Asian Journal of Information Technology 5 (12): 1323-1327, 2006
© Medwell Journals, 2006

Time Bound Authorization in PKI

[1]S. Radhakrishnan, [2]A.Chandrasekar and [2]N.V Ballaji
[1]HOD-CSE, A.K College of Engineering, India
[2]CSE, St.joseph's College of Engineering, Chennai-119, Tamilnadu, India

Abstract: A Public Key Infrastructure (PKI) integrates software, hardware, encryption technologies- and services for managing the cryptographic infrastructure and users' public keys. It uses digital certificates as an authentication mechanism and is built to better manage certificates and their associated keys. The mechanism of certifying and revoking of public keys, private key escrow has been explored. But others have given less attention to regulating access to stored keys. In this study, we are proposing extensions to PKI such as key escrow, a protected use of keys and recovery. We discuss an exclusive protocol known to be Authorization certificate protocol, which allows owners to authorize others to perform various actions, based on their keys (say decryption) perhaps preventing them from knowing the private key. Also this protocol provides more security by canceling out the authorization once the allotted period is over. ACP defines a set of rules for owner to authorize others security and describes how a registration authority and issuance process should get changed in order to enable time bound authorization in PKI.

Key words: RegistrationAuthority, ACP, authorization, PKI

INTRODUCTION

Assuring privacy of communication and data storage is an integral structure of our information infrastructure. The main purpose of cryptography is to make things very difficult for a third party to get access to the secured information. At the same time if the third party is an authority who believes that they have the rights to access the information by legal or social reasons then employing cryptography for security purposes poses lot of problem in that scenario. In this study we need some mechanism to permit the authorized persons to view our information at the same time information should be protected from other third party. Generally those systems are called as key escrow and we have several mechanisms to deal those issues.

Key escrow[1] is arrangements in which the keys needed to decrypt encrypted data are held in escrow by a third party, so that someone else (typically government agencies) can obtain them to decrypt message which they suspect to be relevant to national security.

Authorized third parties who are allowed to eavesdrop may be a Government Organization who wishes to eavesdrop its own citizen or a Corporation seeking to its own information protected by its own employee.

Generally Key Escrow is useful for data recovery and key recovery. Many commercial systems are available for key escrow. Government key escrow is useful to the government but not the user as seen by Clipper system where when key of a person is lost government key escrow is not useful in getting back that key. So to overcome that, many commercial systems are built which allow authorized party to eavesdrop and also help to recover the key when key is lost.

Here we propose a method, which will be useful in corporate environments. In this proposal one can give authorizations to a third party to see this message based on some keywords at the same time protecting his private key. This authorization is for a particular period only. After that period authorization for that person is automatically cancelled.

Related works: The existing commercial systems are either used for Recovery of data or recovery of keys but not on private key escrow for authorization[2].

The Clipper/Capstone Chips[3], The Bell Atlantic Yaksha system[4] is used for data recovery purposes.

Cylink Key Escrow uses Diffie-Hellman Techniques for integrating key escrow services into a public key infrastructure.

Micali and Sidney Resilient Clipper[5] like key proposal allows keys to be split so recovery is possible even if some of the escrow agents fail to produce their key components. Micali Guaranteed Partial key escrow. In this proposal the private keys of users are partially escrowed. The escrow agents verify that the bits in their possession are correct and only a relatively small number of bits are unescrowed.

Corresponding Author: S. Radhakrishnan, HOD-CSE, A.K College of Engineering, India

Threshold Decryption[6] is used to share a secret key by a group of escrow agents' in such a way that collaboration of the agents information can be decrypted without the agents releasing their individual key components. In TIS Commercial key Escrow, data recovery is enabled through master keys held by a data recovery center[7].

Public key infrastructure: PKI is the acronym for Public Key Infrastructure. The technology is called Public Key because unlike earlier forms of cryptography it works with a pair of keys. One of the two keys may be used to encrypt information, which can only be decrypted with the other key. One key is made public and the other is kept secret. The secret key is usually called the private key. Since anyone may obtain the public key, users may initiate secure communications without having to previously share a secret through some other medium with their correspondent. The Infrastructure is the underlying systems needed to issue keys and certificates and to publish the public information.

