workshop, (ATTW'96), Monterey, California, USA., pp: 9-23.

Sandhu, R.S. and E.J. Coyne, 1996. Role based access control models. IEEE Comput., 29: 38-47.

Weiqin, S., 2007. Java Solution for Network Programming. Publishing House of Electronics Industry, Beijing.

Zhensheng, G., 2002. Public Key Infrastructure (PKI) and Certificate Agency(CA). Publishing House of Electronics Industry, Beijing.