PKI is an infrastructure that uses digital certificates as an authentication mechanism and is built to manage certificates and their associated keys. A PKI can be implemented within an organization for the use of users on its network or it can be a commercial entity that issues certificates to Internet users. In both cases PKI has the following components.

- A registration Authority that verify the identity of the user/requester.
- A Certificate Authority to issue certificates
- Policies that govern the operation of PKI
- Issuance, management and revocation of certificates

PKIs are important elements in network and Internet security because may communications such as business and E-commerce transactions are dependent on a reliable method to identify the parties of the transactions.

Operation: Registration Authority delivers the certificate application to users connected to it. Users who want certificates, fills up the application, generate the key pair, send public key along with the certificate requisition format to Registration Authority. (PKI itself can generate the key pair for the users, which has certain advantages and disadvantages). RA verifies the details and if found ok, it generates certificate request and send to Certificate Authority. CA generates certificates in X509 format[8] and sends it to RA. RA sends it back to the user and saves a copy in its database. In some PKI CA itself will do the operation of both CA and RA, which has certain advantages and disadvantages.

The problem: Alice is a senior manager who deals with some important information. He is going on vacation for a period. During that period there is a possibility he may get some important documents. In his absence others cannot read it as it is in encrypted format. To avoid this situation Alice decides to give authorization to Bob. Bob can see the messages but he should not know Alice's private key.

This study typically addresses the problem like this and we will see how our ACP solves this problem.

Proposed architecture

Communication server and business server: Besides the authorized users, in our architecture, there is a centralized online server, as in the traditional single user-oriented PKI. It is this online server that makes authorization possible.

It can be known as communication server since it controls the communication path of users as well as communicate with the business server. (All the business logic (threshold scheme, defined later) can be kept in a separate server.

For implementing the business logic required for threshold scheme (defined later), there can be two approaches.

- Keeping it as a separate process in the same centralized online server.
- Keeping it in a different business server.

Even though having it in a same online server looks like a handy approach, the compute intensive operations that needs to be done will definitely make a server slow down. So, the second approach will be the best one to implement.

Proposed PKI policies: For PKIs implementing ACP, RA must be able to classify the owner's certificate and the authorized person's certificate. The certificate issued to them or the information revealed to them must differ accordingly based on the classification done by RA.To do this, RA should be implemented in such a way that it understands what a AC is and how it should behave when it gets AC from the user. The issuance process of PKI should also be changed when it receives the request for key escrowing. Based on the key escrowing mechanism (defined later) the certificate issued to the owner in such a way that the information on the certificate can be used for decryption but can't be revealed by the owner.

Design principles: There are several design principles for implementing our time bound authorized PKI. Some are straightforward while others are not.

- Centralised server can only be activated either by an owner or an authorized person. Apart from giving public key certificate, it is responsible for distributing the authorization certificate.
- Only an online server can utilize business server by issuing an authentication certificate.
- The communication between online server and business server needs to be more secure since private key is shared between them.
- No full trust is based on the authorized person. He/She should not know the private key so that the person cannot misuse the key once the allotted period is over. Thus, our scheme typically ensures the misuse monitoring process.
- Other than the business logic, the business server is expected to store nothing. It should not store the key in any form in the storage device. Private key should get erased as soon as the process gets over.

Authentication mechanism: If a set of users is provided with dedicated personal computers that have no network connections, then a user's resources and files can be protected by physically securing each personal computer. When a centralized server instead serves these users, then that particular server must provide the security. The operating system can enforce access control policies based on user identity and use the log-on procedure to identify users.

Today neither of these scenarios is typical. More common is a distributed architecture consisting of dedicated user workstations (clients) and distributed or centralized servers. In this environment, three approaches to security[8] can be envisioned:

- Rely on each individual client workstation to assure the identity of its user or users and rely on each server to enforce the security policy based on user identification[9].
- Requires that client systems authenticate themselves to the servers, but trust the client system concerning the identity of its user.
- Require the user to prove identity for each service invoked. Also require that servers prove their identity to clients.

In a more open environment where network connections to other machines are supported the third approach is needed to protect user information and resources housed at the server. But in our closed environment, either of the first two approaches can be implemented.

Key escrow (Threshold scheme): On receiving the request for Key Escrowing[10] from Alice, Server divides the Private key of Alice say K in to n pieces K1, K2…Kn such that

- Knowledge of any k or more pieces makes K to be computable
- Knowledge of any k-1 or fewer pieces leaves K completely incomputable

Server sends the pieces to n members.

To divide K into n pieces, server picks a random k-1 degree polynomial[11]

q(x)=a0+a1x+… ak-1xk-1 in which a0=D and evaluate

D1=q(1)…Di=q(1)…Dn=q(n)

Given K, server picks a prime number, which is bigger than K and n. The coefficients a1, a2…ak are randomly chosen from a uniform distribution over the integers in [0,p]. The values D1, D2…Dn are computed modulo p.

On receiving the authorization certificate from Bob, Server sends messages to all n members who have the pieces of K.After getting a minimum k values from members together with their identifying indices server finds the coefficients of q(x) by interpolation and then evaluate K = q(0) which gives K.

Authorization certificate protocol: Normally server is responsible for giving Public key certificates to users. When users requests, it creates public-private key pair, issues public key certificate.

In our proposal, communication server other than giving Public Key certificates it is responsible for distribution of Authorization Certificate.

In an unprotected network environment, any client can apply to any server for service. The obvious security risk is that of impersonation. An opponent can pretend to be another client and obtain unauthorized privileges on sever machines. To counter this threat, our communication server must be able to confirm the identities of the clients who request service. To provide this confidentiality every interaction between client and server is being encrypted by the public key of the corresponding sender, thus enabling only the authorized persons to decrypt it.

Steps to be followed for authorization

- Alice sends his pair for Escrowing purpose. On receiving the request from Alice for escrowing, server perform Threshold scheme, which is already discussed. When Server is generating the key pair this step may not be necessary.

- Alice sends authorization request to SERVER.

A→SERVER:Ek_{US}[N1||IDa||IDc|
Ek_{RA}|| IDa ||Auth.Req]
Authorization Request → N2 || TD

Since Alice wants to get authorization certificate for Bob, it sends the request to SERVER. This request is encrypted by public key of SERVER. It includes Authorization request, which consists of a nonce N2, time duration TD.This authorization request is encrypted by private key of Alice for authentication along with IDs of Alice and Bob. The nonce N2 is included for identification of a particular authorization certificate. Since the identity of the person who is going to get authorization powers should be disclosed to others it is also included in the message which is getting encrypted so that only server can know (confidentiality).

TD is typically being included to avoid replay attack. The opponent would be able to reuse the authorization request to spoof the server. To counter this threat, our authorization request includes a time duration indicating the length of time for which the person is authorized to use the certificate.

- SERVER sends authorization certificate to Alice

SERVER→A:Ek_{UA}[N1||Ek_{US} [Auth. Certificate]]

Authorization Certificate: [IDa|| IDc|| TS || N2]
Timestamp TS: [T || TD]

The SERVER sends the authorization certificate encrypted by its own public key. It contains ID of Alice and Bob, Time stamp TS and nonce N2 associated with this certificate. Message also includes N1, which is sent by Alice along with its request. This total message is encrypted by public key of Alice thereby only Alice can decrypt it. The authorization certificate includes a Time Stamp (TS) indicating the date and time at which the authorization certificate was issued (creation time T) and Time Duration (TD) indicating the length of the time for which the authorization certificate is valid. This TD will be same as what Alice sends to server during authorization request.

- Alice sends authorization certificate to Bob

Alice→Bob:Ek_{UC}[PKC_{A} |Auth. Certificate]

PKC_{A} = Public Key Certificate of A

After getting the certificate for Bob, Alice forwards the certificate to Bob along with its public key certificate. This is encrypted by public key of Bob so that only Bob can decrypt. This is the authorization certificate, which is required to be reproduced to the registration authority of the communication server in order to get access to the private key of Alice.

- When Bob wants Alice's message to be Decrypted

Bob→SERVER:Ek_{US}[N3]|||[Auth.Certificate]

When Bob wants Alice's message to be decrypted, it sends the authorization certificate to the server. It also sends a nonce N3 encrypted by public key of server.

- Server sends the decrypted message to Bob________ encrypted by Bob's public key.

SERVER→ Bob: Ekuc[N3|Ek_{RS} [MESSAGEa]]

SERVER verifies the authorization certificate sent by Bob for its originality and validity by first decrypting certificate with its private key. It checks the time stamp to verify whether the person is authorized enough to view the message or not. Server sends message to the members, which contains piece of escrowed private keys of Alice. After receiving pieces from k members it performs interpolation to get original key of Alice. As we discussed in the architecture section these operations can be performed in a separate business server, which makes the process easy. Server then decrypts the files with the escrowed private key of Alice. It also verifies that message should have been sent in a date after the authorization certificate is created. Then it encrypts it with its private key for authentication and along with N3. The entire message is encrypted by Bob's public key.

Then it enters the name of the files, which is sent to Bob, into the log of Alice. Then it destroys the private key of Alice.

Security services: Security has been defined as "protecting information system from unintended access".

Network security measures are needed to protect data during their transmission, because network and communication facilities are used for carrying data between user and computer and between computer and computer. The goal of any proposed protocol is to ensure that the users of a network receive the information technology services with the quality of services that they expect. Lets see how our Authorization Certificate Protocol (ACP) provides various security services[12] which need to be taken care of by any proposed protocol.

Confidentiality: As we mentioned before, here, every interaction between client and server in ARP is being encrypted by the public key of corresponding sender thus ensures that the transmitted information are accessible only for reading by authorized parties.

Authentication: The registration authority of a communication server in a proposed PKI verifies whether the person is authenticated to use it or not. In case of authorization, authorization certificate (which cannot be modified by any opponent) sent by the communication server provides an authentication for a person. Thus authentication service has been provided.

Integrity: Since we are not even disclosing the private key to an authorized person, this method ensures that only the owner (authenticated parties) can be able to modify computer systems assets and transmitted information.

Access control: Time stamp included in the authorization certificate enables access control by providing the authorized person to use it for only a specific amount of time.

Availability: It requires that computer system assets be available to authenticated parties when needed. This is typically being done by implementing the time bound authorized PKI.

CONCLUSION

In the light of the growing multi-faceted threats, disclosing our private information to the team member or work group is also not realistic. But in some scenarios, as we described before, it is becoming very necessary to authorize some other persons to view our message in our absence. Thus our approach time bound authorization in PKI enables us to authorize others perhaps not disclosing the private key. Access control has also been implemented by including a time stamp in an authorization certificate. Our ACP has also been seen satisfactory in terms of providing security services.

REFERENCES

1. Key escrow definition from free online dictionary Wikipedia. http://en.wikipedia.org/wiki/Key_escrow
2. Pierangala, S., M.K. Reiter, Sushiljajodia An Authorization Model for a Public Key Management Service ACM Transactions on Information and Systems Security, pp: 453-482.
3. Dorothy E.Denning and Dennis K. Branstad Taxonomy for Key Escrow Encryption Systems Communications of the ACM, pp: 33- 39
4. Ravi Ganesan, The Yaksha Security System Communications of the ACM, pp: 54-60
5. David, M.B., C.M. Ellison, S.B. Lipner and S.T. Walker, 1994. A New Approach to Software Key Escrow Encryption, Trusted Information Systems, 3060 Washington Rd., Glenwood, MD, draft of August.
6. Identity-Based Threshold Decryption, 2003. Joonsang Baek, Yuliang Zheng.
7. Stephen T.Walker, Steven B.Lipner, Carl M.Ellison and David M. Balenson Commercial Key Recovery Communications of the ACM, pp: 40-46.
8. ITU-T Recommendation X. 509, 1997. The Directory: Authentication Framework.
9. Simple authentication and security layer SSL mechanisms [RFC3163] R. Zuccherato and M. Nystrom, ISO/IEC 9798-3 Authentication SASL Mechanism, RFC 3163, August 2001. [RFC4505] K. Zeilenga, Ed., Anonymous Simple ... www.iana.org/assignments/sasl-mechanisms
10. Adi Shamir, How To Share a Secret. Communications of the ACM, pp: 612-613
11. Liu, C.L., 1968. Introduction to combinatorial mathematics. McGraw Hill.
12. Security policies standards http://www.information-security-policies-and-standards.com/